Resource management in wireless networks

Raftopoulou, M.

DOI
10.4233/uuid:1e9082db-02d5-44d6-8c3e-c08932162d65

Publication date
2024

Document Version
Final published version

Citation (APA)
Raftopoulou, M. (2024). *Resource management in wireless networks*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:1e9082db-02d5-44d6-8c3e-c08932162d65

Resource Management in Wireless Networks

Resource Management in Wireless Networks

Dissertation

for the purpose of obtaining the degree of doctor
at Delft University of Technology,
by the authority of Rector Magnificus Prof. dr. ir. T.H.J.J. van der Hagen,
chair of the Board for Doctorates,
to be defended publicly on
Friday 19 April 2024 at 12:30 o'clock

by

Maria Raftopoulou

Master of Science in Electrical Engineering,
Delft University of Technology, The Netherlands and
Master of Engineering in Electrical and Computer Engineering,
National Technical University of Athens, Greece
born in Lemesos, Cyprus.

This dissertation has been approved by the promotors.
Composition of the doctoral committee:

Rector Magnificus	chairperson
Prof. dr. ir. P.F.A. Van Mieghem	Delft University of Technology, promotor
Dr. R. Litjens, MSc	Delft University of Technology & TNO, copromotor
Independent members:	
Prof. dr. K.G. Langendoen	Delft University of Technology
Prof. dr. ir. G.J. Heijenk	University of Twente
Prof. dr. ir. S.M. Heemstra-de Groot	Eindhoven University of Technology
Dr. ir. E. Smeitink	Delft University of Technology & KPN
Prof. dr. ir. R.E. Kooij	Delft University of Technology, reserve member
Other members:	
Prof. dr. H.V. Poor	Princeton University

This research is part of NExTWORKx, a collaboration between TU Delft and KPN on future communication networks.

Keywords: radio resource management, wireless networks, network slicing, federated learning, vehicular communications, drone networks, random geometric graphs

Printed by: Ipskamp Printing, Enschede

Front & Back: Designed by Nils R. Kohrs

ISBN 978-94-6384-560-1

An electronic version of this dissertation is available at http://repository.tudelft.nl/.

To my family and Nils.

CONTENTS

Summary

Following the trend of previous years, the number of devices, and hence the traffic in cellular networks is increasing. Moreover, new applications with stringent requirements are envisioned. Examples of such applications include collaborative learning and coverage extension with drones. To accommodate the traffic with its respective Quality of Service (QoS) requirements and to support new challenging applications in the Radio Access Network (RAN), we need to develop new algorithms and tools for efficient resource management. In this dissertation, resource management in the RAN is considered in three distinct areas.

In Chapter 2, we provide an introduction to the key concepts, which establish the technological context of the following chapters. The first part of this dissertation focuses on serving traffic with diverse requirements in the context of 5G networks. In 5G, RAN slicing has been introduced, to support services with diverse QoS requirements in the same network infrastructure. Moreover, RAN slicing allows the Mobile Network Operators (MNOs) to configure customer-specific slices. In Chapter 3, we assess RAN slicing in terms of the traffic handling capacity for an Industry 4.0-inspired scenario. For the assessment, we compare a network with isolated slices and a non-sliced network. Extensive simulations show that the non-sliced network can serve more traffic than the sliced network while satisfying the same class-specific QoS requirements. Considering that RAN slicing will be adopted by the MNOs, this result highlights that additional radio resource management mechanisms are needed when RAN slicing is configured. To that end, in Chapter 4 we evaluate RAN slicing in combination with allowing slices to use idle resources of other slices, in a realistic smart city environment. The results show that idle resource sharing significantly improves the traffic performance. However, it is not until RAN slicing is further combined with other technology features, i.e. flexible numerology and mini-slots that it provides better traffic performance than non-sliced networks.

The second part of this dissertation focuses on the application of collaborative learning, and more specifically on Federated Learning (FL) in resource-constrained wireless networks. In Chapter 5 we characterise agents by their importance in the learning process and the resource efficiency of their wireless channel. Then, we provide a general agent selection framework to indicate which agents should participate in the learning process. Extensive simulations in various scenarios verify the potential of the proposed framework. Additionally, it is revealed that in scenarios where agents have small data sets or the latency requirement is stringent, it is more beneficial to perform pure learning-based agent selection. In Chapter 6 we extend the previously proposed framework to perform joint agent selection and resource allocation. We describe the problem in resource-constrained vehicular wireless networks with Multi-User Multiple Input Multiple Output (MU-MIMO) capable base stations. To approximate the optimal solution of the problem, we propose the Vehicle-Beam-Iterative (VBI) algorithm. Then, we evaluate the VBI algorithm in scenarios related to vehicular communications. The results show that in scenarios where the vehicles have the same data set sizes, the application-specific accuracy targets are achieved faster than in scenarios where the data set sizes are different. Additionally, it is shown that MU-MIMO improves the convergence time of the global FL model.

In the third part of this dissertation, the deployment of a drone swarm is addressed. In Chapter 7 we study the link density is Random Geometric Graphs (RGGs). Specifically, we very accurately approximate the link density in any two- and three-dimensional rectangular spaces with the Fréchet distribution. Then, we express the minimum number of nodes needed to ensure network connectivity in terms of the link density. Finally, we model a drone swarm with a RGG and we estimate the required size of the swarm such that communication among all drones can be ensured.

The conclusions of this dissertation and the directions for future work are presented in Chapter 8.

SAMENVATTING

In de laatste paar jaren is het aantal mobiele communicatie-apparaten en daarmee de hoeveelheid verkeer in draadloze telecommunicatienetwerken toegenomen. Daarnaast worden nieuwe toepassingen met strikte vereisten verwacht, zoals collaborative learning en het uitbreiden van de netwerkdekking met behulp van *drones*. Om te voldoen aan de kwaliteitseisen *(Quality of Service (QoS))* van het verkeer en om nieuwe uitdagende applicaties in het mobiele toegangsnetwerk *(Radio Access Network (RAN))* te ondersteunen, moeten er nieuwe algoritmen en hulpmiddelen ontwikkeld worden voor efficiënt gebruik van de beschikbare middelen. In deze thesis wordt het gebruik van de beschikbare middelen bekeken in drie verschillende gebieden.

In Hoofdstuk 2, geven we een introductie over de centrale concepten waarop de technologische context van de volgende hoofdstukken gebouwd wordt. Het eerste deel van deze thesis focust zich op bedienen van verkeer met diverse vereisten in de context van 5G netwerken. In 5G is *RAN slicing* geïntroduceerd om services met verschillende QoS vereisten in hetzelfde netwerk te ondersteunen. Bovendien, laat *RAN slicing* de *Mobile Network Operators (MNOs)* toe om klant specifieke slices te configureren. In Hoofdstuk 3, evalueren we *RAN slicing* aan de hand van de netwerk verkeer afhandelingscapaciteiten in een scenario geïnspireerd op *Industry-4.0*. In deze evaluatie vergelijken we een netwerk met geïsoleerde *slices* met een netwerk zonder sclicing. Uitgebreide simulaties laten zien dat het netwerk zonder *slices* meer verkeer kan bedienen dan het netwerk met *slices*, terwijl aan dezelfde klasse specifieke QoS vereisten voldaan wordt. Ervan uitgaande dat *RAN slicing* wordt uitgerold door de MNOs, laat dit resultaat zien dat er meer mechanismen nodig zijn om de beschikbare middelen te verdelen bij de configuratie van *RAN slicing*. Om dat te bereiken, evalueren we in Hoofdstuk 4 *RAN slicing* in combinatie het gebruiken van ongebruikte middelen van andere *slices*, in een realistisch *smart city* scenario. De resultaten laten zien dat het delen van middelen tijdens inactiviteit een significante verbetering geeft in de geleverde dienst. Echter, het is niet totdat *RAN slicing* ook gecombineerd wordt met andere technieken, i.e. een flexibele numerologie en het gebruik van mini-tijdsloten, dat het beter presteert dan netwerken zonder *slices*.

Het tweede deel van deze thesis focust zich op het toepassen van *collaborative learning* en in het specifiek op *Federated Learning (FL)* in draadloze netwerken met gelimiteerde middelen. In Hoofdstuk 5 karakteriseren we agenten op basis van hun belangrijkheid in het leerproces en de efficiëntie van het draadloos kanaal. Vervolgens geven we een al-

gemeen agentenselectieraamwerk om te identificeren welke agenten worden betrokken in her leerproces. Uitgebreide simulaties van verschillende scenario's bevestigen de potentie van het voorgestelde raamwerk. Daarnaast ontdekken we dat in scenario's waar agenten een kleine data set hebben of dat de vertragingstijden vereisten strikt zijn, het beter is om de agenten te selecteren puur gebaseerd op hun bijdrage aan het leerproces. In Hoofdstuk 6 breiden we het voorgenoemde raamwerk uit om zowel agent selectie als ook verdeling van de middelen gecombineerd uit te voeren. We beschrijven het problem in draadloze voertuignetwerken met beperkte middelen en basisstations geschikt voor *Multi-User Multiple Input Multiple Output (MU-MIMO)*. Om de optimale oplossing van het probleem te benaderen, stellen we het *Vehicle-Beam-Iterative (VBI)* algoritme voor. Vervolgens evalueren we het VBI algoritme in scenario's gerelateerd aan communicatie tussen voertuigen. De resultaten laten zien dat in scenario's waar de voertuigen even veel data hebben, de applicatie-specifieke nauwkeurigheidsdoelen sneller bereikt worden dan wanneer de hoeveelheden data verschillend zijn. Daarnaast laten we zien dat MU-MIMO de convergentietijd van het globale FL model verbetert.

In het derde deel van deze thesis wordt het inzetten van een drone zwerm behandeld. In Hoofdstuk 7 bekijken we de lijndichtheid in willekeurige geometrische grafen *(Random Geometric Graphs (RGGs))*. In het specifiek, schatten we, met hoge nauwkeurigheid, de lijndichtheid in willekeurige twee- en driedimensionale rechthoekige ruimtes met de Fréchet-kansverdeling. Daarna drukken we de minimale hoeveelheid knopen vereist voor netwerkconnectiviteit uit in termen van de lijndichtheid. Tenslote modeleren we een drone zwerm met een RGG en schatten we de benodigde grootte van de zwerm die nodig is om communicatie tussen alle drones te garanderen. De conclusies van deze thesis en mogelijke richtingen voor toekomstig onderzoek worden gepresenteerd in Hoofdstuk 8.

Acronyms

2D	two-dimensional
3D	three-dimensional
3GPP	3rd generation partnership project
5G-PPP	5G infrastructure public private partnership
AR	augmented reality
AWGN	additive white Gaussian noise
BB	broadband
BLER	block error rate
BS	base station
BWP	bandwidth part
CBC	COIN-OR branch and cut
CNN	convolutional neural network
CPU	central processing unit
CQI	channel quality indicator
CSI-RS	channel state information reference signal
DL	downlink
EDF	earliest deadline first
EDGE	enhanced data rates for GSM evolution
eMBB	enhanced mobile broadband
ETSD	European traffic sign data set
ETSI	European telecommunications standards institute
EXP-PF	exponential proportional fair
FDD	frequency division duplexing
FL	federated learning
FLOP	floating point operation
FNBW	first-null beamwidth
FR	frequency range
FWA	fixed wireless access
GoB	grid of beams
GPRS	general packet radio service

GSM	global system for mobile communications
GT	guard time
HARQ	hybrid automatic repeat request
HPBW	half-power beamwidth
HSPA	high speed packet access
ICT	information and communication technology
IID	independent and identically distributed
INI	inter-numerology interference
IP	internet protocol
ISD	inter-site distance
KPI	key performance indicator
LC	latency-constrained
LoS	line of sight
LTE	long term evolution
M-LWDF	modified-largest weighted delay first
MAC	medium-access control
MCS	modulation and coding scheme
MIESM	mutual information effective SINR mapping
MIMO	multiple input multiple output
ML	machine learning
mMTC	massive machine-type communication
MNO	mobile network operator
MR	maximum rate
MU-MIMO	multi-user multiple input multiple output
NACK	negative acknowledgement
NLoS	non-line of sight
NMT	nordic mobile telephone
NR	new radio
OFDM	orthogonal frequency division multiplexing
OFDMA	orthogonal frequency division multiple access
OLLA	outer-loop link adaptation
OSI	open systems interconnection
PBCH	physical broadcast channel
PDCCH	physical downlink control channel
PDF	probability density function
PDSCH	physical downlink shared channel
PF	proporional fair
PHY	physical

PPP	Poisson point process
PRACH	physical random access channel
PRB	physical resource block
PUCCH	physical uplink control channel
PUSCH	physical uplink shared channel
QoS	quality of service
RAN	radio access network
ReLU	rectified linear unit
RGG	random geometric graph
RMSE	root mean square error
RRM	radio resource management
RSRP	reference signal received power
SCS	subcarrier spacing
SDMA	space division multiple access
SGD	stochastic gradient descent
SINR	signal-to-interference-plus-noise ratio
SLA	service level agreement
SM	sensor monitoring
SMS	short message service
SNR	signal-to-noise ratio
SRS	sounding reference signal
SS	special slot
SSB	synchronization signal block
SU-MIMO	single-user multiple input multiple output
TDD	time division duplexing
TO	throughput-oriented
TTI	time transmission interval
UE	user equipment
UL	uplink
UMTS	universal mobile telecommunications system
UPRA	uniform planar rectangular array
URLLC	ultra-reliable and low-latency communication
V2I	vehicle-to-infrastructure
V2V	vehicle-to-vehicle
VBI	vehicle-beam-iterative
VGG	visual geometry group
VoLTE	voice over LTE
VR	virtual reality

VS	video surveillance
W-EDF	weighted-earliest deadline first

1

Introduction

"I do not think the wireless waves I have discovered will have any practical application."

Heinrich Rudolph Hertz, ca. 1890

1.1. History of Cellular Networks

"A Dynamical Theory of the Electromagnetic Field" was published in 1865 by James Clerk Maxwell, who suggested that electromagnetic fields can propagate as waves at the speed of light [1]. Twelve years later, in 1887, Heinrich Rudolph Hertz proved the predictions of Maxwell by a series of experiments [2]. However, it was Nikola Tesla, in 1893, who paved the road for wireless communications as he was the first to demonstrate the transmission of electromagnetic energy [3]. Two years later, Guglielmo Marconi developed a wireless transmission system and performed the transmission and reception of a signal using Hertzian waves over a distance of more than a mile [4].

The adoption of radio communications started in the early 20th century with the first distress radio signal transmitted by the East Goodwin Lightship in 1899 [5] and the Detroit police used the first voice communication in 1928 [6]. However, the communication was limited in terms of resources (or channels) and confined to a certain geographical area. In 1947, the Bell Laboratories introduced the concept of cellular radio technology to enable uninterrupted communication regardless of distance [7], while the development of the first handheld analogue cell phone by Martin Cooper was in 1973 at Motorola [8]. Eventually, one of the first widespread systems, the nordic mobile telephone (NMT) system, for the first generation (1G) of cellular communications was launched in Sweden in 1981 [9].

The second generation (2G) of cellular networks, similarly to 1G networks, was initially focused on speech communication and the first 2G standard, global system for mobile communications (GSM), was developed by the European telecommunications standards institute (ETSI) and first implemented in 1991 in Finland [10]. Moreover, the short message service (SMS) was later introduced to the standard [11]. With the importance of the Internet increasing over the years, follow-up standard general packet radio service (GPRS) allowed the access to the Internet and the transmission of packet data, while the enhanced data rates for GSM evolution (EDGE) standard enhanced the attainable data rates and latency [10, 11]. The popularity of and need for data services led to the standardisation of the universal mobile telecommunications system (UMTS) by the 3rd generation partnership project (3GPP), which marked the beginning of the third generation (3G) of cellular networks with the first implementation in 2001 in Japan [10]. UMTS was upgraded with the incorporation of the high speed packet access (HSPA) technology, which further increased the data rates and spectrum efficiency and reduced the latency [12].

The fourth generation (4G) of cellular networks included the long term evolution (LTE) standard which was developed by 3GPP and first launched in 2009 in Sweden and Finland [13]. LTE was focused on mobile broadband services and compared to the 3G standards it further improved the data rates, spectrum efficiency and latency [14]. Also, LTE enabled more internet protocol (IP) services such as voice over LTE (VoLTE) and evolved to support machine-type communication [14]. In 2019, South Korea was the first country to deploy the first fifth generation (5G) network [15]. 5G continues the trend of improving mobile broadband services and also addresses services requiring low-latency and high-reliability or massive connectivity [16]. Although 5G is still in its development cycle, the sixth generation (6G) is envisioned to support more resource-demanding services and entered its early stage of development [17]. Considering that a new generation of cellular networks is developed roughly every 10 years, 6G is expected around 2030 [18].

1.2. CHALLENGES IN 5G AND BEYOND

Following the trend of previous years, the number of connected devices to cellular networks is increasing and at the end of 2023, there were about 1.6 billion 5G subscribers worldwide [19]. The increase of the number of subscribers, in combination with the development of new applications, such as augmented reality (AR), naturally increase the traffic volume of the network. Over just two years, the traffic volume has almost doubled, primarily from an increase of viewing video content, which accounts for over 60% of the total traffic [19]. Figure 1.1 shows that the 2G/3G/4G mobile traffic will peak around 2026 and start its decline, whereas the 5G mobile and fixed wireless access (FWA) traffic is projected to keep increasing over the next five years. The traffic increase is in line with the developments of applications requiring high video resolution, like virtual reality (VR) and AR [19].

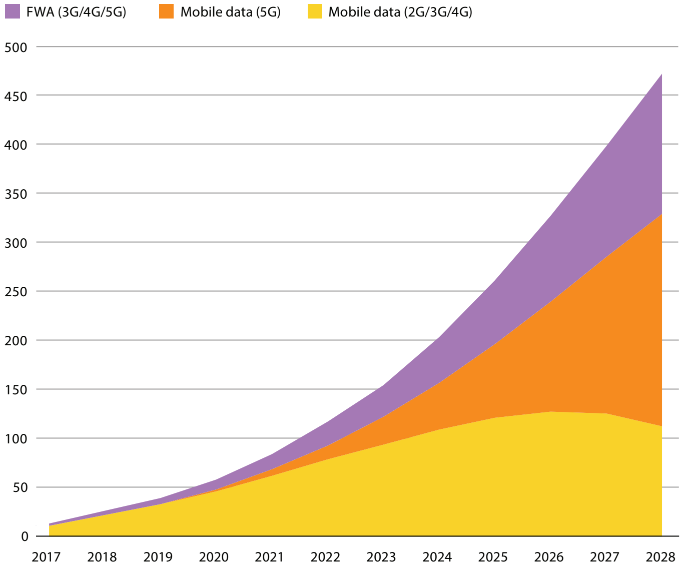


Figure 1.1: Mobile data traffic trend over the years, measured in exabytes per month, as shown in [19].

The increase of the data traffic can be addressed with more spectrum. However, the lower (sub-6 GHz) frequencies are scarce and mostly occupied by other technologies such as 2G/3G/4G, Bluetooth, Wi-Fi and radars. Therefore, 5G aims to utilise some available spectrum in the sub-6 GHz range as well as spectrum in the millimeter-wave bands, which are not over-utilised [16]. However, the millimeter-wave bands come with many challenges and require the development of new receiver and transmitter technologies. Another challenge with spectrum is that frequencies have to be split among the mobile network operators (MNOs) (or other businesses) of each country, who participate in multi-million worth auctions to acquire spectrum licences. For example, in 2020, 1.23 billion Euros were raised by the Dutch government for the allocation of the 700, 1400 and 2100 MHz frequency bands [20]. Given the limited spectrum available and the huge licensing costs, it is crucial to design networks and develop technologies and algorithms for efficient radio resource management.

Apart from addressing the challenges of increased network traffic, 5G also address the servicing of applications with diverse quality of service (QoS) requirements. Traditionally, cellular technologies were designed for the needs of consumers, for example voice and video telephony. However, 5G also addresses the needs of other markets, the so-called 5G verticals. The 5G infrastructure public private partnership (5G-PPP), which is an initiative between the European Commission and the European information and communication technology (ICT) industry, has defined ten verticals, e.g. automotive, Industry 4.0, energy and smart cities [21]. Different applications can be defined for each vertical with applications having diverse characteristics and QoS requirements.

To classify applications based on their QoS requirements, three categories are defined: *enhanced mobile broadband* (eMBB), *ultra-reliable and low-latency communication* (URLLC) and *massive machine-type communication* (mMTC) [22]. Figure 1.2 illustrates how different applications relate to the three categories. For example, self-driving cars strictly belong to the URLLC category because the response times should be very short and the related data should be reliably transmitted to ensure driving safety. Another example is AR which requires relatively high data rates for the video streaming, as well as low-latency for the reaction to motion. AR is envisioned in multiple use cases, for example for at-home gaming and for remote maintenance in factories, hence, different requirements can be derived per use case [23].

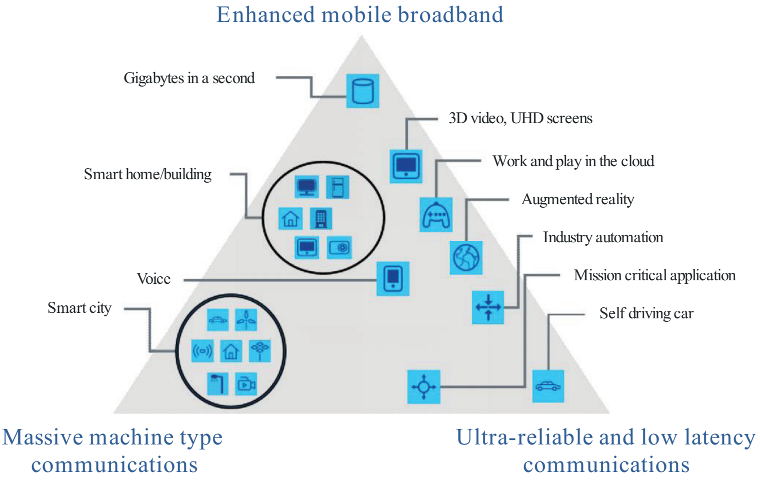

Enhanced mobile broadband
Gigabytes in a second
3D video, UHD screens
Work and play in the cloud
Smart home/building
Augmented reality
Industry automation
Voice
Mission critical application
Smart city
Self driving car
Massive machine type communications
Ultra-reliable and low latency communications

Figure 1.2: Applications relation to the three 5G categories as shown in [22].

To address the challenges of serving traffic with diverse requirements, the concept of network slicing has been introduced in 5G networks. Network slicing allows for multiple independent and logically separated networks, which can be flexibly and dynamically configured while sharing the same physical resources [24]. Each such slice can be designed to serve a specific type of traffic, for example, different slices can be configured for eMBB, URLLC and mMTC traffic. Moreover, a slice can also be configured for a specific customer. Even though network slicing makes the network easily manageable and configurable, it can be inefficient in terms of radio resource utilisation [24]. Apart from network slicing, different features have been incorporated in 5G technology to efficiently serve specific types of applications. For example, flexible numerology [25] and mini-slots [26] can provide low latency while multi-user multiple input multiple output (MU-MIMO) beamforming [27] can offer high data rates and increase capacity.

Even though 5G is still under development, 6G is envisioned to support applications with even more stringent requirements. Examples of such applications include holographic interaction, autonomous machines and precision healthcare [28]. Moreover, stand-alone ‘sub-networks’ are envisioned such as a swarm of drones extending the terrestrial network [29]. Additionally, algorithms leveraging the recent advances in machine learning (ML) are expected to be more prominent, compared to 5G, with many applications, e.g. collaborative learning, heavily relying on them [28]. Therefore, there is no doubt that efficient resource management for diverse and dynamic traffic is and will remain challenging.

1.3. Motivation and Research Questions

While multiple technologies and algorithms are available for resource management in the radio access network (RAN), new applications with stringent requirements and in challenging environments require the re-evaluation of the currently available technologies and algorithms and the design and development of new ones. In this dissertation, resource management in the RAN is considered in three distinct areas: *(i)* serving traffic with diverse requirements, *(ii)* collaborative learning and *(iii)* deployment of a drone swarm.

Network slicing has been standardised for 5G networks to allow flexible configuration of the network, based on the expected traffic. When network slicing is extended in the RAN, the radio resources need to be split amongst the configured slices. However, the resource split could lead to a lower resource efficiency compared to a non-sliced RAN that utilises advanced schedulers to deal with diverse traffic. Considering that RAN slicing will be adopted by the MNOs, what will be the impact on the radio resource efficiency? Next to slicing, features like flexible numerology and mini-slots are also developed for the handling of diverse traffic. Even though the merits of each such individual features are proven, how can those features be combined with each other and with RAN slicing to improve the traffic performance?

ML algorithms can be applied as solutions to efficiently address complex resource management problems. Traditionally, ML algorithms run on a central location and require the aggregation of data from multiple sources, e.g. user devices. However, in a wireless network, the transmission of large amounts of data may congest the network. Federated learning (FL), which is a decentralised ML technique, addresses the concerns of privacy and increased network load by combining trained-at-device models. Hence, the devices transmit the parameters of their locally trained models to the network rather than the raw training data. Considering that it is resource inefficient when all devices in the network participate in the training of the central model, what is a good device selection policy, assuming resource-constrained networks? Moreover, how does the data set sizes of the devices influence the selection of the agents and their radio resource allocation?

Another envisioned use case is the deployment of a swarm of drones to provide coverage in a disaster area or to serve incidental traffic hot spots. When a swarm of drones is deployed, the drones in the swarm are expected to communicate with each other in order to avoid collisions and exchange necessary information for collaborative tasks, e.g. optimising their (relative) positions to ensure joint coverage/capacity provisioning. Thus, any drone should be able to reach any other drone in the swarm to establish a connected network. While many studies in literature focus on deploying a swarm of drones to provide coverage and/or capacity to the RAN, the connectivity among the drones is usually ignored. Can the minimum number of drones needed for connectivity be derived, regardless of the network geometry and propagation conditions?

1.4. CONTRIBUTIONS AND OUTLINE

This dissertation is organised as follows. Chapter 2 introduces the key concepts which establish the technological context of the following chapters. Specifically, transmit antennas, physical layer configuration, wireless channels and network coverage are presented. The chapter is concluded with a high-level introduction to radio resource management methods. Chapters 3-7 are split in three different parts.

The first part comprises Chapters 3 and 4, and it is related to serving traffic with diverse requirements. In Chapter 3, a comparison between optimally configured, in terms of packet scheduler and numerology, non-sliced and sliced networks is provided. Considering Industry 4.0-inspired throughput-oriented and latency-constrained traffic, extensive simulations showed that the non-sliced network can serve more traffic than the sliced network. In Chapter 4, a broader scenario in terms of traffic characteristics and requirements is considered, related to smart cities. Extensive simulations showed that when configuring the numerology per slice, all other applied features should be taken into consideration, rather than solely the traffic type of the slice. Finally, it was concluded that RAN slicing can improve the traffic performance when it is properly combined with other features.

Chapters 5 and 6 constitute the second part of this dissertation, which is related to collaborative learning with FL. Chapter 5 provides a generic device selection framework to improve the convergence of the FL model while considering device- and network-specific constraints and requirements. The framework relies on characterising each device based on its importance in the learning process as well as its resource consumption. The selection policies as derived from the proposed framework are compared to baseline policies under different scenario configurations. Then, for each considered scenario, different conclusions are derived. Focusing on vehicular applications, Chapter 6 addresses the problem of joint vehicle selection and resource allocation in networks with MU-MIMO capable base stations (BSs). We first extended the framework proposed in Chapter 5 to describe the new problem. Then, we approximated its optimal solution with the proposed vehicle-beam-iterative (VBI) algorithm. Simulations showed that MU-MIMO capable BSs improve the convergence time of the global model. Furthermore, it was concluded that the application-specific accuracy targets are reached faster when vehicles have the same data set sizes than when they have different data set sizes.

The third part of this dissertation is related to the deployment of a drone swarm. In Chapter 7, the drone swarm is modelled as a random geometric graph (RGG). Considering a RGG with a distance-based connection function, we very accurate approximated the link density in any rectangular space with the Fréchet distribution. Then, we expressed the minimum number of drones needed to ensure swarm connectivity in terms of the link density.

A summary of the contributions of this dissertation and suggestions for future work are given in Chapter 8. Finally, a list of symbols is provided at the end of the dissertation. The global symbols relate to wireless communications and they are consistent throughout this dissertation. Then, a complimentary list of symbols is provided for each one of the three dissertation parts.

2

The Radio Access Network

In this dissertation we study the problem of resource management in the context of the RAN. This chapter provides the necessary principles related to the RAN and background on how transmissions are performed in the context of 5G networks. Moreover, the key radio resource management (RRM) methods used in this dissertation are explained.

2.1. INTRODUCTION

Cellular networks comprise the RAN and the core network. The RAN implements a radio access technology, known as new radio (NR) in 5G networks, which allows devices such as mobile phones, computers and machines, known as user equipment (UE), to wireless connect to the BSs. Hence, the BSs are responsible for the communication to/from the UEs. The BSs are also connected to the core network, via backhaul links, which provides, for example, voice and/or video calling services and access to the internet. This simplified description of a cellular network is illustrated in Figure 2.1.

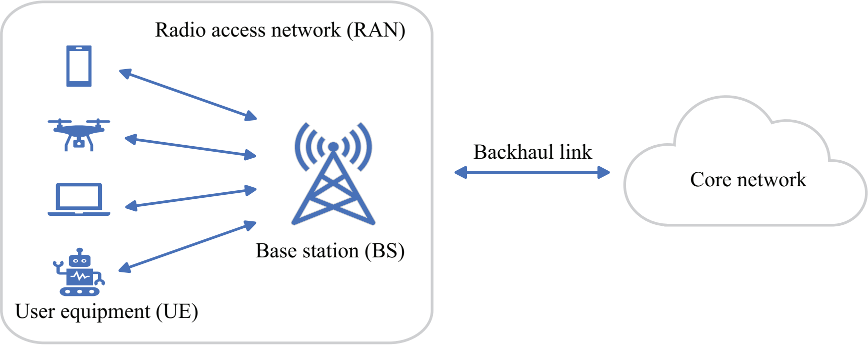


Figure 2.1: Simplified diagram of a cellular network, comprising a RAN and a network.

A RAN can be modelled by a hexagonal grid, which dictates the BS locations. The size of each hexagon is defined by the inter-site distance (ISD), which is the distance between two sites, and further depends on the environment considered, e.g. dense urban, urban or rural [30]. Each BS location is referred to as a site and typically *three-sectorised sites* are used, that is a BS has three transceiver antennas and each antenna covers 120° in the azimuth plane. The service area of each transceiver is called a *cell*. Figure 2.2 shows an example of a cellular network consisting of 19 sites and thus 57 cells and indicates the ISD. The network is configured with the objective that the cells jointly cover the whole hexagonal grid to prevent coverage gaps, which are areas that cannot be served by any transceiver. Typically, UEs are served by the transceiver, and hence the associated cell, towards which they experience the strongest radio link.

The communication between the UEs and the BSs uses the layers 1 and 2 of the open systems interconnection (OSI) model. In this dissertation, we focus on the layer-2 medium-access control (MAC) layer and on the layer-1 physical (PHY) layer. The MAC layer is, among other things, responsible for packet scheduling, error correction and performing scheduling of measurement reporting [16]. The PHY layer is responsible for coding, modulation, multi-antenna processing and the actual signal transmission on the assigned time-frequency (radio) resources [16]. In 5G, the PHY layer supports trans-

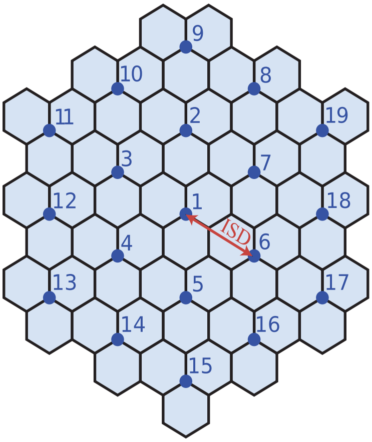


Figure 2.2: Network layout with 19 three-sectorised sites.

missions in the frequency range 0.45-6 GHz, commonly referred to as *frequency range 1* (FR1) and in the frequency range 24.25-52.6 GHz, referred to as *frequency range 2* (FR2) [31]. In this dissertation we do not consider the FR2.

2.2. TRANSMIT ANTENNAS

Antennas are an important element in cellular networks as they are responsible for the radio transmissions and they can influence the quality of the radio transmission. An *omnidirectional antenna* is an antenna which uniformly radiates the signal power P_{TX} in all directions and hence it has no directivity. The transmit power P_{TX} is commonly expressed in Watts (W) or milliwatts (mW) or decibel-milliwatts (dBm) and the maximum transmit power $P_{\text{TX,MAX}}$ typically depends on the system bandwidth B [31]. The *antenna gain*, i.e. the increase of the signal power, is expressed in decibel-isotropic (dBi) and for an omnidirectional transmit antenna, the gain is 0 dBi.

Even though omnidirectional BS antennas are sometimes considered in theoretical studies, in reality, most operationally deployed BSs use *directional antennas* that transmit a *beam* in a specific direction. In 2G, 3G and 4G networks, linear arrays with distinct antenna elements uniformly spaced at e.g. half-wavelength distance on the vertical plane are used as directional antennas. By feeding the antenna elements with the same signal, a *main lobe* is formed, which is the direction with the highest portion of radiation power [32], as shown in Figure 2.3. Additionally, one or multiple *side lobes* are also formed, radiating lower portions of power in other directions [32], as also shown in Figure 2.3. The direction of the main lobe is referred to as the *boresight direction* and the main lobe can be steered to a different direction by adjusting the relative phases of the signals fed to the antenna elements [32]. Assuming a linear array, the main lobe can only be flexibly steered in the elevation plane, because the antenna elements are only placed on the vertical plane. Furthermore, Figure 2.3 illustrates the *half-power beamwidth* (HPBW) and the *first-null beamwidth* (FNBW) of the antenna which are the angles where the power of the main lobe is reduced by half, i.e. -3 dB, and the angle between the null points of the main lobe, respectively [32].

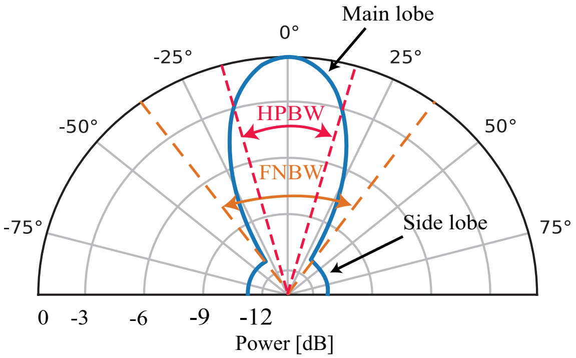


Figure 2.3: A simplified power pattern, in dB, of a directional antenna in the elevation plane.

In 3G networks, the multiple input multiple output (MIMO) transmission technology was introduced, where multiple antennas are used at the transmitter and receiver to increase the throughput. In 4G and 5G networks, MIMO capable antennas evolved to massive MIMO (mMIMO) beamforming antennas, where massive refers to the number of antenna elements. The larger the number of antenna elements used, the narrower and stronger the beam (main lobe) can be and hence, the beam gain is higher. Additionally, linear arrays evolved to planar arrays which consist of multiple antenna elements in both the horizontal and vertical planes and hence allow for three-dimensional (3D) beamforming [32]. Furthermore, multiple beams can be simultaneously formed to either serve one or multiple UEs, as it will be explained in Section 2.3.3. Therefore, mMIMO beamforming antennas can increase the user throughput as well as provide cell capacity gains.

2.3. PHYSICAL LAYER CONFIGURATION

A radio transmission is carried out in the PHY layer using the available radio resources, defined in the frequency and time domains. Specifically, the time and frequency resources are divided in a resource grid and the packet scheduler assigns which resource blocks of the grid are assigned to each transmission [33]. In this section, PHY layer aspects such as the structure and the configuration of the resource grid, multiple access and physical channels are discussed, in the context of 5G networks.

2.3.1. Frame Structure and Numerology

In the frequency domain, and similarly to the 4G RAN, NR makes use of orthogonal frequency division multiplexing (OFDM), which divides a frequency carrier into multiple narrow subcarriers. Each subcarrier can transmit data and all subcarriers are orthogonal to each other to avoid interference. Figure 2.4 depicts five OFDM subcarriers and the subcarrier spacing (SCS), where the orthogonality of the subcarriers is achieved, because at the transmission frequency of one subcarrier, the other subcarriers have zero value.

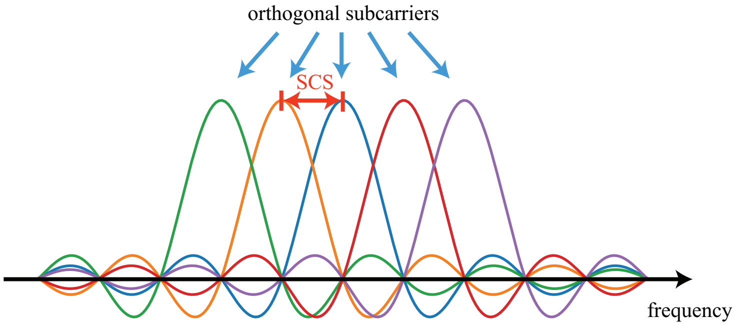


Figure 2.4: Illustration of five orthogonal OFDM subcarriers.

While in 4G networks the SCS is fixed to 15 kHz, in 5G, the SCS can be flexibly set [25] by *numerology* μ. The OFDM symbol duration, i.e. the time duration of transmitting a symbol over an OFDM subcarrier, depends on the SCS and hence, on the numerology, as shown in Table 2.1. Regardless of the numerology μ, a *slot* is defined as the time duration of 14 consecutive OFDM symbols [16] and therefore, the slot duration is also configured by the numerology μ, as shown in Table 2.1. In the time domain, and regardless of the numerology μ, the transmissions are arranged in frames with a duration of 10 ms and each frame consists of ten subframes with a duration of 1ms each [16], as shown in Figure 2.5. Thus, a subframe consists of 2^{μ} slots and hence $14 \cdot 2^{\mu}$ OFDM symbols.

A physical resource block (PRB) is the main scheduling unit, i.e. the time-frequency resources assigned to data transmissions, and constitutes $N_{\text{SUB}} = 12$ contiguous subcarriers in the frequency domain, regardless of the numerology μ. Therefore, the total number N_{PRB} of PRBs depends on the system bandwidth B. The time duration of a PRB can be flexibly defined [16] and in this dissertation we assume the PRB duration equals to the slot duration. For transmissions requiring low-latency, mini-slots can be used, which are contiguous sets of either 2, 4 or 7 OFDM symbols within a slot [26]. Figure 2.5 shows the radio resource grid in both the time and frequency domains and the PRB

Table 2.1: 5G numerologies [25].

Numerology μ	SCS (kHz)	OFDM Symbol Duration (μs)	Slot Duration (ms)	Frequency Range
0	15	71.35	1	FR1
1	30	35.68	0.5	FR1
2	60	17.84	0.25	FR1 and FR2
3	120	8.92	0.125	FR2
4	240	4.46	0.0625	FR2

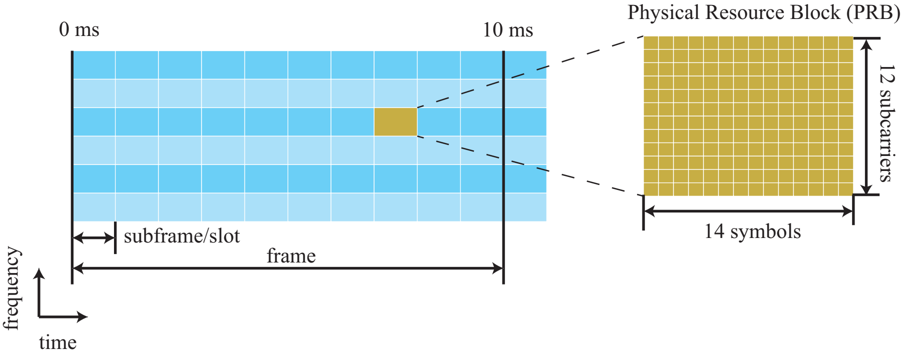


Figure 2.5: Illustration of the radio resource grid and the physical resource block (PRB) structure for numerology $\mu = 0$.

definition for the case of numerology $\mu = 0$. Therefore, the PRB/slot duration is equal to the subframe duration, i.e. 1 ms and each PRB is 180 kHz wide.

Commonly, the configuration of the numerology μ is based on the QoS requirements as services requiring low latency and services requiring high throughput are best served with higher and lower numerologies, respectively. This is related to the intrinsic trade-off between latency and throughput when choosing a numerology μ, which can be explained as follows. The wider the SCS is, the shorter the OFDM symbol duration is, which leads to a shorter slot duration and thus a shorter transmission time and less waiting time until the beginning of the next time slot for packets in the transmission queue, which further reduces latency. Therefore, low-latency services can be supported by higher numerologies. Additionally, a wider SCS reduces the number of PRBs within a given bandwidth B, compared to lower numerologies, which can have a negative impact on the throughput as the gains obtained from frequency-domain channel-adaptive scheduling are reduced, as it will be explained in more detail in Section 2.6.2.

Apart from the QoS requirements, the choice of numerology μ is also restricted by the standard based on whether the carrier frequency is in FR1 or in FR2, as also shown in Table 2.1. For example, higher SCS is needed in FR2 due to implementation limitations [33]. Finally, for large cell sizes and harsh propagation environments, lower numerologies are more appropriate as symbols with long duration are more robust to inter-symbol interference [34].

2.3.2. Bandwidth Parts

A bandwidth part (BWP) is a part of a given carrier [35] that is configured with its own numerology μ. Thus, multiple numerologies can be configured on a single carrier to support multi-service traffic e.g. by configuring BWPs with both high and low numerologies to serve traffic requiring low latency and high throughput, respectively.

Unless appropriate measures are taken, the multiplexing of numerologies on the same carrier introduces inter-numerology interference (INI) because the subcarriers of distinct numerologies are not orthogonal to each other. INI can be eliminated by using windowing, filtering and guard bands between the numerologies. The guard band size depends on the adjacently used numerologies [36].

Another role of BWPs is to reduce the energy consumption of the UEs. Because a UE is configured to only listen at one BWP at a time, the UE needs to monitor only a part of a carrier, compared to the full carrier, and hence consumes less energy. BWP is an important feature, considering that frequency carriers in 5G can be very wide and the number of energy-constrained UEs is increasing.

2.3.3. Multiple Access

To simultaneously serve multiple UEs on the same time-frequency resources, two multiple access schemes are used in 5G, that is orthogonal frequency division multiple access (OFDMA) and space division multiple access (SDMA). Consider a network with a number of PRBs defined within a given system bandwidth B, based on the numerology μ, and with scheduling decisions taken on a per slot basis. With OFDMA, transmissions to multiple UEs can be performed by assigning different PRBs over time and frequency to each transmission. The way PRBs are assigned to each transmission is defined by the scheduler, which is described in Section 2.6.2.

SDMA is a method that exploits the spatial separation of the UEs and allows multiple transmissions over the same time-frequency resources. In 5G networks, SDMA is achieved by using mMIMO beamforming antennas, which are capable of forming multiple simultaneous beams in different directions. This technology is also known as MU-MIMO [37]. Figure 2.6 shows an example radiation pattern of a planar array that simultaneously forms 12 beams at different directions. Therefore, each beam can be assigned to a transmission of a different UE. Note that techniques like zero-forcing are applied to ensure the orthogonality of beams such that the transmissions on each beam do not interfere with each other [38]. Additionally, MU-MIMO can be combined with OFDMA such that UEs that are spatially close to each other, share the PRBs of the same beam. In this dissertation we do not consider this last configuration.

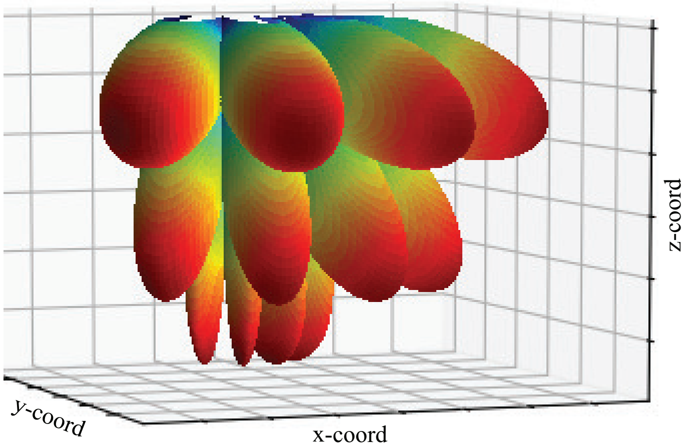


Figure 2.6: Radiation pattern of an example planar array forming 12 beams.

2.3.4. Physical Channels and Signals

The transmissions between the BSs and the UEs occur on different physical channels, depending on the transmission direction. Specifically, for the transmissions from the BS to the UE, the downlink (DL) channels are used while for the transmissions from the UE to the BS, the uplink (UL) channels are used. The transmission of the DL user data occurs on the physical downlink shared channel (PDSCH) while the transmission of the UL user data occurs on the physical uplink shared channel (PUSCH). The physical downlink control channel (PDCCH) is used by the BS to convey control information to the UE, for example, the time-frequency resources to receive its transmission on the PDSCH or the time-frequency resources reserved for its UL transmission on the PUSCH. The physical uplink control channel (PUCCH) is used by the UE to request UL resources and to report quantities such as the channel quality indicator (CQI) and the reference signal received power (RSRP), which will be explained later on. The physical broadcast channel (PBCH) is used to transmit necessary information to the UE in order to connect to and communicate with the BS. Finally, the physical random access channel (PRACH) is used for the random access procedure, which the UE initiates to establish a connection with the BS.

In 5G, the synchronisation signals and the PBCH are transmitted together in the synchronization signal block (SSB) [33]. The UEs can measure their RSRP on the SSB, which can then be reported back to the BS. Typically, RSRP measurements on the SSB are needed for coverage purposes. Moreover, the UEs can measure their RSRP on the channel state information reference signal (CSI-RS), which is a UE-specific periodic signal to support beam management [33]. Apart from measurement and reporting on the DL channel, the UE is configured to transmit a periodic sounding reference signal (SRS) for UL channel estimation at the BS, which can also be used for beam management [33]. More details on how the CSI-RS and the SRS are used for beam management are given in Section 2.6.3.

2.3.5. Duplexing

Duplexing refers to the ability of the system to handle both UL and DL communication. The transmission of UL and DL data via the related physical channels can happen in two duplexing modes. In a frequency division duplexing (FDD) mode, the UL and DL channels are defined on distinct frequency carriers and thus both UL and DL transmissions can occur simultaneously. In a time division duplexing (TDD) mode, the UL and DL channels are multiplexed in time on the same frequency carrier.

Considering that the traffic load might differ significantly between the UL and DL channels, the use of the TDD mode allows for flexible and dynamic resource allocation between the UL and DL channels which consequently leads to a higher resource efficiency compared to FDD mode [33]. Specifically, in TDD mode, network adaptation based on the traffic can in principle be performed with the ratio of resources assigned to the UL and DL channels (dynamically) adjusted to the up/downlink traffic asymmetry. Note that the same TDD configuration should apply in the whole network to avoid interference and that the regulator may impose a fixed TDD configuration to avoid interference between adjacent networks [39], which limits the gains of using TDD mode. Typically, the frequency of the UL and DL channels in the TDD configuration applies for a set period and then periodically repeats.

A drawback of using TDD mode is that a guard time (GT) is necessary when switching from the DL to the UL channels [33]. Therefore, after the slots carrying DL channels, a special slot (SS) which contains the GT is needed before the UL slots will follow. The remaining OFDM symbols within the given TDD periodicity can be assigned to the DL and UL channels based on the expected DL and UL traffic. Figure 2.7 shows an example TDD configuration, characterised by a TDD periodicity of five slots, viz. three DL, one SS and two UL slots. The SS is shown to comprise three DL, two GT and nine UL symbols. In this example, the DL channel is assigned of 66% of the resources, which makes this TDD configuration suitable for a network where the DL to UL traffic ratio is 2:1.

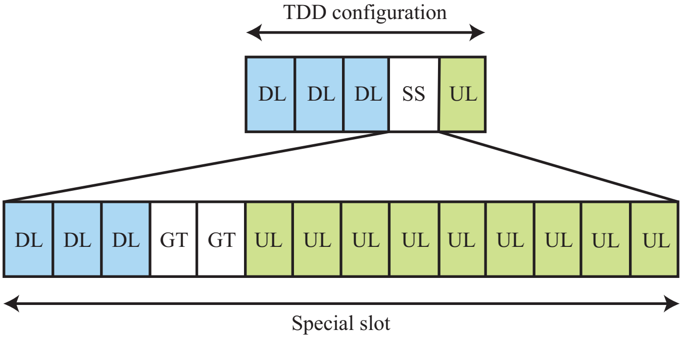


Figure 2.7: TDD configuration example with a periodicity of five slots.

2.4. Wireless Channel

This section describes the propagation of signals and introduces the key performance metrics associated with the radio transmissions.

2.4.1. Signal Propagation

Consider a signal from a transmitter i to a receiver j with power $P_{\mathrm{TX},ij}$. The received signal power $P_{\mathrm{RX},ij}$ attenuates compared to the transmit power $P_{\mathrm{TX},ij}$ as a result of the occurred propagation phenomena, such as the path loss, the shadow fading and the multipath fading. This subsection provides a general description of the propagation phenomena, whereas each model considered in this dissertation is given in its respective chapter.

In outdoor environments, the average signal attenuation over distance is given by the *path loss*. Path loss is typically characterised by empirical models which are tuned for a specific environment e.g. factory, urban and rural [40]. Path loss can be approximately characterised by a power law model, which is used as a general model, hence the path loss $L_{\mathrm{PL},ij}$ between transmitter i and receiver j is given in dB by [40]:

$$L_{\mathrm{PL},ij} = 20\log_{10}\left(\frac{4\pi f_C d_0}{c}\right) + 10\gamma\log_{10}\left(\frac{d_{ij}}{d_0}\right), \tag{2.1}$$

where c is the speed of light, f_C is the carrier frequency, d_{ij} is the 3D distance between the transmitter and receiver, d_0 is a reference distance and γ is the path loss exponent. The range of the path loss exponent γ depends on the propagation environment with typical values [40] ranging between 2 and 6.5. Generally, the value of γ is determined by empirical measurements. The smallest value $\gamma = 2$ corresponds to the ideal case of free-space propagation.

Additionally, the actual signal attenuation varies around the path loss value due to propagation phenomena such as shadowing, caused by the various objects in the environment, e.g. buildings, that are obstructing the signal propagation. The fluctuations on the attenuation due to shadowing are called *shadow fading*. In empirical models, shadow fading is normally distributed with zero mean, in dB, and the standard deviation σ_{SF} ranges between 2 dB to 4 dB in indoor environments and between 5 dB to 12 dB in outdoor environments [40].

Furthermore, the received signal power $P_{\text{RX},ij}$ varies randomly due to signal reflections, caused by the various objects in the environment. Specifically, the received signal consists of multiple copies of the transmitted signal where each copy is received with different power, at a different time and with distinct phase and/or frequency [40]. The random variation of the received signal power $P_{\text{RX},ij}$ due to multiple copies is referred to as *multipath fading*. The movement of any object in the environment, including the transmitter and the receiver, may therefore lead to a received signal power variation, modelled by the time-varying nature of multipath fading. Typically, when there is non-line of sight (NLoS) communication between the transmitter i and the receiver j, in the linear domain, the amplitude of the received signal follows a Rayleigh distribution and hence the received power $P_{\text{RX},ij}$ is exponentially distributed with mean 1. However, when there is a line of sight (LoS) component (or signal copy), the amplitude of the received signal follows a Rice distribution with parameter κ, which defines the ratio between the LoS and NLoS components. For $\kappa = 0$, the Rayleigh distribution is induced while $\kappa \to \infty$ implies communication with a single LoS component.

2.4.2. CHANNEL QUALITY AND CAPACITY

The impact of the wireless channel on the signal transmission between transmitter i and receiver j is reflected by the received signal power $P_{\text{RX},ij}$, which is given in dB by

$$P_{\text{RX},ij} = P_{\text{TX},ij} + G_i + G_j - L_{\text{PL},ij} - L_{\text{SH},ij} - L_{\text{MP},ij}, \tag{2.2}$$

where $L_{\text{SH},ij}$ and $L_{\text{MP},ij}$ are the experienced shadow fading and multipath fading between transmitter i and receiver j, respectively, and G_i and G_j are the antenna gains. Depending on the scenario under investigation, other losses might also be relevant, for example, penetration loss and cable loss.

Often, the quality of a transmission from transmitter i to receiver j is expressed by the *signal-to-noise ratio* (SNR), in dB, by:

$$\Gamma_{ij} = P_{\text{RX},ij} - P_{\text{NOISE}} - NF_j, \tag{2.3}$$

where $P_{\text{NOISE}} = N_0 + 10\log_{10}(B)$ denotes the thermal noise power in a bandwidth B, with the noise power spectral density $N_0 = -174$ dBm/Hz and NF_j denotes the noise figure, in decibels (dB), at receiver j.

When multiple transmissions occur on the same time-frequency resources, they interfere with each other. Specifically, the intended receiver j also receives power from transmissions intended for other receivers. Thus, the *signal-to-interference-plus-noise ratio* (SINR) is introduced to measure the transmission quality under interference and the SINR is given in dB by:

$$\hat{\Gamma}_{ij} = P_{\mathrm{RX},ij} - P_{\mathrm{NOISE}} - NF_j - \sum_{k \in \mathcal{M} \setminus i} P_{\mathrm{RX},kj}, \tag{2.4}$$

where $P_{\mathrm{RX},kj}$ denotes the power at receiver j from each interfering transmitter k and $\mathcal{M}$ is the set of all transmitters in the network. We refer to *inter-cell interference*, when interference is experienced by transmissions in other cells, whereas *intra-cell interference* relates to interfering transmissions within the same cell. Thus, intra-cell interference is avoided with the allocation of orthogonal resources to each transmission, which is achieved using the multiple access methods presented in Section 2.3.3.

The maximum bit rate that can be achieved over a wireless channel, is given by the *channel capacity*. Assuming a received power $P_{\mathrm{RX},ij}$ over an additive white Gaussian noise (AWGN) channel with noise power P_{NOISE} and bandwidth B, the channel capacity C_{ij} in bits per second (bps) is given by the Shannon-Hartley equation [40]:

$$C_{ij} = B \log_2\left(1 + \Gamma_{ij}\right), \tag{2.5}$$

where the SNR Γ_{ij} is in linear units. While (2.5) gives the theoretical upper bound for the bit rate, in reality, the channel capacity depends on the interference and the used modulation and coding scheme (MCS), which is explained in Section 2.6.4, and it is upper bounded by the per technology achievable peak rate.

2.5. Network Coverage

As previously mentioned, cellular networks can be deployed in hexagonal grids with each cell representing the area served by a transceiver. When designing a network, a coverage analysis should be carried out to verify that given the configuration, e.g. the ISD, the maximum transmit power and the antenna downtilt, the network provides coverage at each location. We consider a location as covered when the SSB-based RSRP at that location exceeds the -120 dBm threshold. The coverage analysis reveals whether or not a UE can always connect to a BS, regardless of its location. We consider a UE out of coverage when it cannot connect to any BS due to a low RSRP.

Assuming an SSB transmission from BS i, we measure the SSB-based RSRP, in dBm, at every location j with

$$RSRP_{ij} = P_{\mathrm{SSB}} + G_i + G_j - L_{\mathrm{PL},ij} - L_{\mathrm{SH},ij} - L_{\mathrm{MP},ij}, \tag{2.6}$$

where P_{SSB} denotes the transmit power per resource element, given in dBm by [41]

$$P_{\text{SSB}} = P_{\text{TX,MAX}} - 10\log_{10}(N_{\text{PRB}} \times N_{\text{SUB}}). \tag{2.7}$$

2.6. Radio Resource Management

2.6.1. RAN Slicing

In 5G standardisation, the concept of network slicing has been introduced which enables the co-existence of services with different QoS requirements over the same physical infrastructure. Multiple independent slices can be configured, over the RAN, transport and core network, which can be seen as virtual networks, as illustrated in Figure 2.8. Focusing on the RAN, each slice is designed to support traffic with similar QoS requirements and therefore an optimal configuration, e.g. the numerology and the packet scheduler, can be chosen per slice. Each RAN slice can be dedicated to a particular service category, e.g. eMBB, URLLC and mMTC or to a vertical customer with specific QoS requirements [42].

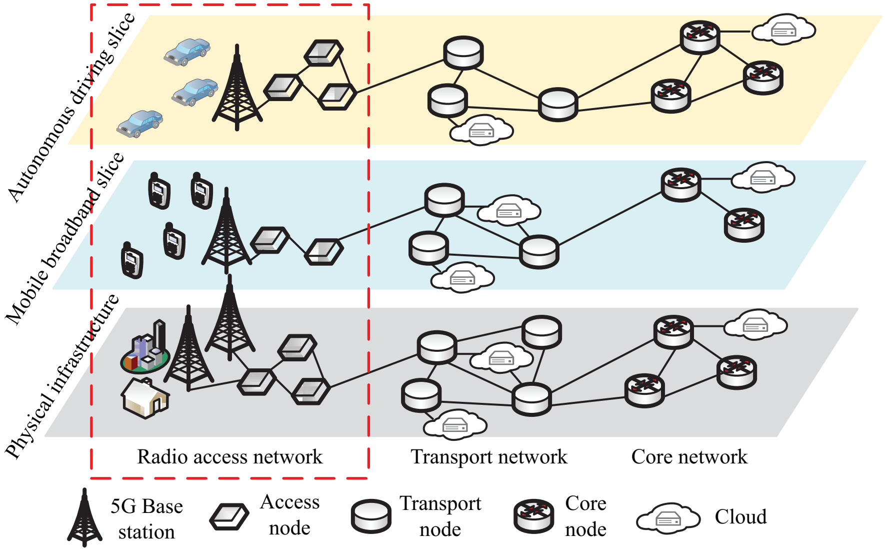


Figure 2.8: Two end-to-end slices are defined: one related to autonomous driving traffic and one related to mobile broadband traffic. Both slices share the same physical infrastructure.

RAN slicing offers an easy way to manage the network and can provide gains from serving traffic with the optimal configuration. However, these gains can be significantly limited or even become losses by a poor radio resource assignment to slices, because all slices share the given total set of radio resources. Assigning the radio resources to each slice in such a way that the QoS requirements for each slice are guaranteed is a non-trivial task as the traffic can be very dynamic and the QoS requirements may be very demanding. Additionally, splitting the radio resources among the slices leads to trunking losses compared to non-sliced networks [43]. A potential way to overcome the losses from statically assigning the radio resources to each slice, is to allow slices to use idle resources from other slices, at the cost of increased interference, or to assign the resources to the slices dynamically, at the cost of higher complexity.

2.6.2. PACKET SCHEDULING

The packet scheduler is responsible of assigning the time-frequency resources to the active QoS flows in the network (or slice). Scheduling can be performed in the time domain, i.e. for a given slot, all PRBs are assigned to a single QoS flow, or in both the time and frequency domains, i.e. for a given slot, different QoS flows are assigned per PRB. When decisions are taken on a per-PRB level, frequency diversity gains are achieved, because the scheduler exploits the fact that some flows perform better than others on particular PRBs due to frequency-selective fading and interference. For both types of schedulers, the resource assignment is performed at every slot and it is based on a scheduling metric, which is calculated for every active QoS flow at every slot (and at every PRB) [43]. The scheduling metric depends on the scheduler and it could be a function incorporating e.g. throughput and latency aspects. Then, the time-frequency resources are assigned to the QoS flow that has the highest scheduling metric value.

While a regular scheduling slot comprises 14 OFDM symbols, a mini-slot consist of 2, 4 or 7 contiguous OFDM symbols within a given slot, as introduced in Section 2.3.1. The scheduler can decide to assign to a QoS flow a mini-slot, instead of a regular slot, using different approaches [44]. The selected mini-slot size depends on the transport block size, defined by the data size, and the channel quality. Thus, the transmission of a small transport block may be shorter than a full slot duration and thus lead to latency improvements and lower inter-cell interference. Also, because of the shorter transmission times, multiple transport blocks can be transmitted in distinct mini-slots within a single slot, rather than wasting otherwise unused symbols, which enhances resource efficiency and consequently, traffic handling capacity and service performance [34]. Furthermore, a transport block can start its transmission at any time within a slot, rather than needing to wait for the next slot to start, which improves the latency.

Massive MIMO antennas are capable of simultaneously forming multiple beams and the scheduler can decide whether to serve QoS flows from one or multiple UEs on the formed beams, at a given scheduling slot. When scheduling one UE, i.e. single-user multiple input multiple output (SU-MIMO), the same signal is transmitted in different beams (transmit diversity) to increase the SINR of the transmission or distinct signals are transmitted in multiple beams (spatial multiplexing) to enhance the bit rate [37]. As previously mentioned, with MU-MIMO, different beams serve multiple UEs, in the same time-frequency resources to enhance the cell capacity.

2.6.3. BEAM MANAGEMENT

Recall from Section 2.2 that massive MIMO antennas can form one or more beams at a specific direction. The goal of beam management is to establish and maintain a *beam pair*, i.e. a beam direction at the BS and a beam direction at the UE. Then, the beam pair is used for both UL and DL transmissions [33]. Figure 2.9 illustrates such a beam pair. The exact beam directions and shapes depend on the antenna arrays and the applied precoder, which is discussed below, and the beam pair may not be perfectly aligned, as also illustrated in Figure 2.9. After a beam pair is established, beam refinement is regularly performed, due to environmental changes, to refine or reevaluate the selected beam. In this dissertation, perfect beam alignment is considered and beam refinement is not addressed.

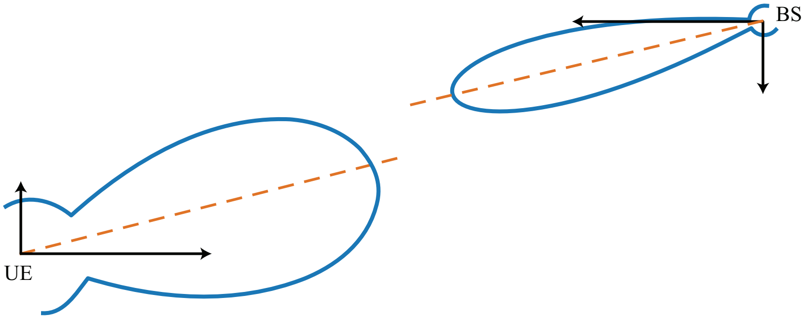


Figure 2.9: A beam pair between a BS and a UE, in the azimuth plane.

A precoder is used to indicate the phase shift and relative amplitudes at each antenna element to form the beam at specific directions [33]. To determine the applied precoder at the BS for a DL signal, two modes can be used: the grid of beams (GoB) mode and the SRS-based mode [33]. With a GoB, the antenna at the BS has a pre-defined set of beams in the two-dimensional (2D) or 3D space and the UE reports to the BS at which beam, i.e. with which precoder, it receives the highest RSRP on the CSI-RS. Another option is for the UEs to report multiple beams with their respective strengths, at the cost of consuming more UL resources for the reporting. With SRS-based beamforming, the UE transmits the SRS signal, from which the BS can estimate the channel and derive the best precoder. Even though SRS-based beamforming allows for a potentially optimal precoder, more resources are needed in the UL for the transmission of the SRS signal.

2.6.4. Adaptive Modulation and Coding

For either an UL or DL transmission, the block error rate (BLER), i.e. the ratio of the number of erroneously received data blocks to the total number of transmitted data blocks, should not exceed a threshold, which is defined based on the QoS requirements of the flow. Typically the BLER threshold is set to 10% for eMBB services and to 0.001% for URLLC services [45, 46]. To ensure that the BLER threshold is not exceeded, the BS indicates the highest attainable MCS that should be used for the transmission. The MCS is chosen based on MCS-specific BLER-vs-SINR curves where the SINR used is an estimation of the channel quality.

The channel quality estimation for the DL channel is derived based on periodic subband or wideband CQIs, reported by the UE. For the UL channel, the channel quality estimation is derived at the BS using a periodic SRS. The experienced SINR of a transmission will be different from the estimated value underlying the choice of MCS, because of imperfections in channel quality reporting, e.g. due to inherent feedback delays. Because the experienced SINR may be worse than the estimated one, the experienced BLER increases if the MCS is not adapted accordingly. Hence, an outer-loop link adaptation (OLLA) scheme can be used to modify the mapping of the estimated SNR to an MCS.

I

Serving Traffic With Diverse Requirements

3

To Slice or not to Slice?

Network slicing is an enabling feature for the effective QoS provisioning to multiple service classes with distinct performance requirements. When applied in a RAN, a class-specific slice is assigned a set of radio resources and can furthermore be optimally configured in terms of the applied numerology and packet scheduler. As both the optimal numerology and the most suitable packet scheduler may be different for e.g. a class of latency-constrained (LC) and a class of throughput-oriented (TO) services, the potential of slicing is clear. However, the inherent trunking loss incurred when applying slicing with dedicated resources provides an argument against such slicing. In this chapter we demonstrate that the performance and traffic handling capacity in an optimally configured non-sliced scenario may exceed that attained when using segregated individually optimised slices. To that end, we use simulations to assess the best-performing numerology and packet scheduler for a sliced scenario with LC and TO services. We then compare the thus optimised sliced scenario with an optimal non-sliced scenario and show that the non-sliced scenario can serve about 20% more traffic than the sliced scenario while satisfying the same class-specific QoS requirements.

This chapter is based on a published paper [43].

3.1. Introduction

5G networks are designed to support new services with diverse characteristics and requirements. Specifically, the new services are categorised in three groups: eMBB, URLLC and mMTC [42]. In the RAN, the choice of numerology and packet scheduler has an impact on the QoS for each service.

In Section 2.3.1, the concept of *flexible numerology* was introduced, where the SCS of the OFDM symbols can be flexibly configured [25] to 15, 30, 60, 120 or 240 kHz. Flexible numerology allows for a shorter OFDM symbol time and thus for shorter slots at the cost of a lower number of PRBs for a given carrier bandwidth, which may come at the cost of reduced throughput gains from frequency-domain packet scheduling, as explained in more detail in Section 2.3.1.

The role of the packet scheduler is to assign the available radio resources to the active QoS flows in the network, as explained in Section 2.6.2. Considering the diverse requirements across services, it is not trivial to accommodate all services with a single scheduling rule. However, scheduling rules have been designed for 4G networks that distinguish between real-time and non real-time traffic [47], which in combination with a suitably configured numerology, as introduced in 5G, can potentially offer even higher throughput and lower latency. Additionally, the concept of mini-slots has been introduced in 5G to enhance support of URLLC transmissions.

A novel concept introduced in 5G networks that is designed in specific support of services with different requirements in the same network infrastructure, is network slicing. With RAN slicing, which was introduced in Section 2.6.1, each slice can use the packet scheduler and numerology that best serves the intended traffic. However, assigning the radio resources to each slice in such a way that the QoS requirements for each slice are guaranteed is a challenging task.

There is a significant amount of work in literature addressing the resource assignment problem in sliced networks. Chang and Nikaein [48] proposes a two-level scheduler with resource virtualisation to perform inter- and intra-slice resource allocation. Khatibi and Jano [49] proposed an ML-based method for assigning radio resources to the slices based on traffic prediction. Other publications demonstrate how services with different requirements can be served in a non-sliced network. For example, Pedersen et al. [50] used punctured scheduling to multiplex low-latency and mobile broadband traffic. Furthermore, Akhtar and Arslan [51] addressed scheduling in multi-numerology networks.

Considering the variety of methods available in literature that address the problem of QoS provisioning, there is no clear indication about the circumstances under which RAN slicing provides the most efficient operation. While sliced networks offer the possibility of an optimal configuration per service in terms of packet scheduler and numerology, non-sliced networks enable full flexibility in resource sharing which leads to maximal trunking gains. The purpose of this chapter is to provide new insights into the relative merits of slice-optimised numerologies and scheduling on the one hand, and the trunking gains in non-sliced scenarios on the other hand, by comparing an optimised sliced with an optimised non-sliced scenario.

The remainder of the chapter is organised as follows. In Section 3.2 different packet schedulers are presented. Modelling aspects and traffic characteristics are presented in Section 3.3. The simulation results are analysed in Section 3.4. Finally, conclusions and recommendations for future work are given in Section 3.5.

3.2. PACKET SCHEDULING

Assuming time and frequency domain scheduling, the packet scheduler S determines at every slot t and for every PRB f, which of the active QoS flows will be served based on the metric $Q_{S,n}(t,f)$, calculated for every QoS flow n. In general, for a network (or slice) with N flows, at a given slot t, with a queue of packets maintained for each flow, the deployed scheduler S assigns PRB f to QoS flow n^* which has the highest $Q_{S,n}(t,f)$ value:

$$n^* = \underset{1\leq n\leq N}{\operatorname{argmax}}\, Q_{S,n}(t,f).$$

If multiple flows have the same $Q_{S,n}(t,f)$ value, which is also the highest value, the scheduler S assigns PRB f to one of those flows based on random selection. Moreover, the scheduler S may initially assign more PRBs than needed to a flow. Hence, a second step is performed where for each flow, the number of packets that can be transmitted on the assigned PRBs is calculated. If more PRBs are assigned to a flow than necessary, the scheduler S will assign the unused PRBs to other flows. Specifically, each unused PRB is assigned to the flow with the next highest $Q_{S,n}(t,f)$ value. This process repeats until all PRBs are utilised or there are no more packets. Finally, the packet scheduler S can decide to not serve (and hence drop) a head-of-line packet if it determines that it cannot be delivered within an imposed latency budget.

For our analysis, we consider a range of packet schedulers. Among these, the *maximum rate* (MR) scheduler aims to maximise the system throughput as it considers the instantaneously attainable bit rate $R_n(t,f)$ at slot t and PRB f of QoS flow n:

$$Q_{\mathrm{MR},n}(t,f) = R_n(t,f),$$

The *proporional fair* (PF) scheduler also aims to provide high system throughput, with a scheduling metric given by:

$$Q_{\mathrm{PF},n}(t,f) = \frac{R_n(t,f)}{\overline{R}_n(t-1)},$$

where $\overline{R}_n(t) = (1 - \frac{1}{t_c})\overline{R}_n(t-1) + \frac{1}{t_c} R_n(t)$ is the exponentially smoothed experienced bit rate of flow n up to and including slot t and t_c is the smoothing parameter [52]. In contrast to the MR scheduler, the PF scheduler provides a more fair resource distribution among the flows as the selection of flows is also based on the flows' experienced bit rate, in the sense that flows with relatively low experienced bit rates have a higher likelihood of being scheduled. The smoothing parameter t_c is effectively setting the trade-off between resource fairness and system throughput. For a very high value of t_c, the PF scheduler behaves similarly to the MR scheduler [52]. Both the MR and PF schedulers are only considering the channel quality of each flow (in terms of the attainable bit rates) and hence are latency-oblivious, which makes them unsuitable to serve TO traffic.

The *earliest deadline first* (EDF) [47] and the *weighted-earliest deadline first* (W-EDF) [53] schedulers explicitly aim to deliver packets within their latency budgets by considering the remaining time until the expiry of the imposed deadline:

$$
\begin{aligned}
Q_{\text{EDF},n}(t) &= \frac{1}{\tau_n - W_n(t)}, \\
Q_{\text{W-EDF},n}(t) &= \frac{W_n(t)}{\tau_n - W_n(t)},
\end{aligned}
$$

where τ_n denotes the latency budget of flow n and $W_n(t)$ denotes the head-of-line packet latency experienced up to slot t for flow n. The difference between the two schedulers is that the W-EDF scheduler uses the head-of-line packet latency $W_n(t)$ as a weight. Both schedulers are purely latency-based and hence are appropriate for serving LC traffic. However, that these schedulers have no channel-adaptive component and may consequently be rather resource-inefficient.

The *modified-largest weighted delay first* (M-LWDF), the *exponential proportional fair* (EXP-PF), the *Log-Rule* and the *EXP-Rule* schedulers [47] are based on the PF scheduler but aim to serve both LC and TO flows, featuring both channel-adaptive and latency-oriented aspects. Specifically, LC flows are served with a weighted version of metric $Q_{\text{PF},n}(t,f)$ and TO flows are served with metric $Q_{\text{PF},n}(t,f)$:

$$
Q_{S,n}(t,f) = \begin{cases} \phi_S(W_n(t))Q_{\text{PF},n}(t,f), & n \in \text{LC}, \\ Q_{\text{PF},n}(t,f), & n \in \text{TO}, \end{cases}
$$

where $\phi_S(W_n(t))$ is the weight function for scheduler $S \in \{$M-LWDF, EXP-PF, Log-Rule, EXP-Rule$\}$. For the M-LWDF scheduler:

$$
\phi_{\text{M-LWDF}}(W_n(t)) = a_n W_n(t),
$$

where $a_n = -\log(\delta_n)/\tau_n$ and $\delta_n \in [0,1]$ is the maximum allowed packet drop rate for flow n. The EXP-PF scheduler tries to guarantee the packet delivery latency by using an exponential function:

$$\phi_{\text{EXP-PF}}(W_n(t)) = \exp\left(\frac{a_n W_n(t) - \overline{aW}(t)}{1+\sqrt{\overline{aW}(t)}}\right)$$

with $a_n = 10/\tau_n$ and $\overline{aW}(t) = \frac{1}{N_{\text{LC}}}\sum_{n\in\text{LC}} a_n W_n(t)$ where N_{LC} is the total number of LC flows. The Log-Rule scheduler tries to guarantee the packet delivery latency based on the logarithmic function which increases more slow than the exponential function:

$$\phi_{\text{Log-Rule}}(W_n(t)) = b_n \log(g + a_n W_n(t))$$

where $a_n = 5/0.99\tau_n$, $b_n = 1/E[Q_{\text{PF},n}(t,f)]$, $g = 1.1$ and $E[\cdot]$ denotes the expected value. Finally, the EXP-Rule scheduler combines characteristics of the EXP-PF and Log-Rule schedulers:

$$\phi_{\text{EXP-Rule}}(W_n(t)) = b_n \exp\left(\frac{a_n W_n(t)}{g+\sqrt{\frac{1}{N_{\text{LC}}}\sum_{n\in\text{LC}} W_n(t)}}\right)$$

where $a_n \in [5/0.99\tau_n, 10/0.99\tau_n]$, $b_n = 1/E[Q_{\text{PF},n}(t,f)]$ and $g = 1.1$.

3.3. MODELLING

This section describes modelling aspects such as the network layout, the propagation environment and the traffic model. We further define the used key performance indicators (KPIs).

3.3.1. SYSTEM MODEL

Although the analysis has much broader validity, an Industry 4.0-inspired use case with distinct services in a factory hall environment is considered. The modelled factory hall is of dimensions 100 m × 100 m × 10 m and an indoor BS with an omnidirectional antenna mounted at the centre of the ceiling [54]. The BS has a 2 dBi gain and a transmit power of 21 dBm. We assume DL transmissions to UEs that are randomly distributed in space, but at a fixed height of 1.5 m, and have a receiver noise figure of 5 dB.

The propagation environment of the factory is generated with the QuaDRiGa Industrial NLoS model [55]. The model includes among others distance-based path loss, shadowing and Ricean fading. Finally, the factory is assumed to be isolated from other traffic, hence there is no interference.

We assume a 20 MHz wide carrier in the 3.5 GHz band (FR1). The carrier applies TDD with a five-slot frame format comprising one UL and four DL slots. UEs report to the BS their DL channel quality through sub-band CQI reporting with a period of 5 ms. The CQI sub-band size is given in [56]. Based on the CQI reporting, the BS selects for the DL transmission the highest attainable MCS with an estimated BLER not exceeding 0.001% and 10% for LC and TO flows, respectively [45, 46]. Modulation schemes up to 64-QAM are supported. MCS-specific BLER-vs-SNR curves have been derived using the Vienna 5G Link Level Simulator [57]. Additionally, the mutual information effective SINR mapping (MIESM) method is used to map a set of PRB-specific SINRs to a single effective SINR value for the full set of PRBs [58]. An OLLA scheme is also used to modify the mapping of the SINR to an MCS [59][60]. Lastly, for the unsuccessful DL transmissions, the BS retransmits the lost transport blocks. For LC flows, the transport blocks are only retransmitted if they can still be delivered within their latency budget.

3.3.2. Traffic Model

We distinguish between persistent LC and non-persistent TO flows with traffic models inspired by the Industry 4.0 use cases 'cooperative robotic motion control' and 'remote access and maintenance', respectively [23]. We further consider that each flow targets a different device. Specifically, we assume the presence of N_{LC} persistent LC flows generating packets of size S_{LC} bytes with a fixed inter-arrival time of 3 ms. The latency budget for each LC packet is 3 ms. The non-persistent TO flows are generated according to a Poisson process with arrival rate λ_{TO} (in flows/s) and each TO flow is modelled as a deterministic file download of S_{TO} MB.

3.3.3. Key Performance Indicator Definitions

Distinct KPIs are defined for the LC and TO flows. For the TO flows the KPI of relevance is the 10th throughput percentile. The applied target level for this KPI is 10 Mbps.

For the LC flows, the KPI of relevance is the fraction of LC flows experiencing a reliability of at least 99.9%. We define reliability as the fraction of packets per LC flow that are successfully received at the targeted device within the latency budget. The LC packet latency is defined as the time between the packet arrival in the buffer at the BS and the successful packet reception at the targeted device. The processing latencies at both the BS and the device are also considered. Figure 3.1 shows the measured latency, which concerns the PHY/MAC layers in the user plane, for a case with one retransmission. Standardised parameter *K1* is signalled to the device, via the PDCCH, to indicate the time between the reception of the DL data on the PDSCH and the transmission of the hybrid automatic repeat request (HARQ) feedback on the PUCCH [35, 56]. The value of *K1* depends a.o. on the device capability and the operational numerology [61]. Additionally, standardised parameter *K3* indicates the time between the reception of the HARQ negative acknowledgement (NACK) on the PUCCH and the retransmission of the DL data on the PDSCH and its value is up to the BS implementation [62]. Finally, the pro-

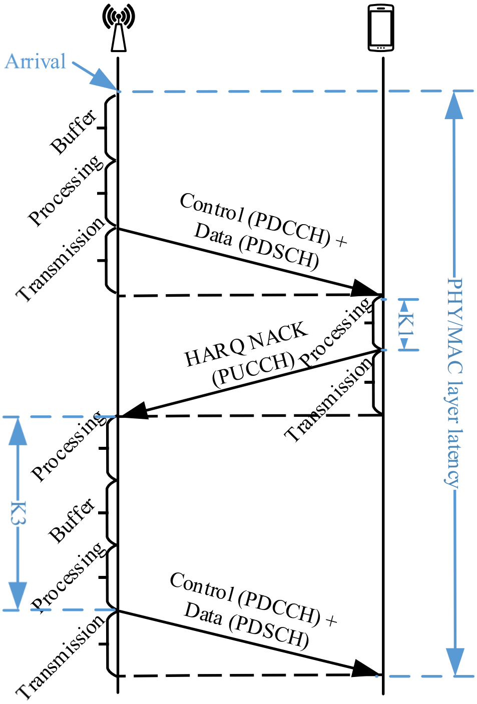


Figure 3.1: PHY/MAC layer latency for a downlink transmission with one transport block retransmission.

cessing latency at the BS is assumed to be one slot for both transmission and reception of data [63].

3.4. SIMULATION RESULTS

This section shows the impact of the packet schedulers and the numerology on the QoS *(i)* for a *sliced scenario* with distinct and isolated LC and TO slices that equally share the radio resources and *(ii)* a *non-sliced scenario* with mixed LC/TO traffic. We then compare the sliced and non-sliced scenarios based on their performance on the QoS targets. For the analysis of both scenarios, dynamic system-level simulations are performed and the results are based on multiple independent simulations with distinct random seeds. The source code generating all results is available in [64]. Considering the use of the 3.5 GHz carrier frequency, from Table 2.1, numerologies 0, 1 and 2 are used in the experiments. Additionally, the parameters related to the schedulers are set to $t_c = 10$ ms, $\tau_n = 3$ ms, $\delta_n = 10^{-5}$ for LC flows, $\delta_n = 10^{-1}$ for TO flows and $a_n = 7/0.99\tau_n$ for the EXP-Rule scheduler.

3.4.1. Sliced Scenario: Latency-Constrained Slice

In the *LC slice*, we evaluate the impact of numerology on the reliability performance when using the M-LWDF scheduler while we vary the number of persistently active flows N_{LC} and the packet size S_{LC}. Figure 3.2(a) shows the fraction of flows that meet the reliability requirement. For numerology 0, the slot duration is 1 ms and the fixed processing latency is 2 ms, according to Section 3.3.3, thus packets can spend a maximum of 1 ms in the buffer given the 3 ms latency budget. Figure 3.2(a) shows that regardless of the offered load, none of the flows can meet the imposed reliability requirement for numerology 0. For numerology 1, the fixed processing latency is reduced to 1 ms, as the slot duration is 0.5 ms, which allows packets to spend up to 2 ms in the buffer. Figure 3.2(a) shows the benefits of increasing the numerology from 0 to 1 as for some loads the required reliability can be achieved by about 95% of the flows. The fixed processing latency is further reduced to 0.5 ms for numerology 2. Moreover, packets can be retransmitted if they are not correctly received at the targeted device, which is now possible because of the shortened slot duration. Due to the retransmissions, the fraction of flows meeting the required reliability is further increased, even reaching up to 100% for cases with 30 active flows and a 100-byte packet size.

A realistic packet size S_{LC} for the considered Industry 4.0 LC use case is 150 bytes [23]. The highest number of flows N_{LC} that can be supported with $S_{\mathrm{LC}} = 150$ bytes such that the KPI target is met, is 25 flows. For this scenario, Figure 3.2(b) shows the schedulers' comparison for all three numerologies including 90% confidence intervals for the shown KPI. From Figure 3.2(b) the same observations for the impact of numerology on the KPI hold as discussed for the M-LWDF scheduler. Observe from the results that *the optimal configuration for the LC slice is the M-LWDF scheduler and numerology 2* as it provides the highest fraction of users that meet the KPI target. This is the reason the M-LWDF scheduler was used in the more detailed analysis of the numerology impact on the KPI in Figure 3.2(a).

The M-LWDF scheduler outperforms the EDF and the W-EDF schedulers as it considers both the latency budget and the instantaneous bit rate in contrast to the EDF and W-EDF schedulers that are channel-oblivious. Figure 3.2(b) also shows that the EXP-PF and EXP-Rule schedulers yield relatively poor performance as they somehow consider the normalised sum of the head-of-line latency of all LC flows. Also, their fair design limits the gains of retransmissions at numerology 2 as the retransmitted packets are closer to their deadline than packets that are transmitted for the first time. The Log-Rule scheduler performs worse than the EXP-Rule scheduler for numerology 1 due to its logarithmic component, while for numerology 2 they perform similarly as the Log-Rule scheduler benefits more from the retransmissions than the EXP-Rule scheduler.

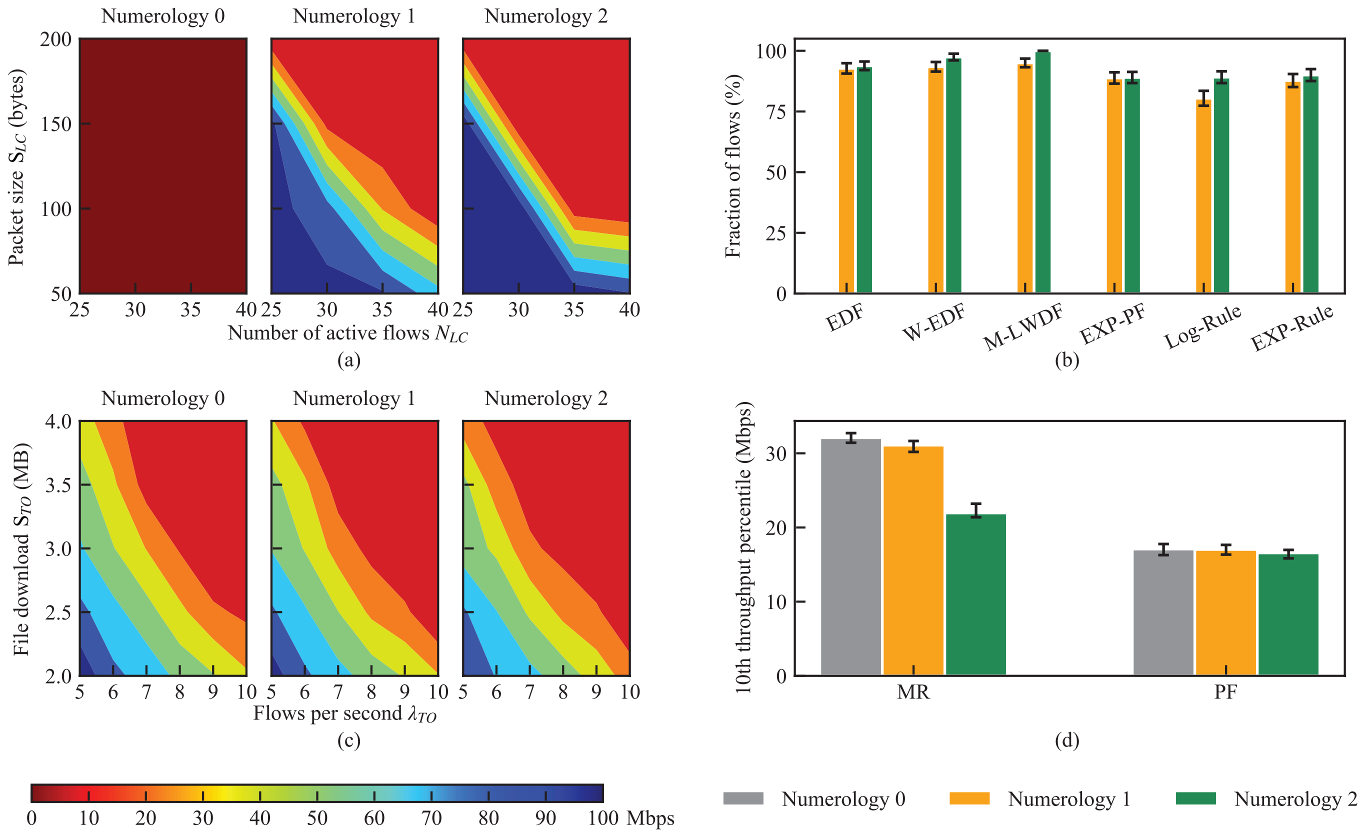


Figure 3.2: (a) Fraction of LC flows meeting the reliability requirement for a *LC slice* when the M-LWDF scheduler is used. (b) Fraction of LC flows meeting the reliability requirement for a *LC slice* with 25 LC flows and 150-byte packet sizes. (c) 10th throughput percentile in Mbps for a *TO slice* when the MR scheduler is used. (d) 10th throughput percentile for a *TO slice* with an offered traffic load of 6 TO flows per second and file sizes of 3.5 MB.

3.4.2. Sliced Scenario: Throughput-Oriented Slice

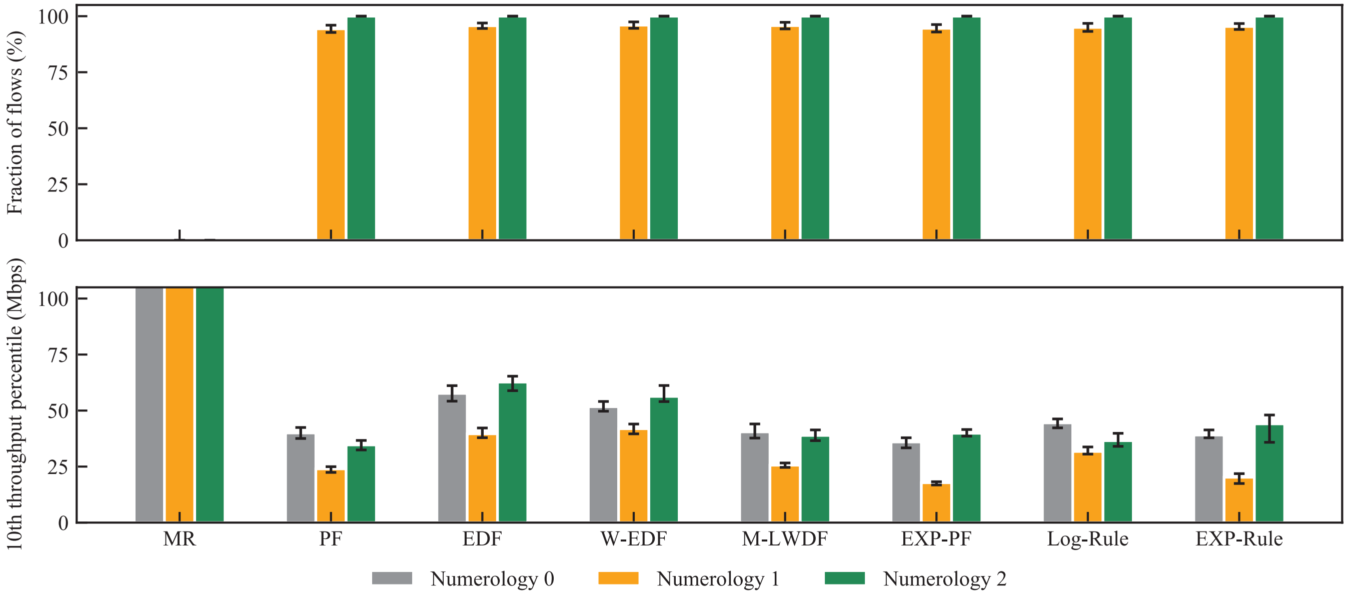
Equivalently to the LC slice, in the *TO slice*, we vary the arrival rate λ_{TO} of TO flows and the download file size S_{TO} to evaluate the impact of numerology on the 10th throughput percentile when the MR scheduler is used. Figure 3.2(c) shows the results measured in Mbps and it is observed that the throughput decreases slightly in the numerology due to reduced gains from frequency-domain channel-adaptive scheduling. For example, for a flow arrival rate of $\lambda_{\mathrm{TO}} = 8$ flows per second and $S_{\mathrm{TO}} = 3$ MB, the throughput decreases from about 13.5 Mbps to about 9.9 Mbps for numerologies 0 and 2, respectively. This effect is however rather modest in the considered scenarios due to the very good propagation conditions and the lack of interference which implies a generally very high channel quality across all PRBs, offering little potential for frequency-domain scheduling.

For a file size of 3.5 MB, which relates to the Industry 4.0 TO use case [23][65], the maximum arrival rate λ_{TO} that satisfies the KPI requirement for numerology 0 is about 6 flows per second, considering discrete integer choices. For this load scenario, Figure 3.2(d) shows the comparison between the MR and PF schedulers for all three numerologies including 90% confidence intervals for the shown KPI. Observe that the MR scheduler is performing better than the PF scheduler and thus *the optimal configuration for the TO slice is the MR scheduler with numerology 0*. The good propagation conditions in combination with the non-persistent nature of the flows allow the MR scheduler to more efficiently use the resources and make the channel more quickly available to the flows that experience weaker channels. The fairness aspect of the PF scheduler is effectively reducing all transmission rates, resulting in a reduced 10th throughput percentile. Furthermore, the fair design of the PF scheduler prevents the full exploitation of frequency diversity and thus the attained throughput gains for higher numerologies are also relatively modest.

3.4.3. Non-Sliced Scenario

For the non-sliced scenario we combine the two maximum class-specific loads found for the sliced scenario, which still satisfy the KPI targets, from the previously considered slices, i.e. 25 persistent LC flows transmitting 150-byte packets and non-persistent TO flows originating at a rate of 6 flows per second and with file size of 3.5 MB. Also, the full bandwidth is available as it does not have to be split between slices. Figure 3.3 shows the fraction of LC flows that meet the 99.9% reliability requirement and the 10th throughput percentile of TO flows for different schedulers and for all three numerologies.

Regarding the impact of numerology on the performance of LC flows, a similar observation as for the sliced scenario holds: with numerology 0 none of the flows meet the reliability requirement due to the high processing latencies regardless of the scheduler,



c

Figure 3.3: Fraction of LC flows meeting the reliability requirement and 10th throughput percentile for TO flows in a non-sliced scenario.

while with numerology 2 all flows meet the reliability requirement due to the possibility of a retransmission.

Moreover, Figure 3.3 illustrates that all schedulers designed to support LC flows (EDF, W-EDF, M-LWDF, EXP-PF, Log-Rule and EXP-Rule) perform similarly and with numerology 2 they all satisfy the KPI target. Because resources are not split over distinct slices, the above-mentioned schedulers can assign more resources to the LC flows compared to the sliced scenario which improves the performance of LC flows. For example, with the M-LWDF scheduler and numerology 2, the average packet latency is 1.12 ms and 0.92 ms for the sliced and non-sliced scenarios, respectively. In other words, packets are transmitted more quickly in the non-sliced scenario compared to the sliced scenario. Additionally, Figure 3.3 shows that the MR scheduler is outperformed by all the other schedulers from the perspective of the LC flows, regardless of the choice of numerology. This clearly reveals the unsuitability of the MR scheduler for LC flows. On the other hand, even though the PF scheduler is also not specifically designed to support LC flows, its fair design, in combination with the trunking gains inherent to the non-sliced scenario, make the PF scheduler perform similarly as those schedulers that have been specifically designed to support LC flows.

Figure 3.3 also shows the impact of the numerology on the 10th throughput percentile of the TO flows which is different compared to the sliced scenario. The impact of the numerology on the 10th throughput percentile is the net effect that a higher numerology has in terms of *(i)* an increased load, since fewer LC packets are dropped and hence the carried LC traffic load is larger; *(ii)* reduced frequency-diversity gains; and *(iii)* a reduced transfer time of LC packets. When the numerology increases from 0 to 1, there is a significant increase of carried traffic. For example, with the EDF scheduler, the percentage of packets of LC flows that are dropped by the scheduler with numerology 0 and 1 are 22.698% and 0.024%, respectively, resulting in more LC packet transmissions with numerology 1. This traffic increase is the dominant factor and causes the observed drop in the 10th throughput percentile of TO flows. When the numerology increases from 1 to 2, there is a further (yet more modest) traffic increase, as packet retransmissions of LC flows occur, and also a further reduction of the frequency-diversity gains. These effects are however relatively modest compared to the gains due to faster LC packet transmissions, which dominate in causing the observed increase in the 10th throughput percentile.

Regarding the performance of schedulers on the 10th throughput percentile of TO flows, the MR scheduler is performing significantly better than the other schedulers for all three numerologies and also in comparison with the sliced scenario. Specifically, with numerology 0, the 10th throughput percentile increases by about a factor six compared to the sliced scenario which is primarily due to trunking gains. However, this throughput increase comes at the performance cost of the LC flows, as indicated before. The PF, M-LWDF, EXP-PF, Log-Rule and EXP-Rule schedulers perform similarly as they serve TO flows with the same scheduling rule. Their small performance differences are based on the efficiency of each scheduler to serve LC flows. Also, observe that the PF scheduler performs better compared to the sliced scenario due to trunking gains. Further, Figure 3.3 shows that the EDF and W-EDF schedulers perform the best after the MR scheduler in terms of the 10th throughput percentile of TO flows. The non-persistent nature of the TO flows allows the two schedulers to efficiently transmit the packets of LC flows and make the channel more quickly available to TO flows. Considering that, only with numerology 2, the KPI for LC flows is met for all schedulers, except for the MR scheduler, *the optimal combination for the non-sliced scenario is given by the EDF scheduler in combination with numerology 2*, noting that (when disregarding the MR scheduler) the EDF scheduler provides the highest 10th throughput percentile for TO flows.

3.4.4. Comparison of Scenarios

To quantify the gains from the non-sliced scenario over the sliced scenario, we conducted additional simulations. We gradually increased the aggregate traffic in the non-sliced scenario, up to the level where the non-sliced scenario no longer outperforms the sliced scenario. This analysis revealed that a load increase of up to 20% can be handled in a non-sliced scenario. Comparing the optimal sliced and the optimal non-sliced scenarios, as defined in the previous subsections, we conclude that the non-sliced scenario performs better than the sliced scenario due to the trunking gains that are bigger than the gains from separately configuring slices.

3.5. Concluding Remarks

The need to support new services with diverse requirements has introduced the concepts of flexible numerology and network slicing in 5G networks. There is evidence in literature that the QoS requirements for particular services can be efficiently guaranteed with the use of RAN slicing or with novel packet schedulers and/or the use of flexible numerology. In this chapter, we have compared an optimal sliced scenario with isolated slices and an optimal non-sliced scenario. We showed that the trunking gains obtained from the non-sliced scenario are greater than the gains obtained by separately optimising the packet scheduler and numerology for each slice. In particular, we show that the non-sliced scenario can serve about 20% more traffic than the sliced scenario while providing the required performance to each service class.

4

Assessment of RAN Features

In the previous chapter, RAN slicing has been assessed in terms of optimally configuring the numerology and the packet scheduler and considering that the slices are isolated and therefore idle resources of a slice cannot be used by another slice. In this chapter, the problem of serving traffic with diverse characteristics and requirements is addressed by also considering (non-)pre-emptive mini-slot based scheduling and idle resource sharing between slices. Specifically, an extensive simulation-based assessment of the relative merit of the considered 5G features in the context of a smart city environment is presented. Moreover, the optimal feature combination and associated configuration which best handles the services related to the smart city environment, given their performance requirements, is derived. The obtained insights confirm the commonly argued potential of slicing, emphasising that the optimal configuration of the slice-specific numerology depends not only on the nature of the handled services but also on the selected RAN features. Among these features, non-preemptive mini-slot based scheduling and idle resource sharing reveal significant performance potential.

This chapter is based on a published paper [44].

4.1. Introduction

In smart cities, people, objects and machines are connected via wireless technologies to exchange data and collectively improve sustainability, traffic and safety, among others [66]. The cellular network will support services like VR, video surveillance and environment monitoring and thus the network should be designed to simultaneously support services of the URLLC, eMBB and mMTC categories.

The RAN should be properly configured to best support the performance requirements associated with the mix of handled services. *Flexible numerology* allows for a shorter time transmission interval (TTI) at the cost of a lower number of PRBs for a given carrier bandwidth [25], as explained in Section 2.3.1. The concept of *bandwidth part*s (BWPs), presented in Section 2.3.2, allows to configure multiple distinct numerologies on a given carrier, enabling the use of a tailored numerology for different service categories [35]. However, the split or radio resources among the BWPs leads to trunking losses which can be (partly) compensated by inter-BWPs *idle radio resource sharing*.

Furthermore, on *time division duplexing* (TDD) based carriers, the resource split between the DL and UL channels can be flexibly configured thus, allowing for a better adaptation to the actual DL and UL traffic [35], as explained in Section 2.3.5. Also, *packet scheduling* i.e. the assignment of the available radio resources to the active QoS flows in the network, can be performed using *mini-slots*, which are contiguous sets of either 2, 4 or 7 OFDM symbols within a normal time slot, enabling latency-optimised and resource-efficient scheduling for e.g. URLLC-type services [26], as explained in Section 2.6.2.

RAN *slicing* allows to independently configure each slice to serve traffic with a specific service level agreement (SLA) [24]. Thus, each slice can have its own numerology (or multiple ones) as well as packet scheduler such that it best serves the intended traffic, as it was considered in Chapter 3. Similarly to BWPs, and as it was shown in Chapter 3, there are inherent trunking losses due to the radio resource split [43] which can be (partially) compensated by inter-slice idle radio resource sharing. Moreover, BWPs in non-sliced networks allow for a similar configuration to sliced networks with the key difference that unlike slicing, BWPs cannot have their own packet scheduler. Because there are no advantages of BWPs over slicing when configuring different numerologies, we will only consider slicing in our analysis.

Several papers study the multi-numerology networks [51], the use of mini-slots [50, 67] or combinations of them [34, 68] to serve traffic with diverse QoS requirements. Most papers on RAN slicing focus on the resource assignment problem [69–71], neglecting the possibility that sliced networks may be outperformed by non-sliced networks due to the trunking losses which may be caused by slicing [43], which was shown in Chapter 3. The purpose of this chapter is to evaluate the merit of the various RAN features as well as combinations of these features in a smart city environment. Specifically, we consider flexible numerology, packet scheduling, mini-slots, RAN slicing and idle radio resource sharing. In addition, the feature combination which best handles the services from all three service categories (eMBB, URLLC, mMTC) of the smart city environment simultaneously in regards to their QoS requirements, is found.

The remainder of the chapter is organised as follows. In Section 4.2 the mini-slot and idle radio resource sharing implementations are presented. The modelling aspects, traffic characteristics and RAN configurations considered for the smart city environment are presented in Section 4.3. The simulation results are analysed in Section 4.4 and finally, the conclusions are given in Section 4.5.

4.2. FEATURE IMPLEMENTATIONS

In this chapter we consider configurations with flexible numerology, time and frequency domain packet scheduling (as considered in Chapter 3), mini-slots, RAN slicing and idle radio resource sharing. This section presents the implementation of mini-slot based scheduling and idle radio resource sharing.

4.2.1. MINI-SLOTS

Three different mini-slots based scheduling approaches are considered:

- *Default:* At the start of a regular slot, the scheduler decides which packets from the buffer are served in the upcoming slot and determines the number of symbols needed per packet.

- *Non-pre-emptive:* There are two distinct types of scheduling moments: *(i)* at the start of a regular slot, transmissions are scheduled as under the default scheme; *(ii)* during a regular slot, and given on-going transmissions, the scheduler continuously checks for new URLLC packet arrivals, which are then transmitted using a mini-slot, assuming sufficient unused resources.

- *Pre-emptive:* This approach is largely the same as the non-pre-emptive scheme. However, in case of insufficient unused resources for a new URLLC packet, the scheduler pre-empts an ongoing eMBB or mMTC transmission to schedule the URLLC packet. The transmission to be pre-empted is the one whose resources maximise the scheduling metric $Q_{S,n}$ of the new URLLC packet, as this minimises the used resources and, consequently, the degree of pre-emption of the ongoing transmissions. We assume that the pre-empted transmission has failed.

4.2.2. Idle Radio Resource Sharing

Idle radio resource sharing allows slices (denoted as source slices) to use idle resources of other slices (denoted as target slices), and thus it improves the performance and the spectral efficiency of the network. However, there are limitations when sharing the radio resources, based on the configuration of the source and target slices [72]. In particular, a source slice with a different numerology compared to the target slice can only use the resources of the target slice when the scheduling times at both slices are aligned. This drawback is mitigated with mini-slots which allow scheduling of data at any time.

4.3. Modelling

This section describes modelling aspects such as the network layout, the propagation environment and the traffic model. Furthermore, the scenario configurations and the KPI definitions are provided.

4.3.1. System Model

An urban macro-cellular environment is considered with 19 three-sectorised sites in a hexagonal layout with an inter-site distance of 500 m [54]. Three types of UEs are uniformly distributed in space, with each type of UE having one data session related to either eMBB or mMTC or URLLC traffic. Table 4.1 shows the configuration of the BSs and the UEs while the antenna diagram used at the BS is shown in [73].

Table 4.1: Configuration of the BS and the UE [54, 73].

Parameter	BS	UE
Height	25 m	1.5 m
Maximum antenna gain	17 dBi	0 dBi
Transmit power	49 dBm	23 dBm
Noise figure	3 dB	9 dB
Electrical tilt	10^o	N/A

The propagation environment has been generated with the QuadRiGa 3GPP Urban Macro-cell NLoS model, including normally distributed shadowing with zero mean and standard deviation $\sigma_{\text{SF}} = 6$ dB and Rayleigh multipath fading [55].

We assume that each cell is assigned a 15 MHz wide carrier in the 3.5 GHz band (FR1) and we consider both DL and UL transmissions [72]. In each cell, the packet scheduler determines, in both the DL and UL, which time-frequency resources are assigned to each of the active UEs, based on channel quality estimates/reports. The DL channel quality of the UEs is reported to the BS via sub-band CQIs, while the UL channel quality of the UEs is measured at the BS based on the SRS, both applying a 5 ms periodicity [56].

Based on the scheduling decisions, the BS selects for the DL and UL transmission the highest attainable MCS (up to 64-QAM) with an estimated BLER not exceeding 0.001% and 10% for URLLC and eMBB/mMTC services, respectively [45, 46]. MCS-specific BLER-vs-SINR curves have been derived using the Vienna 5G Link Level Simulator [57]. To map a set of PRB specific SINRs to a single effective SINR value for the full set of PRBs, we use the MIESM method [58]. Finally, an OLLA scheme is used to modify the mapping of the SINR to an MCS due to imperfections in channel quality reporting [60].

4.3.2. TRAFFIC MODEL

The services in smart cities are broadband access everywhere (BB), virtual reality (VR) relating to gaming sessions, video surveillance (VS) used by security officials to improve the safety of the city and sensor monitoring (SM) e.g. environmental conditions and smoke in public buildings and houses [74]. The QoS requirements for each service are shown in Table 4.2 based on [72, 74–76], with reliability defined as the fraction of packets delivered within the latency budget.

The BB and VR sessions follow a spatially uniform Poisson process with arrival rates $\lambda_{\text{BB,DL}}$ (for DL) and $\lambda_{\text{BB,UL}}$ (for UL) and λ_{VR} (for both DL and UL) sessions per second, respectively. The BB sessions are active until the file of size S_{BB} KB is fully transferred. The VR sessions are modelled to be active for a relatively short period of 5 seconds yet with an increased VR session arrival rate, to ensure both a realistic VR traffic load as well as a sufficient number of handled VR sessions to allow reliable performance assessment within a reasonable simulation time. For the other two services i.e. VS and SM, there are a total of N_i persistent sessions per service, where i denotes the service name, which are spatially uniformly distributed. The packets of each session have size $S_{i,j}$, where j denotes the channel direction, and they arrive with a constant period of $T_{i,j}$ seconds. For the persistent flows, the arrival time of the first packet is randomly chosen at $[0, T_{i,j}]$. Table 4.2 shows all of the modelling parameters based on [65, 72, 74, 77–79].

4.3.3. SCENARIO CONFIGURATIONS

Each RAN configuration consists of a number of features i.e. numerology, scheduler, mini-slots, slicing and resource sharing and the TDD configuration, which is derived based on the expected DL to UL traffic ratio network-wide. Based on the previously discussed traffic model, the TDD configuration consists of two DL slots, followed by a special slot and then by two UL slots, thus considering a TDD periodicity of five slots. The special slot consists of ten DL symbols, followed by two symbols for the GT and then

Service	Service category	Channel	QoS requirements	Session arrival	Packet arrival	File/Packet size (KB)
Broadband access everywhere (BB)	eMBB	DL and UL	50 Mbps (DL) and 10 Mbps (UL)	Non-persistent with $\lambda_{\mathrm{BB,DL}} = 100$ sessions/s and $\lambda_{\mathrm{BB,UL}} = 40$ sessions/s	Full Buffer	$S_{\mathrm{BB}} = 2000$ (both DL and UL)
Virtual reality (VR)	URLLC	DL and ULL	90% of sessions with 96% reliability (latency budget: 10ms) (both DL and UL)	Non-persistent with $\lambda_{\mathrm{VR}} = 40$ sessions/s (both DL and UL)	Poisson Process with $\lambda_{\mathrm{VR,DL}} = 200$ packets/s and Periodic with $T_{\mathrm{VR,UL}} = 0.004s$	$S_{\mathrm{VR,DL}} = 1.25$ and $S_{\mathrm{VR,UL}} = 0.5$
Video surveillance (VS)	eMBB	UL	25 Mbps	Persistent with $N_{\mathrm{VS}} = 676$ sessions	Periodic with $T_{\mathrm{VS,UL}} = 0.036s$	$S_{\mathrm{VS,UL}} = 4.5$
Sensor monitoring (SM)	mMTC	UL	latency budget: seconds to minutes	Persistent with $N_{\mathrm{SM}} = 675600$ sessions	Periodic with $T_{\mathrm{SM,UL}} = 60s$	$S_{\mathrm{SM,UL}} = 0.2$

Table 4.2: Summary of traffic modelling and their respective QoS requirements.

by two UL symbols [72]. Additionally, for all considered RAN configurations, the chosen packet scheduler is the *modified-largest weighted delay first* (M-LWDF) scheduler, which aims to serve URLLC, eMBB and mMTC flows simultaneously, featuring both channel-adaptive and latency-oriented aspects. Another motivation for choosing the M-LWDF scheduler is based on Chapter 3, where it was shown that the M-LWDF scheduler performs the best in a URLLC slice and also has a good performance in non-sliced networks.

For the configurations using slices, the radio resources are distributed among the slices. This resource assignment is based on the resource utilisation of each service, which is derived via simulations, as shown in [72]. Also, when multiple numerologies are configured, the relevant guard bands are applied. Additionally, edge guard bands are used at either edge of the carrier, whose size depends on the numerology used at the carrier's edge [80]. Figure 4.1 shows an example with two slices configured with numerologies 0 and 1, respectively. The size of the edge guard bands as well as the inter-numerology guard band are dependent on the applied numerologies. Finally, we consider two slicing options: *(i)* slicing per numerology i.e. a slice for each distinct numerology used and *(ii)* slicing per service category i.e. a total of three slices, with each slice related to eMBB, URLLC and mMTC services, respectively.

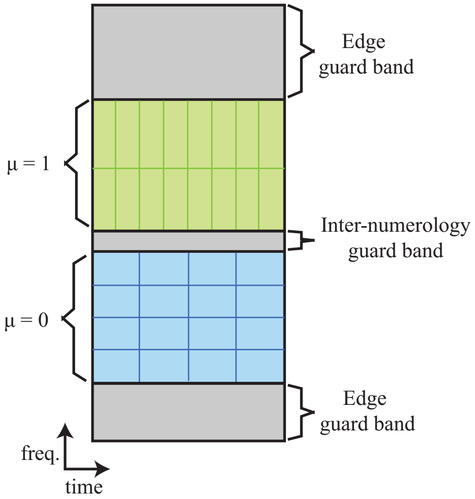


Figure 4.1: Edge and inter-numerology guard bands for a carrier bandwidth with two slices.

In this chapter we analyse the respective (dis)advantages of the flexible numerology, mini-slots, slicing and resource sharing features, as well as sensible combinations of these 5G features in mixed-traffic scenarios. For example, one potential way of handling a mix of URLLC and eMBB/mMTC traffic, which is intrinsically best served with higher and lower numerologies, respectively [43], is to configure a single slice with numerology (in support of eMBB/mMTC services) in combination with mini-slot based scheduling to ensure good URLLC performance. An alternative approach could be to configure distinct URLLC and eMBB/mMTC slices with tailored numerologies, noting however the resource inefficiencies due to the required inter-numerology guard bands and (potentially) trunking losses.

4.3.4. Key Performance Indicator Definitions

Distinct KPIs are defined for the eMBB, URLLC and mMTC services. For the eMBB services the KPI of relevance is the 5th throughput percentile and its target level per service is shown in Table 4.2 under the QoS requirements column. For the URLLC services, the KPI of relevance is the fraction of URLLC sessions per service experiencing the required degree of reliability, as shown in Table 4.2. We define reliability for a URLLC session as the fraction of packets that are successfully received within the given latency budget. Finally, the KPI related to the mMTC services is the 95th latency percentile. Because the latency budget for mMTC services varies between seconds and minutes, a specific target value is not set.

4.4. Simulation Results

The above-described models and KPIs have been implemented in a Python-based dynamic system-level simulator which has been used to conduct a series of scenario-based assessments. The source code generating all results is available in [81]. This section presents the evaluation of the flexible numerology, mini-slot based scheduling, slicing and resource sharing features, as well as combinations of these features. To describe each RAN configuration, we adapt a notation, consisting of the features, combined with an underscore as:

$$\{\text{NS,S}\}\_(\mu_1/\cdots/\mu_Y)\_\{\text{NM,M(x)}\}\_\{\text{NR,R}\},$$

to indicate no slicing (NS) or slicing (S), the use of Y numerologies and their respective values $\mu_1, \cdots, \mu_Y$, no mini-slot based scheduling (NM) or mini-slot based scheduling with approach x (M(x)), where x ∈ {D, NP, P} denotes the default, non-pre-emptive and pre-emptive approaches, respectively, and no resource sharing (NR) or resource sharing (R). The abbreviations of the considered services are shown in Table 4.2. Figure 4.2 illustrates, for each feature combination, the results obtained for each service and their corresponding KPIs separated in three sub-figures, one for each service category (eMBB, URLLC, mMTC). All scenarios have been simulated with a range of distinct random seeds and the correspondingly obtained 95%-confidence intervals are shown in the sub-figures, revealing a reasonable degree of attained statistical accuracy. Furthermore, the dashed lines indicate the target KPI level for the URLLC and eMBB services.

4.4.1. Feature Evaluation

The impact of *numerology* on the performance of the services is shown in Figure 4.2 under configurations 1-3. As expected based on the qualitative arguments in Chapter 2 and the results in Chapter 3, URLLC services benefit from higher numerologies, in particular $\mu = 2$, due to the shortened TTIs while the eMBB and the mMTC services benefit from lower numerologies i.e. $\mu = 0$ due to the higher frequency-domain channel-adaptive scheduling gains.

Because eMBB and mMTC services benefit from numerology 0, configurations 4-6, in Figure 4.2, consider the use of numerology 0 in combination with the three approaches of *mini-slot based scheduling*, respectively, to enhance the performance of the URLLC service. All sub-figures show that configuration 4, which uses the *default approach* for mini-slots, performs better compared to normal slot-based scheduling with numerology 0 (configuration 1). All services benefit from the faster, more resource-efficient and less interfering transmissions enabled by the mini-slots, as qualitatively argued above. In particular, multiple packets can be transmitted in distinct mini-slots within a given slot, whereas under configuration 1 each of those packet transmissions would utilise a full slot by themselves. However, the URLLC sub-figure shows that configuration 4 performs worse than when numerology 2 is used (configuration 3), illustrating the importance of configuring the appropriate numerology.

Configuration 5 considers the use of the *non-pre-emptive mini-slot based scheduling* and the URLLC sub-figure illustrates the gains for the URLLC services compared to the default mini-slot base scheduling (configuration 4). Specifically, the performance is improved because the URLLC packets are immediately transmitted after their arrival in the buffer, assuming that there are enough available resources. Consequently, the performance of the eMBB and mMTC services, which are scheduled only at the start of each slot, is also improved because the buffer is kept short, as shown in the respective sub-figures. Note that the performance of the URLLC service is almost the same compared to only using numerology 2 (configuration 3) whereas the performance of the eMBB and mMTC services is better than only using numerology 0 (configuration 1).

The use of *pre-emptive mini-slot based scheduling* is considered in configuration 6. The URLLC sub-figure shows that the pre-emptive approach provides further gains compared to the non-pre-emptive approach (configuration 5) because the URLLC packets are transmitted immediately upon arrival, pre-empting an on-going transmission if there are no available resources. For the same reason, the eMBB and mMTC sub-figures show a significant performance degradation, which is also reflected by the number of eMBB and mMTC packet retransmissions which is increased by 49% compared to the

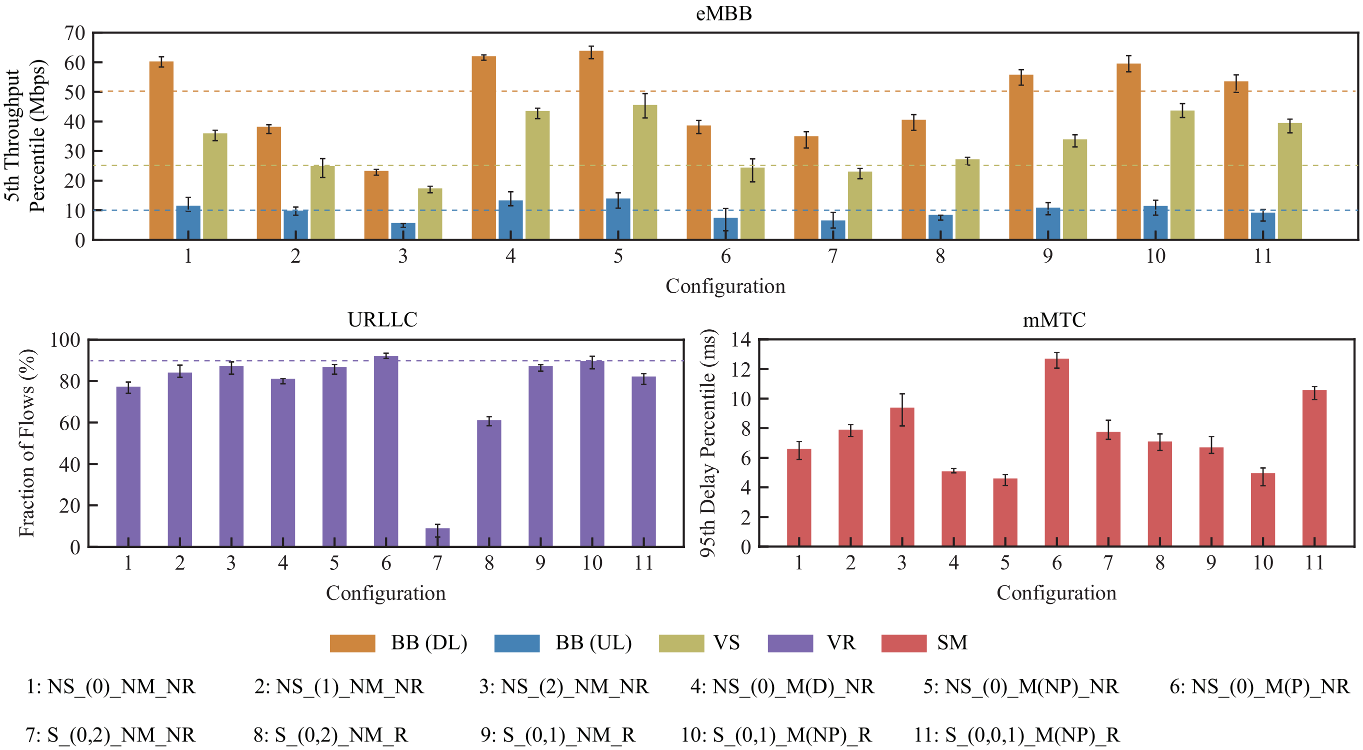


Figure 4.2: Performance of the considered services for configurations related to the numerology, mini-slot based scheduling, slicing and resource sharing features and combinations of these features. The dashed lines indicate the target KPI level for each URLLC and eMBB service.

non-pre-emptive approach. Therefore, among the three mini-slot based scheduling approaches, the non-pre-emptive approach is considered best when taking all services into consideration.

Based on the outcome of the numerology comparisons done in configurations 1-3, configuration 7 assumes two *slices*, one with numerology 0 for the eMBB and mMTC services, and one with numerology 2 for the URLLC service. Figure 4.2 shows that the resource split causes an immense performance degradation compared to a non-sliced RAN. Specifically, the resource split results to trunking losses, the reduction of usable resources due to the needed inter-numerology guard band (further causing higher interference levels) and the decrease of frequency-domain channel-adaptive scheduling gains as the eMBB data are only scheduled on the resources of the assigned slice.

To decrease the effects of the resource split, configuration 8 in Figure 4.2 allows the *sharing of idle radio resources* between the slices. All sub-figures show performance improvement compared to configuration 7, however, configuration 8 performs worse than simply configuring the RAN with numerology 0 (configuration 1), which is due to the guard band effects.

4.4.2. Optimisation of Feature Combinations

Based on the previous results, an evaluation of some promising candidate feature combinations to find the optimal configuration is presented. First, similarly to configuration 8, we consider configuration 9 but the slice related to the URLLC service is now configured with numerology 1 instead of 2 to reduce the size of the middle and edge guard bands by 28%. In Figure 4.2, all sub-figures show that the guard band reduction brings gains to all services, mainly because more resources are now available for data transmissions (in both slices) and secondly because the frequency-selective channel-adaptive scheduling gains are increased in the slice related to the URLLC service. The comparison between configurations 8 and 9 also reveals that choosing the numerology per slice solely based on the service type may not yield optimal performance. Rather the effects of all features should be taken into account when choosing the numerologies. Also, the URLLC sub-figure shows that the performance with configuration 9 is equivalent to that obtained when only numerology 2 is configured (configuration 3) while the eMBB and mMTC sub-figures show that the performance with configuration 9 is similar compared to only configuring numerology 0 (configuration 1). Additionally, the performance of configuration 9 is similar to the performance of configuration 5, with both configurations achieving the targets for the eMBB services and providing a similar performance for the URLLC service.

Subsequently, the combination of two slices, with numerology 0 and 1, idle resource sharing and non-pre-emptive mini-slot based scheduling, which was found to be the best performing mini-slot approach, is considered as configuration 10. Figure 4.2 shows that the performance of all services improves, compared to configuration 9, because of the gains that mini-slots bring. Furthermore, with configuration 10, the targets for all services are achieved.

To evaluate the way of slicing, i.e. per numerology or per service category, configuration 11 considers the same feature combination as configuration 10 but with three slices i.e. eMBB, mMTC and URLLC slices, with numerologies 0, 0 and 1, respectively. Figure 4.2 shows that configuration 11 performs worse compared to configuration 10 because of the additional trunking losses introduced by the extra slice.

4.4.3. Summary

The best overall performance is provided with configuration 10, achieving the KPI target values of all the services. This result supports the commonly argued potential of RAN slicing and highlights that its inherent trunking losses can be compensated by smartly applying available RAN features.

4.5. Concluding Remarks

In this chapter we have assessed the merit of distinct 5G RAN features to support an integrated services smart city environment, particularly concentrating on flexible numerology, mini-slots, slicing and idle resource sharing. Performance was shown to be optimised when all RAN features are appropriately combined and configured. This highlights the commonly argued potential of RAN slicing, for which it was shown that the slice-specific numerology should not be solely based on the service type in the respective slices, but rather on the combined effects of all involved RAN features.

II

Collaborative Learning

5

Agent Selection Framework

Federated learning is an effective method to train a machine learning model without requiring to aggregate the potentially sensitive data of agents in a central server. However, the limited communication bandwidth, the hardware of the agents and a potential application-specific latency requirement impact how many and which agents can participate in the learning process at each communication round. In this chapter, we propose a selection metric characterising each agent's importance with respect to both the learning process and the resource efficiency of its wireless communication channel. Leveraging this importance metric, we formulate a general agent selection optimisation problem, which can be adapted to different environments with latency or resource-oriented constraints. Considering an example wireless environment with latency constraints, the agent selection problem reduces to the 0/1 Knapsack problem, which is solved with a fully polynomial approximation. Then, the agent selection method is evaluated in different scenarios, using extensive simulations for an example task of object classification of European traffic signs. The results indicate that agent selection methods which consider both learning and channel aspects provide benefits in terms of the attainable global model accuracy and/or the time needed to achieve a targeted accuracy level. However, in scenarios where agents have a limited number of data samples or where the latency requirement is very stringent, a pure learning-based agent selection method is shown to be more beneficial during the early or late stages of the learning process.

This chapter is based on a report [82].

5.1. Introduction

Traditional ML algorithms are performed in a central location, where large amounts of data are aggregated and used to train the ML model. However, data can belong to different agents who may be unwilling to share them due to privacy concerns. Additionally, agents can generate a large amount of data in a short period of time which can saturate the communication channel if all data from all agents need to be aggregated at a central location.

Addressing the difficulties of centralised ML algorithms, McMahan et al. [83] introduced *federated learning* (FL), a decentralised ML technique to train a centralised *global model* using decentralised data from multiple agents and without sharing the raw data at the agents. Specifically, the agents train their own *local model*, which has the same neural network architecture as the global model, with their own data. After local training, the agents only transmit the tuned parameters of their local model to the FL server, which is then responsible for generating a new global model by combining the received model parameters from the contributing agents. The process repeats for a number of *communication rounds* until the global model converges to a satisfactory accuracy level. The most popular method for FL is `FederatedAveraging` (or `FedAvg`) [83]. Agents apply the stochastic gradient descent (SGD) optimiser, for a number of local iterations, and the global model is generated by averaging the submitted local models.

A challenge of FL is that the exchange of model parameters between the agents and the FL server can come at a high communication cost, especially for models with a large number of parameters [84]. This is of particular relevance in wireless network scenarios, e.g. when considering applications such as autonomous driving and the internet of things, which rely on resource-constrained wireless networks [85]. Also, wireless channels impose further challenges as they are susceptible to interference, have limited resources and their quality varies over location, time and frequency. To address these challenges, a subset of agents is selected to participate in a given communication round. Furthermore, although the `FedAvg` method can perform very well [83, 86], its performance can degrade significantly when data at the agents are not independent and identically distributed (non-IID), i.e., heterogeneous, across all agents [87]. Therefore, the selection of agents influences the convergence time and accuracy of the global model.

5.1.1. Related Work

In the literature, the agent selection problem has been addressed from both a pure FL perspective and for the specific setting of a wireless network. From the FL perspective, the effects of randomly selecting a large number of agents is addressed by Charles et al. [88]. Rather than randomly selecting agents, Cho et al. [89] show that selection of agents based on their local loss improves the convergence of the global model, even for scenarios with heterogeneous data. The local loss is also considered by Lai et al. [90], who perform agent selection with a statistical utility function. Nguyen et al. [91] perform agent selection considering the gradient information of each agent, while Chen et al. [92] use the norms of the updates of each agent. Ribero and Vikalo [93] suggest the selection of agents based on the progression of the agents' local weights with respect to time. However, none of the above works consider a wireless network.

Considering a wireless network, Nishio and Yonetani [94] propose a greedy method to maximise the number of selected agents during a time interval. Yang et al. [95] compare the performance of the random, round robin and proportional fair schedulers, in terms of the FL convergence rate, for scenarios with limited bandwidth and interference. Amiri et al. [96] show that selecting agents based on both their wireless channels and the l_2-norm of their local model update provides better performance than only considering one of the two metrics individually. Shi et al. [97] consider latency-constrained systems and aim to maximise the model accuracy within a given total latency budget. However, none of the above works provide a single metric to characterise an agent based on its importance to the learning process and its resource consumption.

The global model convergence, with the `FedAvg` method, under non-IID training data is addressed in the literature in many ways, including with data sharing [98, 99] and with regularisation [100–102]. Convergence guarantees have also been derived for scenarios with non-IID data [103] and it is suggested that for many real-world applications, the `FedAvg` method can provide identical performance for IID and non-IID data [104]. Therefore, motivated by the research so far, we employ the `FedAvg` method in this work.

5.1.2. CONTRIBUTIONS AND OUTLINE

The main contributions of this chapter are the following:

- We propose a metric to characterise the agents based on their importance in the learning process as well as their resource consumption, which depends on the FL model and the agents' wireless channels and hardware.

- We propose an agent selection framework, which considers both agent-specific and system constraints, in terms of an optimisation problem. The optimisation problem can be adjusted to cater the network's and agents' needs and constraints. Additionally, the optimisation problem can be easily extended to address the joint agent selection and resource allocation problem in FL scenarios over wireless networks.

- We show that for scenarios with non-IID data, the local loss is a better metric than the deviation between the local and the global model to characterise the importance of an agent in the learning process.
- We demonstrate that the learning accuracy is improved when both learning and channel aspects are considered to characterise the agents. However, when the agents have few samples or when stringent latency requirements apply, a higher accuracy is achieved at the global model when the wireless channels are ignored.

The remainder of this chapter is organised as follows. Section 5.2 provides the learning and communication models. In Section 5.3, the agent characterisation is presented, while the general problem formulation is shown in Section 5.4. Section 5.5 presents the considered use case for evaluating the agent selection framework and Section 5.6 provides the evaluation of the agent selection policies. Finally, the conclusions are presented in Section 5.7.

5.2. System Model

Consider a cellular network with one BS, which also acts as an FL server and a set $\mathcal{V}$ of agents, where $V = |\mathcal{V}|$ is the number of agents. The FL server and the agents collaboratively train a global model, which allows the training of a local model without requiring the transmission of the data sets gathered by the agents. Therefore, each agent $v \in \mathcal{V}$ holds its own training data set $\mathcal{K}_v$ and testing data set $\mathcal{K}_{T,v}$, where $K_v = |\mathcal{K}_v|$ and $K_{T,v} = |\mathcal{K}_{T,v}|$ denote the number of training and testing data samples available at agent v, respectively.

Figure 5.1 shows the schematic overview of a communication round i, assuming a network with $V = 2$ agents. First, the FL server selects and notifies the agents that will participate in the learning. The agent selection and notification, potentially using a broadcast transmission, are assumed to be performed within a time interval of duration τ_{SCH}. Then, each selected agent $v \in \mathcal{V}_G[i]$ trains its local model, where $\mathcal{V}_G[i]$ is the set containing the selected agents at communication round i. An agent $v \in \mathcal{V}_G[i]$ has a training time $\tau_{\text{T},v}$, which can be different to other agents, as also shown in Figure 5.1, because agents may have different data set sizes and/or processing capabilities.

Once each selected agent $v \in \mathcal{V}_G[i]$ finishes its local training, it transmits its local model to the BS/FL server, via the UL channel. The UL transmission time $\tau_{\text{UL},v}$ for agent v depends on its channel quality, as it will be outlined later. In Figure 5.1, agent 2 does not initiate its UL transmission until the completion of the UL transmission of agent 1, because the BS can only receive data from only one agent at a time, in this specific example. After all required local model uploads are completed, the FL server is responsible to update the global model for the next communication round $i+1$. We assume

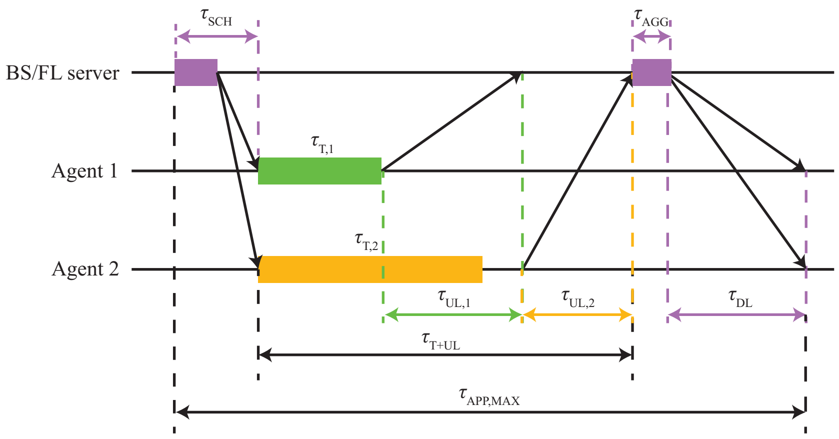


Figure 5.1: Schematic overview of the different steps involved in a single communication round and the corresponding time duration intervals.

that this process occurs within a time interval of duration τ_{AGG}. Finally, the FL server broadcast, after some time τ_{DL}, the new global model to each agent $v \in \mathcal{V}$. The process repeats until sufficient accuracy is achieved at the global model, based on an FL server- or agent-specific testing data set, or an application-specific latency deadline is reached.

An application-specific latency deadline $\tau_{\text{APP,MAX}}$ can be set on the time duration of each communication round i to prevent the selection of agents with limited training power and/or with poor wireless channel quality, thus

$$\tau_{\text{APP,MAX}} \geq \tau_{\text{SCH}} + \tau_{\text{T+UL}} + \tau_{\text{AGG}} + \tau_{\text{DL}}, \tag{5.1}$$

where $\tau_{\text{T+UL}}$ is the time needed for all selected agents to perform local training and upload their local models to the BS/FL server, as also shown in Figure 5.1. We assume that the processing times at the BS/FL server are negligible, because it has significantly more powerful hardware compared to the agents. We further assume that the time duration of the broadcast to notify the selected agents for training is negligible, because the control data transmitted is very small in size. Therefore, $\tau_{\text{SCH}} \approx \tau_{\text{AGG}} \approx 0$. Finally, in this chapter we neglect the broadcast time τ_{DL} because it is network-dependent and fixed for every communication round i. Consequently, using (5.1), agent selection should be performed such that

$$\tau_{\text{T+UL}} \leq \tau_{\text{APP,MAX}}. \tag{5.2}$$

Additionally, the FL process can be bound to the available transmission resources $C_{\text{R,MAX}}$ allocated to the FL task, e.g. in a slice in 5G networks, which can restrict the number of selected agents per communication round. The rest of this section describes in more detail the learning model and the model for the communication between the agents and the FL server.

5.2.1. Learning Model

Consider an agent v, with training data set $\mathcal{K}_v$. Then, its input data $\mathbf{X}_v = [\mathbf{x}_{v1}, \cdots, \mathbf{x}_{vK_v}]$, where $\mathbf{x}_{vk} \in \mathbb{R}^{n_X}$ denotes the k^{th} input vector to the model of agent v, with n_X as the size of the input vector. Additionally, the output data $\mathbf{Y}_v = [\mathbf{y}_{v1}, \cdots, \mathbf{y}_{vK_v}]$, where $\mathbf{y}_{vk} \in \{0,1\}^{n_C}$ denotes the real output vector associated with the k^{th} input vector $\mathbf{x}_{vk}$ and n_C is the size of the output vector and hence, the number of model outputs. For example, for an object classification learning task with n_C classes, the real output $\mathbf{y}_{vk}$ indicates with value 1 the class that sample k belongs to, whereas for all other classes, it holds a 0 value.

During local training with training data set $\mathcal{K}_v$, the model output (or predictions) $\hat{\mathbf{Y}}_v = [\hat{\mathbf{y}}_{v1}, \cdots, \hat{\mathbf{y}}_{vK_v}]$ is generated, where $\hat{\mathbf{y}}_{vk} \in \mathbb{R}^{n_C}$ denotes the predicted output vector related to the k^{th} input vector $\mathbf{x}_{vk}$. The model output $\hat{\mathbf{Y}}_v$ depends on the considered model architecture, e.g. the number of hidden layers in the case of a deep neural network. The weights $\mathbf{W}_v$ parameterise the considered local model and the goal of the local model is to tune its weights $\mathbf{W}_v$ such that the predictions $\hat{\mathbf{Y}}_v$ will represent the real output $\mathbf{Y}_v$, given the input data $\mathbf{X}_v$. The relation between the predictions $\hat{\mathbf{Y}}_v$ and the real output $\mathbf{Y}_v$ is typically measured with the loss function $F(\mathbf{W}_v;\mathbf{X}_v,\mathbf{Y}_v)$, which also depends on $\mathbf{X}_v$ and $\mathbf{Y}_v$. From here onwards, we omit this dependency for the sake of simplifying the notation.

The objective of tuning the local model is

$$\min_{\mathbf{W}_v} F(\mathbf{W}_v) = \frac{1}{K_v} \sum_{k \in \mathcal{K}_v} f_k(\mathbf{W}_v),$$

where $f_k(\mathbf{W}_v)$ is the loss function of sample k, which is commonly set to the cross-entropy loss for classification problems [105]. To find the weights $\mathbf{W}_v$ which minimise the loss function $F(\mathbf{W}_v)$, a number of iterations n_{LE} are performed, known as local epochs. Assuming the SGD optimiser [106], the weights $\mathbf{W}_v$ are adapted at every local epoch based on the learning rate η, which controls the learning speed of the model.

In an FL setting, a data set $\mathcal{K}$, where $K = |\mathcal{K}|$, is the collection of the data sets $\mathcal{K}_v$ from all agents in set $\mathcal{V}$ and hence $\mathcal{K} = \cup_{v \in \mathcal{V}} \mathcal{K}_v$. With the `FedAvg` method [83] and assuming that the global model is generated only based on the models of the selected agents, the loss of the FL server, at communication round i, is upper bounded by the weighted average of the local losses

$$\min_{\mathbf{W}_G} F(\mathbf{W}_G[i]) \leq \sum_{v \in \mathcal{V}_G[i]} \frac{K_v}{K} F(\mathbf{W}_v[i]),$$

where $\mathbf{W}_v[i]$ denotes the weights of the local model of agent v, after local training during communication round i and $\mathbf{W}_G[i]$ are the weights of the global model. Using this upper bound, FL approximates the global objective function $F(\mathbf{W}_G[i])$ by the weighted average of the local losses. Then, assuming the SGD optimiser, the weights $\mathbf{W}_G[i]$ of the global model at the end of communication round i are updated as follows

$$\mathbf{W}_G[i] \leftarrow \sum_{v \in \mathcal{V}_G[i]} \frac{K_v}{K} \mathbf{W}_v[i] \tag{5.3}$$

and then transmitted to all of the agents for the next communication round $i+1$.

5.2.2. Communication Model

For the transmission of the local model, assuming a wireless link, we measure the bit rate R_v of agent v in bits per second with the Shannon–Hartley equation [40] as

$$R_v = B_v \log_2 \left(1 + \frac{P_{\text{TX},v} G_{T,v}}{P_{\text{NOISE}}} \right), \tag{5.4}$$

where B_v is the transmission bandwidth of agent v in MHz, $P_{\text{TX},v}$ is the transmit power in Watt, $G_{T,v}$ is the transmission gain in linear units and P_{NOISE} is the thermal noise power in Watt. The transmission gain $G_{T,v}$ is given, in dB, by [40]

$$G_{T,v} = 20 \log \left(\frac{c}{4\pi f_C} \right) - 10\gamma \log(d_v) + \psi_{\text{SF}}, \tag{5.5}$$

where c is the speed of light, f_C is the carrier frequency, d_v is the three-dimensional distance between agent v and the BS, γ is the path loss exponent and ψ_{SF} is a Normally-distributed random variable with zero mean and standard deviation σ_{SF}, capturing the effects of shadow fading.

5.3. Agent Characterisation

In real-world applications, agents are diverse in terms of their training data as well as processing capabilities (e.g. central processing unit (CPU)) to train their local model, wireless channel quality and energy availability. In this section we define two metrics for measuring the importance of agents in the learning process and we describe the potential resource consumption of the agents.

5.3.1. Learning Process Importance

Non-random agent selection can improve the FL model convergence [89–93]. Hence, we characterise an agent v at communication round i based on its importance $q_{L,v}[i]$ in the learning process. Specifically, we consider two metrics which can express the importance $q_{L,v}[i]$ of agent v: *(a)* the deviation $\delta_v[i]$ and *(b)* the loss $F(\mathbf{W}_{G,v}[i])$, which are both defined below.

Inspired by regularisation to address the challenges of non-IID data [100], we propose as a metric for the importance $q_{L,v}[i]$ in the learning process, the *deviation*

$$\delta_v[i] = ||\mathbf{W}_v[i_v] - \mathbf{W}_{\mathbf{G}}[i-1]||_2^2,$$

which represents the deviation between the local model $\mathbf{W}_v[i_v]$ of agent v and the global model $\mathbf{W}_{\mathbf{G}}[i-1]$, where $||\cdot||_2$ denotes the Euclidean norm and i_v denotes the most recent communication round that agent v was selected for training. The deviations can be calculated at the FL server and consequently used in the agent selection process without any additional signalling from the agents. The reason is due to the assumption that the FL server always stores the weights of agents from their last participation in the learning process until their next participation.

Alternatively, the importance $q_{L,v}[i]$ in the learning process can also be expressed [89, 99] in terms of the *loss* function $F(\mathbf{W}_{G,v}[i])$ of agent v, which is locally computed at agent v with the testing data set $\mathcal{K}_{T,v}$ and the newly generated global weights $\mathbf{W}_G[i]$ at the end of communication round i. Then, the loss $F(\mathbf{W}_{G,v}[i])$ is transmitted to the FL server as an input for the agent selection process of communication round $i+1$. The time needed to calculate the loss $F(\mathbf{W}_{G,v}[i])$ is addressed in the following subsection and we consider the corresponding transmission time to be negligible due to the loss being a scalar value.

5

5.3.2. Resource Consumption

When an agent participates in the learning process during communication round i, it consumes resources. We characterise the total resource consumption of an agent v based on the resource consumption for the transmission $C_{R,v}[i]$, processing $C_{T,v}[i]$ and energy $C_{E,v}[i]$, respectively. The transmission $C_{R,v}[i]$ and processing $C_{T,v}[i]$ consumption relate to system-specific resources, e.g. bandwidth and time, while the energy $C_{E,v}[i]$ consumption is agent-specific. In the following, we detail the consumption for these three types of resources.

Transmission Resources

The consumption of the transmission, i.e. time-frequency, resources $C_{R,v}[i]$ of agent v is related to the upload of the local model $\mathbf{W}_v[i]$ at communication round i and depends on the communication system. We consider an orthogonal frequency division multiple access (OFDMA) system and assume that the channel is static during the communication round i and over the applied frequency carrier. Then, the consumed transmission resources $C_{R,v}[i]$ are given by

$$C_{R,v}[i] = \tau_{\mathrm{UL},v}[i]B_v[i] = \frac{Z}{R_v[i]}B_v[i], \tag{5.6}$$

where $\tau_{\mathrm{UL},v}[i]$ is the transmission time in seconds, as shown in Figure 5.1, $B_v[i]$ is the transmission bandwidth in MHz, $R_v[i]$ is the bit rate as given in (5.4) and Z is the size of the model in MB. The resource consumption $C_{R,v}[i]$ is important for systems with limited resources because it can be exploited for efficient agent selection. Also, the calculation of the resource consumption $C_{R,v}[i]$ does not require additional communication between the agents and the BS, because the bit rates can be estimated by the BS via the periodic channel quality indicator feedback that all agents report to the network.

PROCESSING RESOURCES

The consumption $C_{T,v}$ related to the local model training by agent v is measured in terms of time and depends on the agent's processing capability g_v, as well as on its data set size K_v and other training-related parameters. Specifically, the consumption $C_{T,v}$ is identical to the training time duration $\tau_{\mathrm{T},v}$, which was introduced in Section 5.2. Assuming a fixed training data set size K_v, the consumption $C_{T,v}$ is the same for any communication round. Specifically, the processing capability g_v of agent v is measured in floating point operations (FLOPs) per second as [107]

$$g_v = n_{\mathrm{CORES},v}\, \nu_v\, \omega_v,$$

where $n_{\mathrm{CORES},v}$ is the number of CPU cores at agent v, ν_v is the CPU clock frequency at agent v in cycles per second and ω_v is the number of FLOPs per cycle at agent v. Then, the time consumption $C_{T,v}$ for training at agent v is

$$C_{T,v} = \left\lceil \frac{K_v}{s_B} \right\rceil \frac{n_{\mathrm{FLOP,G}}\, n_{\mathrm{LE}}}{g_v},$$

where $n_{\mathrm{FLOP},G}$ denotes the number of FLOPs to train the model for a batch of size s_B and $\lceil\cdot\rceil$ represents the ceiling operation.

When the loss $F(\mathbf{W}_{G,v}[i])$ of agent v is considered for the importance $q_{L,v}[i]$ in the learning process, an additional term can be added to the training time consumption $C_{T,v}$ to represent the loss calculation:

$$C_{T,v} = \left\lceil \frac{K_v}{s_B} \right\rceil \frac{n_{\mathrm{FLOP,G}}\, n_{\mathrm{LE}}}{g_v} + \left\lceil \frac{K_{T,v}}{s_B} \right\rceil \frac{n_{\mathrm{FLOP,G}}}{g_v}, \tag{5.7}$$

assuming a fixed testing data set size $K_{T,v}$.

The time consumption $C_{T,v}$ can only be measured locally at the agent and hence, it should be communicated to the FL server when the agent first enters the network. The knowledge of the time consumption $C_{T,v}$ at the FL server is significant because it allows the FL server to perform resource-efficient agent selection within a given latency bound.

Energy Resources

The energy consumption $C_{E,v}[i]$ of an agent v, during communication round i covers both the training and the wireless transmission. Applying the model in [107], the energy consumption $C_{E,v}[i]$ is given, in Joules, by

$$C_{E,v}[i] = \frac{e_v}{\omega_v^3} \left\lceil \frac{K_v}{s_B} \right\rceil g_v^2 \, n_{\text{FLOP},G} \, n_{\text{LE}} + P_{\text{TX},v}[i] \tau_{\text{UL},v}[i],$$

where e_v is the energy consumption coefficient, measured in Watt(cycles/s)$^{-3}$ and it is based on the CPU. When considering agents with energy limitations, the energy consumption $C_{E,v}[i]$ is an important metric because the available energy level $E_v[i]$ of the agent should exceed the energy consumption $C_{E,v}[i]$ required to participate in the learning process.

When energy aspects are taken into account during agent selection, the energy consumption $C_{E,v}[i]$, as well as the total available energy $E_v[i]$, need to be reported to the FL server. Specifically, the training-related energy consumption needs to be reported to the FL server once, when first entering the network, while the transmission-related energy consumption can be estimated at the FL server for the same reasons given for the transmission resource consumption $C_{R,v}[i]$. Furthermore, the energy level $E_v[i]$ of an agent v is dependent on the communication round i, because it decreases by the agent's energy consumption, every time agent v is selected and hence the energy level $E_v[i]$ should be periodically reported to the FL server.

5.4. Problem Formulation

We define the *agent importance* $q_v[i]$ as the metric governing the agent selection process to improve the performance of the FL model by exploiting the different characteristics of the agents. The agent importance $q_v[i]$ is defined to capture the trade-off between the importance $q_{L,v}[i]$ of the agent v in the learning process against the total resource consumption of the agent v as follows:

$$q_v[i] = \frac{q_{L,v}^{\rho_L}[i]}{C_{R,v}^{\rho_R}[i] C_{T,v}^{\rho_T} C_{E,v}^{\rho_E}[i]}, \tag{5.8}$$

with $\rho_L + \rho_R + \rho_T + \rho_E = 1$, where $\{\rho_L, \rho_R, \rho_T, \rho_E\} \in [0,1]$ are constants to tune the relative significance of the learning importance $q_{L,v}[i]$ and the consumed transmission $C_{R,v}[i]$, processing $C_{T,v}$ and energy $C_{E,v}[i]$ resources, respectively. Moreover, the agent importance $q_v[i]$ can be fine-tuned to the requirements and constraints of the agents and the system. For example, when the network is very congested, higher emphasis can be given to the transmission resource consumption of agents by increasing the value of ρ_R.

For a given communication round i, we formulate the following general agent selection optimisation problem to maximise the total agent importance:

$$\max_{s_1[i],\cdots s_V[i]} \sum_{v\in\mathcal{V}} q_v[i]s_v[i] \tag{5.9a}$$

$$\text{subject to} \sum_{v\in\mathcal{V}} C_{R,v}[i]s_v[i] \leq C_{\text{R,MAX}}[i], \tag{5.9b}$$

$$(C_{T,v} + \tau_{\text{UL},v}[i])s_v[i] \leq \tau_{\text{APP,MAX}},\ \forall v \in \mathcal{V}, \tag{5.9c}$$

$$C_{E,v}[i]s_v[i] \leq E_v[i],\ \forall v \in \mathcal{V}, \tag{5.9d}$$

$$g_v s_v[i] \geq g_{\text{MIN}},\ \forall v \in \mathcal{V}, \tag{5.9e}$$

$$s_v[i] \in \{0,1\},\ \forall v \in \mathcal{V}, \tag{5.9f}$$

where the binary optimisation variable $s_v[i]$ indicates whether agent v is selected at communication round i or not. Constraint (5.9b) indicates that the transmission resources allocated to the agents should not exceed the total system resources allocated to the FL task at communication round i. Constraint (5.9c) shows that the selected agents should train and transmit their models within an application-specific latency budget $\tau_{\text{APP,MAX}}$. Constraint (5.9d) ensures that the selected agents have sufficient energy levels to participate in the learning process. Finally, constraint (5.9e) ensures that the processing capabilities of the selected agents exceed a minimum requirement g_{MIN}, to avoid selecting agents with potentially long training times.

5

The optimisation problem in (5.9) is general and it can be adjusted to the communication system, the application-specific requirements and the energy constraints of the agents. Moreover, the optimisation problem can be extended to jointly consider agent selection and radio resource allocation, for example when multiple agents can be simultaneously served with beamforming antennas. The goal of this chapter is to analyse the performance of problem (5.9) and investigate the trade-offs between learning and wireless communication performance measures.

We consider the communication system described in Section 5.3 and wideband radio resource scheduling. Then, the available transmission resources $C_{\text{R,MAX}}[i]$, in constraint (5.9b), can be expressed in terms of the application-specific latency budget $\tau_{\text{APP,MAX}}$. For example, the transmissions of the local models to the FL server will start once the agent with the shortest training time that participates in the given communication round i finishes its training. Considering that the duration of each communication round is given by the latency budget $\tau_{\text{APP,MAX}}$, the available transmission resources $C_{\text{R,MAX}}[i]$ are given by:

$$C_{\text{R,MAX}}[i] = B\left(\tau_{\text{APP,MAX}} - \min_{v\in\mathcal{V}_G[i]} C_{T,v}s_v[i]\right), \tag{5.10}$$

where B is the system bandwidth. Moreover, when agents with the same training time

$C_{T,1} = \dots = C_{T,V} = C_T$ are considered, (5.10) is simplified to

$$C_{\mathrm{R,MAX}} = B(\tau_{\mathrm{APP,MAX}} - C_T), \tag{5.11}$$

and the available transmission resources $C_{\mathrm{R,MAX}}$ are independent of the communication round i. For the remainder of this work, for simplicity reasons, we consider that agents have the same training time C_T. Furthermore, when using (5.11) in constraint (5.9b), constraint (5.9c) becomes redundant.

It is also important to consider scenarios with powerful agents, e.g. vehicles or agents that have powerful hardware and access to charging points. For such scenarios, constraints (5.9d) and (5.9e) can be ignored. Then, we can further simplify the optimisation problem in (5.9) to:

$$\max_{s_1[i],\cdots s_V[i]} \sum_{v\in\mathcal{V}} q_v[i] s_v[i] \tag{5.12a}$$

$$\text{subject to} \sum_{v\in\mathcal{V}} C_{R,v} s_v[i] \leq B(\tau_{\mathrm{APP,MAX}} - C_T), \tag{5.12b}$$

$$s_v[i] \in \{0,1\},\ \forall v \in \mathcal{V}. \tag{5.12c}$$

The problem formulation in (5.12) emphasizes the latency constraints that might be imposed by the system. Moreover, when problem (5.12) has all parameters discretised, it becomes the classic 0/1 Knapsack problem, which is a known NP-hard problem [108]. For problems with a small number of variables and constraints a pseudo-polynomial algorithm using dynamic programming solves the integer 0/1 Knapsack problem optimally in $O(qV)$ time [108], where $q = \sum_{v\in\mathcal{V}} q_v$. The complexity can be reduced by a fully polynomial approximation. In this work, we apply the algorithm presented in [108], with $\epsilon = 0.001$, to find the approximate solution of problem (5.12). To apply the algorithm solving the integer 0/1 Knapsack problem, all parameters in (5.12) need to be integers. For this reason, we discretise all parameters by multiplying them with a large number and rounding them to the closest integer.

5.5. Use Case

This section presents the considered use case for evaluating the agent selection framework in terms of the learning task, agent selection policies and configured parameters.

5.5.1. Learning Task

As an example application, we perform the learning task of object classification on the European traffic sign data set (ETSD) which consists of 164 classes of signs aggregated over data sets from six European countries [109]. Due to the limited number of training samples per class, we only select the $n_C = 10$ classes with the highest number of samples.

For the classification task, we use a convolutional neural network (CNN) architecture similar to that by Serna and Yuichek [109] and Chiamkurthy [110], which are both inspired by the typical visual geometry group (VGG) architecture [111]. Figure 5.2 shows the considered architecture. Specifically, four convolutional layers are activated with a rectified linear unit (ReLU) function and followed by batch normalisation. Further, max pooling and dropout regularisation with a range of 0.25 are performed. For the fully connected layer, the output of the convolutional layer is flattened and then activated with ReLU. Then, another dropout regularisation is performed with a range of 0.5, followed by a batch normalisation. Finally, the last layer is activated by a softmax with 10 outputs with each output indicating the probability for each class. Batch normalisation makes the network learn robustly [106], while the dropout layers prevent overfitting [106]. In total the network consists of 3349418 trainable parameters and thus, the size of the model $Z = 13.4$ MB, assuming 32-bit precision per parameter.

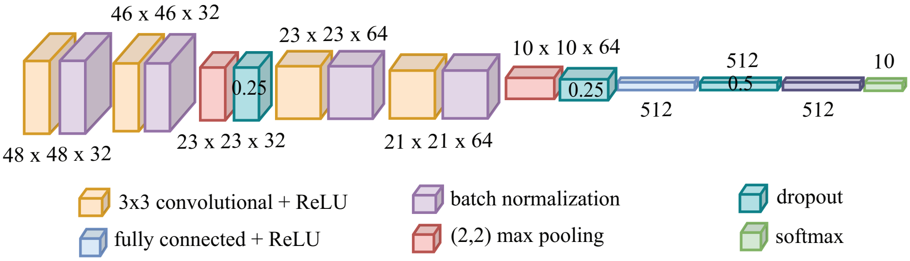


Figure 5.2: The considered CNN architecture to perform object classification task for the ETSD.

5.5.2. AGENT SELECTION POLICIES

Depending on the configuration of the agent importance $q_v[i]$ in (5.8), different agent selection policies are derived as solutions to problem (5.12). Considering that all agents have the same hardware and hence the same training time C_T, we set $\rho_T = 0$ for the agent importance calculation in (5.8). The energy consumption in relation to training is also the same for all agents. Thus, the total energy consumption $C_{E,v}[i]$ depends only on the model transmission and consequently on the bit rate, which is covered by the consumption of the transmission resources $C_{R,v}[i]$. Thus, we set $\rho_E = 0$ in (5.8).

Considering that the agent importance $q_v[i]$ in (5.8) is now tuned with the constants ρ_L and ρ_R, we consider two extreme cases. For the extreme case of $\rho_L = 1$ and $\rho_R = 0$, two solutions of the problem in (5.12) are derived. The first considers the deviation $\delta_v[i]$, whereas the second considers the loss $F(\mathbf{W}_{G,v}[i])$ as a metric for the learning importance $q_{L,v}[i]$ of agent v. We refer to the two solutions as `max-sum-dev` and `max-sum-loss`, respectively, because they aim to maximise the sum of the deviations/losses over all selected agents. For the other extreme case, i.e. where $\rho_L = 0$ and $\rho_R = 1$, only one solution exists, which is denoted as `max-sum-rate` because it aims to maximise the sum of the bit rates of the selected agents. Simulations showed that the performance of the policies with $\rho_L \in (0,1)$ and $\rho_R = 1-\rho_L$ are bounded by the `max-sum-dev` (or `max-sum-loss`) and the `max-sum-rate` policies. Therefore, we do not consider these policies in our evaluation.

Apart from comparing the above-mentioned agent selection policies, we compare them to baseline selection policies, viz. the `random`, `max-dev` and `max-loss` policies. The latter two baseline policies are derived as an approximation to the solution of the problem in (5.12). Specifically, they sort the agents in descending order based on their importance $q_v[i]$ (based on deviation or loss, respectively) and select as many agents as possible until constraint (5.12b) is violated. In line with this approach, sorting and selecting agents based on their bit rates by setting $\rho_R = 1$, yields a selection policy that is identical to the `max-sum-rate` policy. Hence, it does not offer a further selection policy to be assessed. The main difference between the policies derived as a solution to the problem in (5.12) and the baseline policies is that the former class of policies explicitly takes the bit rates into account, even when $\rho_R = 0$, due to constraint (5.12b). The latter class of policies implicitly take the bit rates into account and thus we refer to them as channel-oblivious.

5.5.3. Parameters

In the analysis we consider scenarios with $V = 50$ agents and both IID and non-IID data. For the IID scenario, all agents have the same number of samples K_v, which are evenly distributed over the ten classes. For the non-IID scenario, all agents have K_v samples, which are unevenly split over two classes such that on average all classes are equally represented in the training data set $\mathcal{K}$. For the calculation of the loss $F(\mathbf{W}_{G,v}[i])$ of agent v at communication round i, the categorical cross-entropy loss function is applied on the testing data set $\mathcal{K}_{T,v}$, which is unique for every agent and three times smaller than the training data set $\mathcal{K}_v$.

For the training, the agents invoke the SGD optimiser with learning rate $\eta = 0.05$, batch size $s_B = 64$ and with each agent performing $n_{\mathrm{LE}} = 2$ local epochs. The number of FLOPs required from the agents to train the CNN for a batch size $s_B = 64$ is measured by the Keras library, in Python, which is $n_{\mathrm{FLOP},G} = 6.55$ GFLOPs. Regarding the hardware of the agents, we consider the processing capabilities $g_v = 64$ GFLOPs per second. Such processing power will be given, for example, by $n_{\mathrm{CORE}} = 1$ CPU core, $\nu = 2$ GHz CPU frequency and $\omega = 32$ FLOPs per cycle. Since the agents are assumed to have the same hardware, their energy coefficient [112] is also identical $e_v = 10^{-27}$ Watt(cycles/sec)$^{-3}$.

For the wireless communication scenario, we consider an urban macro environment at $f_C = 3.5$ GHz and a bandwidth of $B = 50$ MHz [30]. For the wireless propagation, we assume a path loss exponent $\gamma = 3.7$ and shadowing with $\sigma_{\mathrm{SF}} = 8$ dB, which are typical values for outdoor dense urban environments [40]. Additionally, the agents are uniformly distributed in a cell of radius 150 m. The transmission of the local model is performed assuming the agents' maximum transmit power $P_{\mathrm{TX},v} = P_{\mathrm{V,MAX}} = 24$ dBm [113]. Finally, the thermal noise power is $P_{\mathrm{NOISE}} = -97$ dBm.

5.6. EVALUATION

This section presents the evaluation of the considered agent selection policies in terms of the accuracy of the global model, which is measured at the FL server based on its specific testing data set. First, we consider a Scenario 0, where all agents have the same bit rates. For this scenario, our aim is to study the policies from a purely learning perspective and hence provide insights into the learning behaviour when the deviation $\delta_v[i]$ and the loss $F(\mathbf{W}_{G,v}[i])$ are applied as metrics for the agent importance $q_{L,v}[i]$ to the learning. Besides Scenario 0, we compare the policies in Scenarios 1, 2 and 3, in which the agents have distinct bit rates. Specifically, in Scenarios 1, 2 and 3, we show the impact of the wireless channel, the number of samples K_v and the application-specific latency budget $\tau_{\mathrm{APP,MAX}}$, respectively.

Sections 5.6.1 to 5.6.4 present the accuracy of the global model for Scenarios 0-3, respectively. Then, in Section 5.6.5, we provide a comparison of Scenarios 1, 2 and 3, in terms of what accuracy level is reached within a given deadline and how long it takes to reach a certain accuracy level. Finally, Section 5.6.6 provides the total energy consumed by the agents within a given deadline and to reach a certain accuracy level. We present all results as an average of 70 independent simulations and the source code generating all data is available in [114]. For the sake of presentation, we also include a short summary of the key result observed for each analysed scenario.

5.6.1. SCENARIO 0: PURE LEARNING PERSPECTIVE

To study the behaviour of the policies from a purely learning perspective, we consider the scenario where all agents have the same bit rate, which is equal to the average achievable bit rate in the considered wireless environment. Hence, the `max-sum-loss`, `max-sum-dev` and `max-sum-rate` policies behave like the `max-loss`, `max-dev` and `random` policies, respectively. Because of the identical bit rates, the number of selected agents per communication round is constant and the same for all policies, even for the `max-loss` policy that requires extra time for the loss calculations. We set $K_v = 300$ samples at each agent and hence, the training time $C_T = 1.02$ s, excluding the time for the loss calculation for the `max-loss` policy. Furthermore, we set $\tau_{\mathrm{APP,MAX}} = 5s$, which allows approximately $4s$

of uploading time.

IID Data

Figure 5.3 shows the increase of the accuracy over time and illustrates that the `max-loss` policy has a slower convergence than the `max-dev` and `random` policies. The slower convergence of the `max-loss` policy is explained by its persistence to select the same agents over time while the `max-dev` and `random` policies tend to more evenly cover the entire agent population over time. For IID data, all agents have samples from all ten classes and hence, regardless of the selected agents in a given communication round, the loss change of all agents in that communication round will be similar. Consequently, the agent sorting of the `max-loss` policy at the beginning of each communication round, does not change significantly over time and results in frequently selecting the same agents. Therefore, the global model is mostly trained on a subset of the total available samples which leads to a slower convergence.

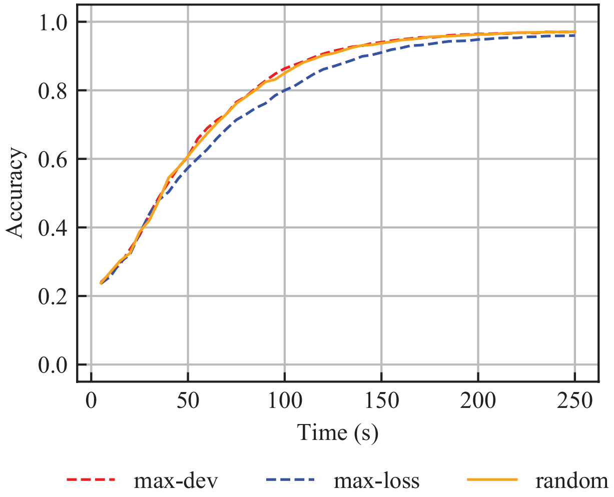


Figure 5.3: Accuracy over time for IID data, when agents have identical bit rates, $K_v = 300$ samples and $\tau_{\text{APP,MAX}} = 5s$.

When an agent is selected for training, its deviation will be relatively small and for every round that the agent is not selected, its deviation will be relatively large. Hence, the `max-dev` policy behaves in a round robin fashion, with some initial agent sorting. With this, an even agent selection is achieved, which allows to consistently train on all available samples. Furthermore, Figure 5.3 shows that the `max-dev` and the `random` policies perform similarly because the `random` policy also tends to cover the agent population well and hence trains the model on all samples. Therefore, we have the following important result for the case of the IID data:

Result 5.1 *From a purely learning perspective, for scenarios with IID data, the learning process benefits from evenly selecting the agents over time and hence the* `max-dev` *and* `random` *policies tend to outperform the* `max-loss` *policy.*

Non-IID Data

When non-IID data are considered, the agents have samples from only two classes and therefore, the selection of agents in a given communication round is more crucial than for IID data. Also, a larger number of communication rounds is needed to reach a given accuracy level compared to the scenario with IID data. Figure 5.4 illustrates that the for non-IID data, the `max-loss` policy, outperforms the `random` policy and provides better convergence than the `max-dev` policy, in the sense that accuracy fluctuate with the `max-dev` policy. In contrast to IID data, for non-IID data, the losses of the agents after a given communication round will differ depending on the selected agents. Consequently, the `max-loss` policy does not persistently select the same agents. However, it does select some specific agents more times than others, which allows to train more on samples that could have a bigger benefit to the learning process. This result highlights that not all agents are equally important to the learning process when the data are non-IID.

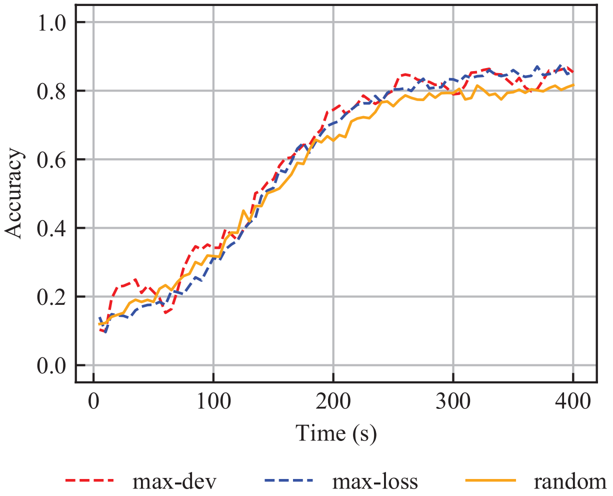


Figure 5.4: Accuracy over time for non-IID data, when agents have identical bit rates, $K_v = 300$ samples and $\tau_{\text{APP,MAX}} = 5s$.

Figure 5.4 also shows that the accuracy with the `max-dev` policy fluctuates over time, where the period of the fluctuation is equal to the time needed to select all agents once, i.e. the round robin period. Because agents do not have equally important data, the initial sorting of the deviations $\delta_v[i]$, is essentially based on the data importance of the agents. Then, the agents are selected the same number of times and in sequence, which can harm the accuracy and lead to fluctuations. The accuracy fluctuation is amplified by selecting the same number of agents per communication round. Finally, Figure 5.4 shows that the random policy has the worst performance because it does not consider any learning metrics. Therefore, the key takeaway result for the case of the non-IID data is the following:

Result 5.2 *From a pure learning perspective, for scenarios with non-IID data, not all agents have equally important data. Hence, the max-loss policy provides the highest accuracy level and stable gains by selecting the most appropriate agents per communication round.*

5.6.2. Scenario 1: Learning and Communication Perspective

In this scenario, the agents have distinct bit rates, based on the communication model in (5.4), with time-varying wireless channels that vary at each communication round. Similarly to Scenario 0, we assume $K_v = 300$ samples at each agent and set $\tau_{\text{APP,MAX}} = 5s$.

IID Data

Figure 5.5 shows the accuracy of the considered policies over time and illustrates that all considered policies, apart from the max-loss policy, perform similarly. The slower convergence of the max-loss policy is due to the uneven agent selection, as explained in Scenario 0. Even though the max-sum-loss policy also relies on the loss of the agents, it takes into account the bit rates of the agents which eventually leads to selecting different agents per round and consequently achieving a higher accuracy level than the channel-oblivious max-loss policy. Figure 5.5 also shows that the channel-aware policies, i.e. max-sum-loss, max-sum-dev and max-sum-rate, perform similarly to the channel-oblivious max-dev and random policies. This similarity exists despite the fact that the former policies can select more agents than the latter. Therefore, we can conclude that there are no significant gains from exploiting the wireless channels. For the remaining Scenarios 2 and 3, we will not consider IID data. Therefore, the takeaway result is:

Result 5.3 *For IID data, the exact agent selection policy is not crucial as long as different agents are selected over time. Additionally, the gains of channel-aware agent selection are minimal.*

Non-IID Data

Figure 5.6 shows the accuracy of the policies over time and illustrates that the max-loss policy performs better than the max-dev and random policies, as expected from Result 5.2. Figure 5.6 also shows that the max-sum-loss policy performs better than the

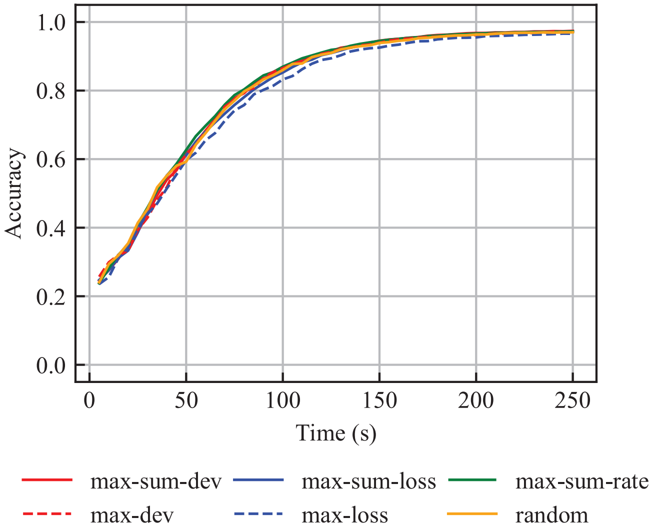


Figure 5.5: Accuracy over time for IID data, considering time-varying wireless channels, $K_v = 300$ samples and $\tau_{\text{APP,MAX}} = 5s$.

max-loss policy because it also exploits the gains from the wireless channels, which leads to selecting more agents per communication round. For the same reason, the max-sum-dev policy performs better than the max-dev policy. The max-sum-loss and max-sum-dev policies behave similarly and achieve a higher accuracy than the rest of the policies, throughout the studied time period. Thus, we can conclude that agent selection based on both channel and learning aspects is beneficial to the learning process, regardless of the learning metric considered, i.e. the deviations $\delta_v[i]$ or the loss $F(\mathbf{W}_{G,v}[i])$.

Additionally, Figure 5.6 shows that the max-sum-rate and max-loss policies perform similarly, even though the max-sum-rate policy selects on average about double the number of agents per round than the max-loss policy. This result highlights the effectiveness of the loss $F(\mathbf{W}_{G,v}[i])$ as a metric to indicate the importance of an agent in the learning process. Another observation from Figures 5.4 and 5.6 is that the channel-oblivious policies, i.e. max-loss, max-dev and random, behave similarly in both scenarios. However, the accuracy with the max-dev policy in Figure 5.6 does not fluctuate as it did in Figure 5.4, which is due to the fact that the wireless channel variation impacts the number of selected agents per communication round. This leads to the averaging of the peaks that were observed in Figure 5.4. Therefore, the important message is:

Result 5.4 *Choosing agents based on both channel and learning aspects is advantageous for the learning process in non-IID data scenarios. The learning aspect ensures the selection of agents with suitable data, while the channel aspect benefits from the wireless*

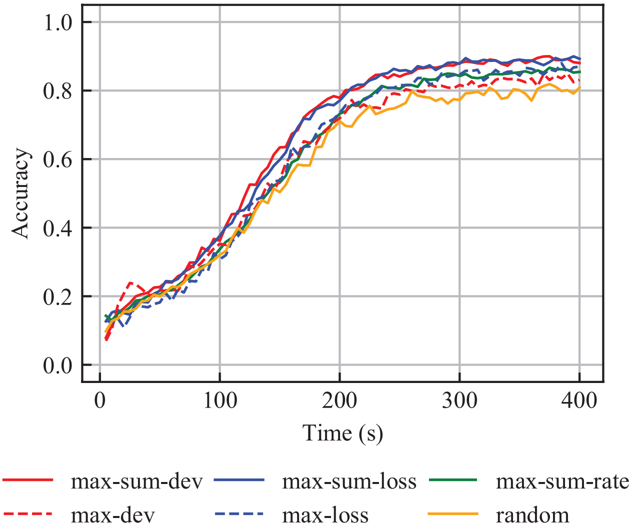


Figure 5.6: Accuracy over time for non-IID data, considering time-varying wireless channels, $K_v = 300$ samples and $\tau_{\text{APP,MAX}} = 5s$.

channels, thus enabling the selection of as many as possible agents per communication round.

5.6.3. Scenario 2: Different Number of Samples

In this scenario, we continue to have varying bit rates, while only considering non-IID data and reducing the number of samples per agent from $K_v = 300$ to $K_v = 100$. Because the training time C_T reduces with the number of samples K_v, for comparison reasons, we adjust the application-specific latency budget to $\tau_{\text{APP,MAX}} = 4.3s$. With this, we ensure that the time interval for uploading the FL models is the same as in Scenarios 0 and 1.

Figure 5.7 shows the accuracy of the policies over time and in comparison to Scenario 1, it now takes a longer time for the accuracy to reach a more stable level, because the agents now hold less data. Moreover, Figure 5.7 shows that during the initial learning phase (until 400s), the `max-loss` policy learns quicker than the rest of the policies. The good performance of the `max-loss` policy is a result of selecting the most appropriate agents for the learning, which is more crucial in this scenario, since given that the agents have fewer samples, the likelihood of an agent holding non-beneficial data for the learning process is higher. During this initial learning phase, the `max-sum-loss` and `max-sum-dev` policies perform worse than the `max-loss` policy because they sacrifice agents that are important to the learning process for agents with high bit rates. After 400s, the `max-sum-loss` and `max-sum-dev` policies perform similarly to the `max-loss`

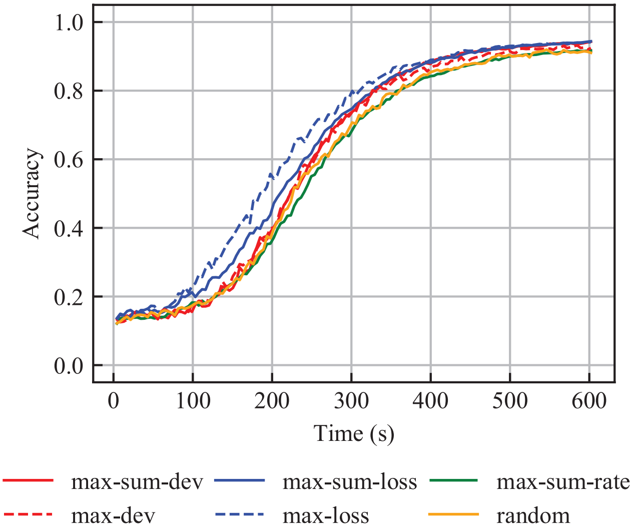


Figure 5.7: Accuracy over time for non-IID data, considering time-varying wireless channels, $K_v = 100$ samples and $\tau_{\text{APP,MAX}} = 4.3s$.

policy because they selected enough agents with important data over time.

Moreover, Figure 5.7 shows that the `max-sum-rate` policy under-performs throughout the learning process, even though it is the policy selecting on average the most agents per communication round. This poor performance is attributed to not taking into account the learning aspect, which is dominant in this scenario. We therefore have the following takeaway message:

Result 5.5 *When agents have a small data set size in a non-IID setting, the agent selection becomes very important, especially during the initial learning phase. For this reason, the `max-loss` policy provides higher accuracy during the initial learning phase than the other channel- and learning- aware policies.*

5.6.4. SCENARIO 3: DIFFERENT LATENCY BUDGET

To investigate the impact of the application-specific latency budget $\tau_{\text{APP,MAX}}$ on the accuracy, we consider a scenario with non-IID data, $K_v = 300$ samples and $\tau_{\text{APP,MAX}} = 2s$, instead of $\tau_{\text{APP,MAX}} = 5s$ that was considered in Scenarios 0 and 1. The reduction of $\tau_{\text{APP,MAX}}$ limits the number of agents that can be selected in a communication round as well as the set of agents that can be selected. The reason is that the cell edge agents, who suffer from low bit rates, may only sporadically be able to transmit their local model within the latency budget $\tau_{\text{APP,MAX}}$. Therefore, lower accuracy levels are expected within a given time period, compared to Scenario 1.

Figure 5.8 shows the accuracy of the policies over time, which fluctuate more than in Scenario 1 because the global model is updated more frequently. Specifically, within 400s, 80 and 200 communication rounds are executed in Scenario 1 and 3, respectively. Moreover, Figure 5.8 shows that the initial learning phase in this scenario lasts for about 100s while in Scenario 1, it lasts for about 200s, as a result of setting a different latency budget $\tau_{\text{APP,MAX}}$. However, in both scenarios, the initial learning phase lasts for a comparable number of communication rounds.

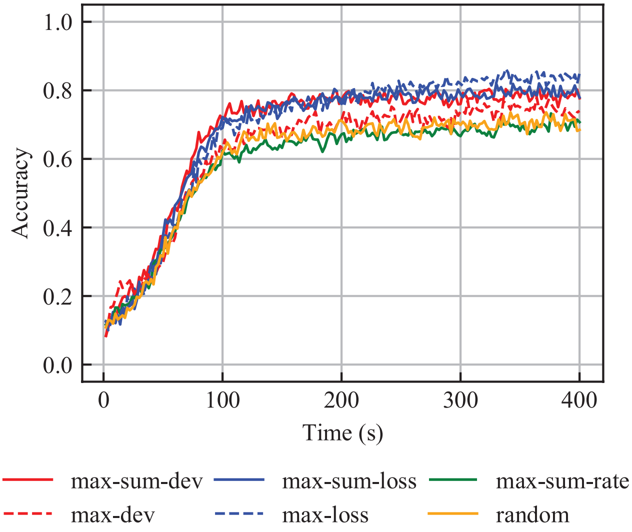


Figure 5.8: Accuracy over time for non-IID data, considering time-varying wireless channels, $K_v = 300$ samples and $\tau_{\text{APP,MAX}} = 2s$.

Another observation from Figure 5.8 is that the performance of the `max-sum-loss` and `max-sum-dev` policies is better than the performance of the `max-loss` policy until 200s, when the `max-loss` policy becomes the best performing policy. The reason is that the reduction of the latency budget $\tau_{\text{APP,MAX}}$ limits the extra number of agents that the channel-aware policies can select compared to the channel-oblivious policies. Hence, until 200s, there are some gains from exploiting the wireless channel but after 200s the accuracy with the channel-aware policies does not improve further, because the channel-aware policies avoid selecting agents with poor bit rates. Due to this reason, the channel-oblivious `max-loss` policy can converge to a higher accuracy level in the long term. This implies that it selects agents at the cell edge more often than the channel-aware policies. For the same reason, the `max-sum-rate` policy under-performs, thus making it even slightly worse than the `random` policy. Overall, the key message from the analysis of Scenario 3 is:

Result 5.6 *A short latency budget $\tau_{APP,MAX}$ in a non-IID setting limits the gains of the channel-aware policies and in the long term, the `max-loss` policy can provide a higher accuracy because it selects agents with persistently poor bit rates that have beneficial data for the learning process.*

5.6.5. Scenario Comparisons

The policies in Scenarios 1-3 can be compared in terms of what accuracy levels they have reached after a given deadline as well as in terms of how much time is needed to reach a certain accuracy level.

Deadline

Considering that some applications may require the training to be completed within a given deadline, we compare the policies over the three scenarios after a deadline of 300s, i.e., 5 minutes. Figure 5.9 shows the accuracy for every policy and scenario around the 300s deadline, while Table 5.1 shows the measured accuracy level, which is derived by averaging the accuracy over a 30s period, therefore from 270s to 300s. The averaging of the accuracy in Table 5.1 is performed to ensure that the provided results are not dominated by the accuracy fluctuations. From both Figure 5.9 and Table 5.1, it is observed that each policy reaches a higher accuracy level in Scenario 1 than in Scenarios 2 and 3. The policies in Scenario 1 perform better than in Scenario 2, because the agents do not suffer from a small data set size. In Scenario 1 the policies perform better than in Scenario 3, because the larger latency budget $\tau_{\text{APP,MAX}}$ allows to select more agents in a given communication round. Additionally, Figure 5.9 shows that even though the accuracy of the policies in Scenarios 1 and 3 are roughly stable, the accuracy of the policies in Scenario 2 is still sharply increasing because more communication rounds are needed to reach convergence when the agents have a small data set.

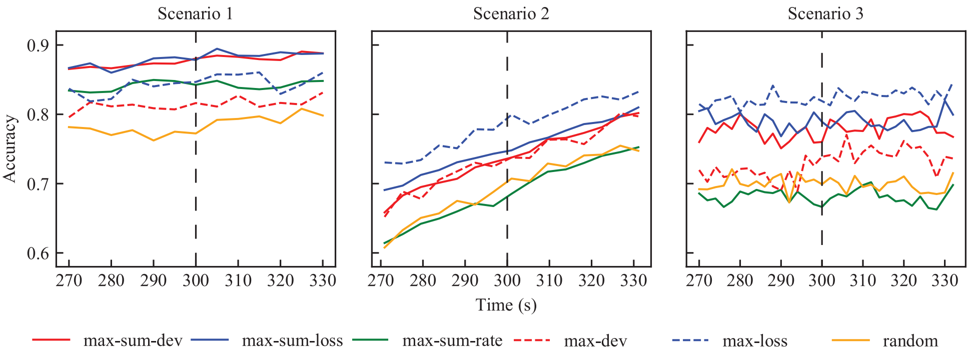


Figure 5.9: Accuracy of the considered policies for non-IID data around the time intervals of interest, where the dashed line indicates the 300s deadline.

Table 5.1: Accuracy level reached for every policy after 300s for each scenario, where the highest accuracy per scenario is marked in bold.

Policy	Scenario 1	Scenario 2	Scenario 3
`max-sum-dev`	**0.87**	0.72	0.77
`max-sum-loss`	**0.87**	0.73	0.79
`max-sum-rate`	0.84	0.66	0.68
`max-dev`	0.81	0.71	0.71
`max-loss`	0.84	**0.76**	**0.82**
`random`	0.77	0.68	0.70

Table 5.1 also shows that among the channel-oblivious policies, the `max-loss` policy converges to a higher accuracy level, as also concluded in Result 5.2. Moreover, the `max-sum-dev` and `max-sum-loss` policies converge to approximately the same accuracy level, regardless of the scenario and they provide the highest accuracy in Scenario 1, as explained in Result 5.4. However, in Scenario 2, where the agents have limited samples and in Scenario 3, where the latency-budget $\tau_{\text{APP,MAX}}$ is short, the highest accuracy level is provided by the `max-loss` policy, for the reasons provided in Results 5.5 and 5.6, respectively.

ACCURACY TARGET

Some applications require to train the global model until a specific accuracy target is met. Therefore, we compare the policies in the three scenarios in terms of how much time is needed to reach the 75%, 80% and 85% accuracy levels. We consider that an accuracy level is reached if the average accuracy over a period of 30s is above the accuracy target. Table 5.2 shows the time in seconds to reach each accuracy level, where a hyphen indicates that the accuracy level could not be reached within the simulated 400s while the values in parenthesis under Scenario 2 indicate that the accuracy level is measured after 400s.

Table 5.2 shows that the accuracy levels are reached faster in Scenario 1 than in Scenario 2 because agents have more samples in Scenario 1 and hence, the FL server can train on more samples in a given time period. Table 5.2 also shows that when the latency budget $\tau_{\text{APP,MAX}}$ is set to a small value, i.e. in Scenario 3, the `max-sum-dev`, `max-sum-loss` and `max-loss` policies reach the 75% accuracy level faster than when $\tau_{\text{APP,MAX}}$ is set to a larger value, i.e. in Scenario 1. This is because in Scenario 3 more communication rounds are performed within a given time interval than in Scenario 1. However, in Scenario 1, higher accuracy levels can be achieved within the 400s time interval compared to Scenario 3 because more agents can consistently contribute to the learning process. For example, in Scenario 1, the `max-sum-dev` policy can reach the 85%

Table 5.2: Time, in seconds, needed to reach the 75%, 80% and 85% accuracy levels for every policy in each scenario, where the shortest time per level and scenario is marked in bold.

Policy	Scenario 1			Scenario 2			Scenario 3		
	75%	80%	85%	75%	80%	85%	75%	80%	85%
`max-sum-dev`	**195**	**225**	270	318	344	378	**150**	-	-
`max-sum-loss`	200	**225**	**255**	314	340	374	156	358	-
`max-sum-rate`	225	250	355	340	370	(421)	-	-	-
`max-dev`	230	275	-	318	348	391	364	-	-
`max-loss`	220	250	315	**297**	**323**	**357**	168	**242**	-
`random`	260	380	-	335	370	(409)	-	-	-

accuracy level within 270s while in Scenario 3, none of the policies can reach the 85% accuracy level within 400s.

Moreover, Table 5.2 shows that the policies in Scenario 2 generally reach the 75% accuracy target at a later time compared to Scenarios 1 and 3. However, higher accuracy targets can be achieved in Scenario 2 than in Scenario 3, within the 400s time period. This observation is also illustrated in Figure 5.10, as the accuracy curves in Scenario 2 are still in an increasing phase while in Scenario 3 they are fairly constant, which shows that the accuracy will not further improve significantly. This result highlights that the more agents can participate in the learning process, even if those agents have a small data set, the higher accuracy levels can be achieved within a given long-term time period, because more diverse data are used for the training. Therefore, the comparison among scenarios has the following important messages:

Result 5.7 *When the latency-budget $\tau_{APP,MAX}$ is set to a small value, the policies initially learn faster than when the latency-budget $\tau_{APP,MAX}$ is large. However, in the long term, a higher accuracy is achieved with a large latency-budget $\tau_{APP,MAX}$.*

Result 5.8 *Regardless of the considered scenario, the more agents with diverse data are selected, the higher the accuracy level that can be achieved.*

5.6.6. Energy Considerations

In this section, we analyse the aggregated energy consumption of the agents for every policy and scenario. Figure 5.11 shows the energy consumption in Joules after a time interval of 300s and it illustrates that the total energy consumption is dominated by the training, rather than the transmissions. Therefore, the total energy consumption depends primarily on the aggregated number of agents selected within the given time in-

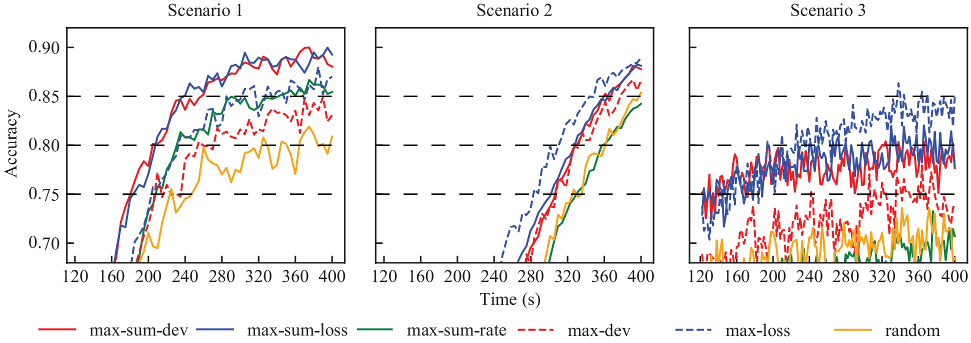


Figure 5.10: Accuracy of the considered policies for non-IID data around the accuracy levels of interest, where the dashed lines indicate the 75%, 80% and 85% accuracy levels.

terval. Consequently, the energy consumption is higher when channel-aware policies are applied, regardless of the scenario, as such policies generally end up selecting more agents.

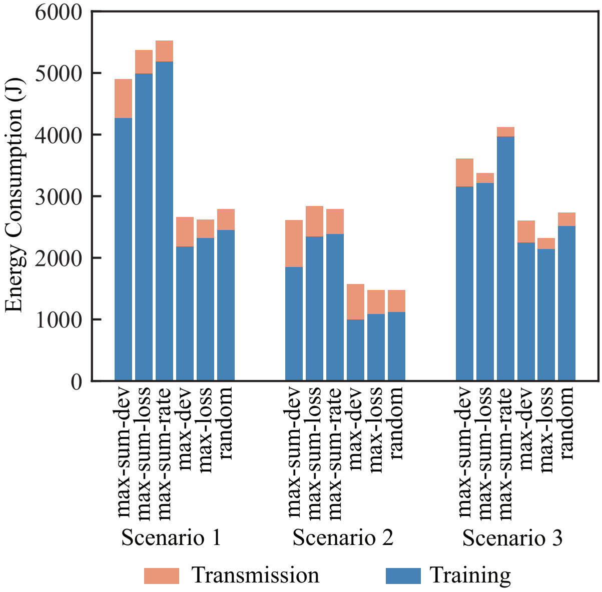


Figure 5.11: Aggregated energy consumption of agents for every policy and scenario after a time period of 300s.

Figure 5.11 also shows that in Scenario 3, the ratio of the training energy consumption between the channel-aware and channel-oblivious policies is smaller than in Scenarios 1 and 2, because the channel-aware policies select fewer agents compared to Scenarios 1 and 2, due to the shorter latency budget $\tau_{\text{APP,MAX}}$. Despite more rounds occurring within the 300s interval, the total number of selected agents is lower than in Scenario 1. Consequently, both transmission and training energy consumption are reduced in Scenario 3 compared to Scenario 1. Moreover, the total energy consumption in Scenario 2 is lower than in Scenarios 1 and 3, which is due to the agents having fewer samples and consequently, shorter training times. However, because of the shorter training times, more communication rounds are performed during the 300s time interval compared to Scenario 1. In addition, Scenarios 1 and 2 have a similar number of agents selected per communication round, implying that the transmission related energy consumption is higher in Scenario 2 than in Scenario 1.

Table 5.1 shows that in Scenario 1, the `max-sum-dev` and `max-sum-loss` policies provide the highest accuracy, whereas Figure 5.11 shows that their energy consumption is high. However, the `max-loss` policy achieves a slightly lower accuracy than the two above-mentioned policies while consuming about half the amount of energy. Therefore, the `max-loss` policy provides a good trade-off between accuracy and energy consumption. Additionally, the `max-loss` policy is the best in terms of both energy and accuracy in Scenarios 2 and 3 because it can achieve the highest accuracy while selecting few agents.

Figure 5.12 shows the total energy consumption of the agents per policy and scenario, until the time that the 80% accuracy level is reached. The absence of a bar in Figure 5.12 implies that the 80% accuracy level was not reached within the simulated 400s. Comparing the energy consumption in Figure 5.12 to the accuracy levels in Table 5.2, a trade-off between time and energy is observed in Scenario 1. Specifically, the `max-sum-dev` and `max-sum-loss` policies reach the 80% accuracy level the fastest. However, both policies have a higher energy consumption than the `max-loss` policy, which reaches the 80% accuracy level 25s later. Another observation is that even though the policies in Scenario 2 take longer to reach the 80% accuracy level compared to Scenario 1, they consume less energy. Overall, our takeaway message from the energy impact on the system is:

Result 5.9 *The `max-loss` policy provides a good balance between achieving high accuracy levels fairly quickly and consuming less energy due to selecting fewer agents per communication round.*

5.7. Concluding Remarks

This chapter has investigated the agent selection problem for FL in wireless communication environments. We proposed a generic optimisation problem, which can be adapted to a range of applications, depending on the needs and capabilities of the network and the agents. We focused on the important subproblem of latency-constrained networks,

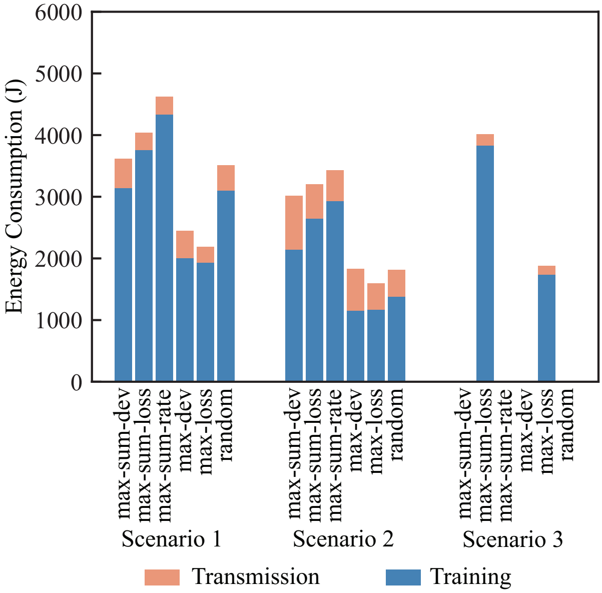


Figure 5.12: Aggregated energy consumption of agents for every policy and scenario to reach an accuracy level of 80%.

which is a 0/1 Knapsack problem. We obtained its solution with a pseudo-polynomial algorithm with low complexity. Extensive simulations showed that the loss is a very good metric to describe the importance of an agent in the learning process. Additionally, we showed that the policies derived from the optimisation problem perform better than only channel-aware and only learning-aware policies. Moreover, we showed that learning-based policies performed well when the agents have few samples and when the wireless channel could not be largely exploited due to short latency budgets.

6

Joint Vehicle Selection and Resource Allocation with Beamforming

Many algorithms related to vehicular applications, such as enhanced perception of environment, benefit from frequent updating and using data from multiple vehicles. Hence, FL is a promising method to improve the accuracy of algorithms in the context of vehicular networks. In the previous chapter, a general agent selection framework was introduced, which selects agents based on a metric characterising each agent's importance with respect to both the learning process and the resource efficiency of its wireless communication channel. In this chapter, we extend the previously proposed framework to address the joint agent selection and resource allocation problem in vehicular wireless networks, considering multi-cell networks with MU-MIMO capable BSs. We approximate the solution of the defined optimisation problem with the proposed vehicle-beam-iterative (VBI) algorithm. For the evaluation of the VBI algorithm we perform extensive simulations with realistic road and mobility models, for the task of object classification of European traffic signs. The results indicate that MU-MIMO improves the convergence time of the global model. Furthermore, the application-specific accuracy targets are reached faster in scenarios where the vehicles have the same training data set sizes than in scenarios where the data set sizes are different.

This chapter is based on a report [115].

6.1. Introduction

In recent years, multiple advances have been made regarding autonomous vehicles. Autonomous vehicles rely on information they receive from sensors, as well as other vehicles and the network, through vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) wireless links, respectively, to make decisions for route planning, speed adjustment and collision avoidance, among others [116]. To ensure driving safety, driving decisions should be accurate and the communication via the V2V and V2I links should be fast and reliable. To address these challenges, ML algorithms are widely applied. Examples of ML applications include optimal radio resource assignment and optimal handover [116].

When driving, the environment is very dynamic and it can change drastically over time. Additionally, driving should be adjusted based on the location/area and the enforced driving rules. Hence, the applied ML algorithms should constantly update based on new sensor data. The ML algorithms can be further improved by using data from other vehicles. For example, vehicles can share their camera data to enhance the perception of environment, which then allows vehicles to observe obstacles or dangerous situations that are out of the reach of their cameras, yet in close proximity [116]. Due to the distributed nature of the data, FL is a promising method for collaborative learning in vehicular networks and it is shown to accelerate the learning rate compared to traditional centralised ML methods [117]. A range of applications has been discussed in literature that can benefit from FL in vehicular networks, including traffic prediction for resource management, co-operative perception of environment [85, 118] and steering wheel angle precision [119].

In Chapter 5, the agent selection problem for FL in resource-constrained wireless networks has been addressed, by providing an agent selection framework based on the agent characteristics while also considering an application-specific latency budget. In this chapter, the previously proposed framework is extended to address the joint agent selection and resource allocation problem in vehicular wireless networks. Moreover, in this chapter, road and mobility models are considered as well as a multi-cell network with MU-MIMO capable BSs.

The joint agent selection and resource allocation problem for wireless networks is addressed in literature with mostly considering stationary agents and not in the context of vehicular networks. For example, Chen et al. [112] address the minimisation of the training loss while considering parameters related to the wireless channels, whereas Zeng et al. [120] concentrate on minimising the energy consumption. Fan et al. [121] claim to have the first work that addresses the minimisation of the time duration of each communication round while also considering a practical mobility model. However, their evaluation does not consider a learning task related to vehicular networks nor they consider

MU-MIMO capable base stations. One of the few works that addresses FL in vehicular networks, is by Deveaux et al. [122]. Specifically, they highlight the need for algorithms addressing the unevenly distributed data and propose a high-level protocol that allows the network to retrieve information of what kind of data each vehicle has.

The main contributions of this chapter are the following:

- We extend the agent selection framework from Chapter 5 to perform joint vehicle selection and resource allocation in latency-constrained vehicular wireless networks, considering MU-MIMO capable BSs. We then propose the VBI algorithm to approximate the solution of the defined optimisation problem.
- We perform evaluations in realistic vehicular scenarios, based on road and mobility models from 3GPP. Moreover, we consider the learning task of object classification on the ETSD, which is a relevant data set for vehicular applications.
- We show that MU-MIMO capable BSs improve the convergence time of the global model because they allow the selection of multiple vehicles on the same time-frequency resources and improve the quality of the wireless channels. Specifically, when vehicles have the same data set sizes, the 85% accuracy level is reached within 2.3 seconds, whereas more than 4 seconds were needed in any scenario considered in Chapter 5.
- We show that the local loss is a good agent selection metric for scenarios with non-IID data, assuming that all vehicles have the same data set sizes. When vehicles have different data set sizes, the loss-based policies do not provide any gains.
- We demonstrate that the convergence time in scenarios where vehicles have different data set sizes is longer than in scenarios where vehicles have the same data set sizes.

The rest of the chapter is organised as follows. Section 6.2 provides the system model. In Section 6.3, the joint vehicle selection and resource allocation optimisation problem is derived and the VBI algorithm is introduced. Section 6.4 provides the configuration of the considered scenarios and in Section 6.5 the evaluation of the VBI algorithm is performed, in the considered scenarios. Finally, the conclusions are given in Section 6.6.

6.2. SYSTEM MODEL

Consider a cellular network with one FL server, a set $\mathcal{M}$ of BSs, where $M = |\mathcal{M}|$ is the number of BS in the network, and a set $\mathcal{V}$ of vehicles, where $V = |\mathcal{V}|$ is the number of vehicles in the network. The vehicles and the FL server collaboratively train a global model, without requiring the transmission of the data sets gathered by the vehicles, and the BSs facilitate the communication between the vehicles and the FL server. For that, we assume that the FL server is connected to all BSs with fiber-optic cables, hence their communication latency is negligibly small and that the communication between between the FL server and the BSs is synchronised. Figure 6.1 illustrates the considered system model for a simple example network with $M = 2$ BSs and $V = 3$ vehicles. Moreover, and similarly to Chapter 5, each vehicle $v \in \mathcal{V}$ holds its own training data set $\mathcal{K}_v$ and testing data set $\mathcal{K}_{T,v}$, where $K_v = |\mathcal{K}_v|$ and $K_{T,v} = |\mathcal{K}_{T,v}|$ denote the number of training and testing data samples available at vehicle v, respectively.

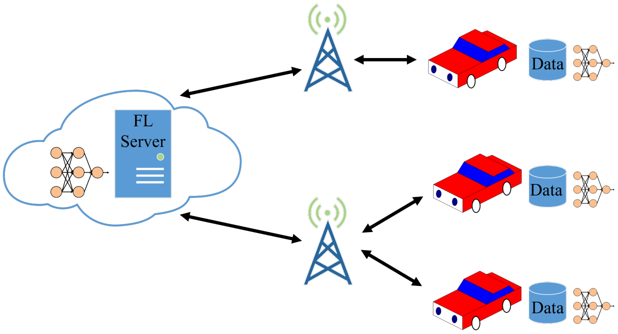


Figure 6.1: A simplified system model for FL in vehicular networks.

Recall the schematic overview of a communication round i from Figure 5.1 in Chapter 5. In this Chapter, we consider the same schematic overview with the only difference that the FL server is now connected to multiple BSs, as also shown in Figure 6.1. Also recall from Chapter 5 that the set $\mathcal{V}_G[i]$ denotes the vehicles participating in the learning process, $\tau_{\mathrm{T},v}$ and $\tau_{\mathrm{UL},v}$ denote the training time and uploading time of vehicle v, respectively, and $\tau_{\mathrm{APP,MAX}}$ is the application-specific deadline. In contrast to Chapter 5, we now also consider the broadcast time in the DL channel, denoted as τ_{DL}, which is network-dependent and fixed for every communication round. It is typically set such that a minimum bit rate is ensured at the cell edge, as it will be explained in detail later on. Therefore, and with using (5.1), in this chapter, we perform vehicle selection and resource allocation over a time period

$$\tau_{\mathrm{T+UL}} \leq \tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}, \tag{6.1}$$

at every communication round. In other words, at every communication round, the selected vehicles train and upload their local model within the time interval $\tau_{\mathrm{T+UL}}$.

In reality, during the time interval $\tau_{\mathrm{T+UL}}$, there are multiple scheduling time slots, and hence multiple resource allocation moments, occurring on a ms-level. The scheduling decision at every time slot depends on the experienced SNR of the vehicles, which vary on a ms-time scale due to mobility. In this chapter, we focus on the resource allocation problem from a higher time scale perspective. Hence, we perform periodic resource allocation over a period $\tau_{\text{T+UL}}$ and we assume that the effects occurring on the ms-level, e.g. multipath fading, can be averaged.

Regarding the learning model, we consider the same model as presented in Section 5.2.1. The remainder of this section presents the communication model and the antenna model.

6.2.1. Communication Model

We consider that the BSs and the vehicles are equipped with beamforming antenna arrays to form narrow and strong beams. Specifically, beam pairs are formed between the BSs and the vehicles and the same beam pair is used for both the UL and DL transmissions [33], as also explained in Section 2.6.3. Moreover, we assume that the BS and vehicle beams are directly pointing at each other and hence interference between different transmissions is neglected. For the BS antenna array, we assume a GoB mode, i.e. a BS $m \in \mathcal{M}$ can form a pre-defined set of beams $\mathcal{B}_m$ in the 3D space. We further assume that all BSs have the same set of beams and thus each BS m has $B_M = |\mathcal{B}_m|$ beams. Regarding the antenna array of the vehicles, we assume a single beam that can be steered in any direction. The detailed modelling of the antenna arrays is provided in Section 6.2.2.

Considering MU-MIMO capable BSs, transmissions from multiple vehicles, using different reception beams at the serving cell, can occur on the same frequency resources at a given time, as it was explained in Section 2.3.3. Additionally, beams from multiple vehicles can be paired to the same BS beam and assuming wideband transmissions, i.e. transmissions on all frequency resources in a beam, the related vehicles can perform their UL transmission to the same BS beam in different time intervals. Therefore, during communication round i, a vehicle $v \in \mathcal{V}_G[i]$ is assigned to beam $b \in \mathcal{B}_{\text{TOT}}$ (of BS $m \in \mathcal{M}$) for a fraction of the time period $\tau_{\text{T+UL}}$, where $\mathcal{B}_{\text{TOT}} = \cup_{m\in\mathcal{M}}\mathcal{B}_m$ is the set with all BS beams and $B_{\text{TOT}} = |\mathcal{B}_{\text{TOT}}| = B_M M$ is the total number of beams in the network. Moreover, we assume that during the period $\tau_{\text{T+UL}}$, vehicles stay connected to the same beam. An analysis is later on carried out to indicate under which range of latency budget $\tau_{\text{APP,MAX}}$ this assumption holds. From here onwards, for the sake of simplifying the notations, we omit the dependency to BS $m \in \mathcal{M}$ as it is implicitly captured via the beam $b \in \mathcal{B}_{\text{TOT}}$.

For the transmission of the local model, and as an input for deriving the periodic resource assignment, we estimate the bit rate R_{vb} of vehicle v from beam $b \in \mathcal{B}_{\text{TOT}}$ with

$$R_{vb} = B \min\left(\log_2\left(1 + 10^{\Gamma_{vb}/10}\right), 15\right), \tag{6.2}$$

where 15 bits/Hz/s is the target peak spectral efficiency in the UL channel in 5G [30]. Also, B denotes the system bandwidth in MHz and Γ_{vb} is the estimated UL SNR at vehicle v from beam b and it is given in dB by

$$\Gamma_{vb} = P_{\text{V,MAX}} + G_{V,vb} + G_{M,vb} + G_{T,vb} - P_{\text{NOISE}} - NF_b, \tag{6.3}$$

where $P_{\text{V,MAX}}$ is the maximum transmit power of the vehicles in dBm, $G_{V,vb}$ and $G_{M,vb}$ are the vehicle and BS antenna gains in dBi, respectively, $G_{T,vb}$ is the transmission gain between vehicle v and the BS that beam b belongs to in dB, P_{NOISE} is the thermal noise power in dBm and NF_b is the noise figure at the BS that beam b belongs to in dB. The transmission gain $G_{T,vb}$ is given by (5.5) when applying the distance d_{vb} between vehicle v and the BS that beam b belongs to. Because of the periodic nature of the resource assignment approach and the assumption that vehicles stay connected to the same beam during the period $\tau_{\text{T+UL}}$, the SNR Γ_{vb} is assumed to be constant during the period $\tau_{\text{T+UL}}$.

Finally, the time duration τ_{DL} of the broadcast is fixed as the broadcast bit rate is constant and set such that a minimum bit rate is ensured at the cell edge. Moreover, it is assumed that there are many vehicles in the network, spread in different directions, and therefore all BS beams are used in the broadcast. For the derivation of the broadcast bit rate, the network layout and antenna array configuration are taken into consideration, which are given along with the broadcast time τ_{DL} in Section 6.4.3.

6.2.2. Antenna Model

In this chapter, we assume for both the BSs and the vehicles, uniform planar rectangular array (UPRA) with a total of $N_{E,k}$ antenna elements, where $k = M$ for the BS antenna arrays and $k = V$ for the vehicle antenna arrays. The antenna elements are positioned at $d_E = 0.5\lambda$ spacing in both the horizontal and vertical plane, where λ is the wavelength. We assume that each antenna element has an omnidirectional radiation pattern, thus the gain per antenna element $G_E = 0$ dBi, and the number of antenna elements in the horizontal and vertical plane are equal.

As previously mentioned, the antenna array at each BSs is configured in the GoB mode. We approximate the beams formed by the vehicles and all beams in the GoB, regardless of their direction, to have one continuous uniform side lobe. Moreover, all beams in the GoB are assumed to have the same HPBW. Additionally, given the symmetry of the antenna arrays, the HPBW $\Delta\varphi_k$ and the FNBW $\varphi_{0,k}$ in the azimuth and elevation planes are assumed to be equal and they are given by [32, 123]

$$\Delta\varphi_k \approx 2\left[\frac{\pi}{2} - \cos^{-1}\left(\frac{1.391\lambda}{\pi d_E\sqrt{N_{E,k}}}\right)\right] \approx \sqrt{\frac{3}{N_{E,k}}}, \tag{6.4}$$

$$\varphi_{0,k} = 2\left[\frac{\pi}{2} - \cos^{-1}\left(\frac{\lambda}{d_E\sqrt{N_{E,k}}}\right)\right]. \tag{6.5}$$

The number of beams in the GoB, is given by $B_M = B_{A,M}B_{E,M}$, where $B_{A,M}$ and $B_{E,M}$ denote the number of beams in the azimuth and elevation planes, respectively, and the number of beams in the GoB is calculated from the FNBW $\varphi_{0,M}$. Specifically, setting the boresight direction of each beam at the null of its adjacent beam, the angular resolution $\phi_{B,M} = \frac{\varphi_{0,M}}{2}$, for both the azimuth and elevation planes. Thus, the number of beams in a given plane, i.e. $B_{A,M}$, $B_{E,M}$, is given by dividing the angular range of the antenna, e.g. for the azimuth plane the angular range is 120° for three-sectorised antennas, by the angular resolution $\phi_{B,M}$. The configuration of the UPRAs considered in this chapter, are given in Section 6.4.3.

The effective beam gains $G_{A,vb}$ and $G_{E,vb}$ experienced by vehicle v from beam b in the azimuth and elevation planes, respectively, are given by [73]:

$$\begin{aligned} G_{A,vb} &= -\min\left\{12\left(\frac{\varphi_{vb}}{\Delta\varphi_M}\right)^2, \mathrm{FBR}_M\right\} + G_{M,\mathrm{MAX}} \\ G_{E,vb} &= \max\left\{-12\left(\frac{\vartheta_{vb}}{\Delta\varphi_M}\right)^2, \mathrm{SLL}_M\right\}, \end{aligned} \tag{6.6}$$

where φ_{vb} is the angle off the boresight direction of beam b, ϑ_{vb} is the negative elevation angle relative to the direction of beam b, FBR_M is the front back ratio in dB, $G_{M,\mathrm{MAX}}$ is the maximum gain in dBi and SLL_M is the side lobe level in dB, relative to the maximum gain of the main lobe. The maximum beam gain $G_{M,\mathrm{MAX}}$ is given, in dBi, by

$$G_{M,\mathrm{MAX}} = 10\log(N_{E,M}) + G_E. \tag{6.7}$$

The side lobe gain $G_{S,M}$ of each beam for the UPRA is given, in dBi, by [123]

$$G_{S,M} = 10\log\left(\frac{\sqrt{N_{E,M}} - \frac{\sqrt{3}}{2\pi}N_{E,M}\sin\left(\frac{\sqrt{3}}{2\sqrt{N_{E,M}}}\right)}{\sqrt{N_{E,M}} - \frac{\sqrt{3}}{2\pi}\sin\left(\frac{\sqrt{3}}{2\sqrt{N_{E,M}}}\right)}\right), \tag{6.8}$$

and therefore the side lobe level is given, in dBi, by

$$\mathrm{SLL}_M = G_{M,\mathrm{MAX}} - G_{S,M}. \tag{6.9}$$

We further assume [123] that $\mathrm{SLL}_M = -\mathrm{FBR}_M$. The total BS beam gain at vehicle v from beam b is then given,in dB, by

$$G_{M,vb} = G_{A,vb} + G_{E,vb}. \tag{6.10}$$

For the beams formed at the vehicles we assume that they can be steered to the direction of the serving BS beam b. Therefore, the vehicle antenna gain $G_{V,vb}$ is equal to the maximum vehicle beam gain $G_{V,\mathrm{MAX}}$, given in dBi by

$$G_{V,vb} = G_{V,\mathrm{MAX}} = 10\log(N_{E,V}) + G_E. \tag{6.11}$$

6.3. Problem Formulation and Analysis

In real-world applications, vehicles are diverse in terms of their training data as well as processing capabilities to train their local model and wireless channel quality. In this section we define the vehicle importance metric to characterize the vehicles. We then present the latency considerations, which also depend on the processing capabilities and the wireless channel quality of the vehicles. We then combine the vehicle importance with the latency considerations to formulate the joint vehicle selection and resource allocation optimization problem. Finally, the VBI algorithm is introduced as an approximation to the optimisation problem.

To address the joint vehicle selection and resource allocation problem, we consider two optimisation parameters; one related to the vehicle selection and one related to the beam/resource allocation. Specifically, the optimisation parameter $\mathbf{s}[i]$ is a $V \times 1$ vector containing the selected vehicles for communication round i and $s_v[i] = 1$ when vehicle v is selected for training, and otherwise $s_v[i] = 0$. We define the $V \times B_{\mathrm{TOT}}$ optimisation matrix $\mathbf{A}[i]$, with $A_{vb}[i] \in \{0,1\}$, holding the beam associations between the selected vehicles and the BS beams. Assuming that all selected vehicles stay connected to one and the same beam during the time interval $\tau_{\mathrm{T+UL}}$, it must hold that

$$\mathbf{A}[i] \cdot \mathbb{1}_{B_{\mathrm{TOT}} \times 1} = \mathbf{s}[i], \tag{6.12}$$

where $\mathbb{1}_{B_{\mathrm{TOT}} \times 1}$ denotes the all ones $B_{\mathrm{TOT}} \times 1$ vector.

6.3.1. Vehicle Characterisation

To capture the diversity of the vehicles, we use the *vehicle importance* $q_{vb}[i]$, which is the metric governing the vehicle selection and resource allocation problem at communication round i. Similarly to Chapter 5, the vehicle importance $q_{vb}[i]$ captures the trade-off between the importance $q_{L,v}[i]$ of vehicle v in the learning process against its resource consumption on beam b.

Based on the results of Chapter 5, we characterise the importance $q_{L,v}[i]$ of vehicle v in the learning process in terms of the loss $F(\mathbf{W}_{G,v}[i])$. Moreover, this chapter focuses on the trade-offs between the learning and wireless communication performance measures. Therefore, for the resource consumption of vehicle v on beam b, we only consider the consumption $C_{R,vb}[i]$ of the transmission resources, which is given by (5.6), when applying the bit rate R_{vb} of vehicle v on beam b. Then, the importance $q_{vb}[i]$ of vehicle v on beam b at communication round i, is given by

$$q_{vb}[i] = \frac{F(\mathbf{W}_{G,v}[i])^{\rho}}{C^{1-\rho}_{R,vb}[i]}, \tag{6.13}$$

where $\rho \in [0,1]$ is a constant to tune the relative significance of the learning importance and the resource consumption. We further define the $V \times B_{\mathrm{TOT}}$ matrix $\mathbf{Q}[i] = [\mathbf{q}_1[i], \cdots, \mathbf{q}_{B_{\mathrm{TOT}}}[i]]$, where $\mathbf{q}_b[i]$ is a column vector holding the importance $q_{vb}[i]$ of each vehicle v on beam b.

6.3.2. Latency Considerations

In Chapter 5, we considered a fixed amount of available UL transmission resources $C_{\mathrm{R,MAX}}$, which was expressed as a product of the bandwidth and the upload time. Because we assume wideband scheduling, for the sake of simplicity, in this chapter, we express the transmission resources $C_{\mathrm{R,MAX}}$ only in terms of time. Therefore, the selected vehicles should perform their UL transmission within the available transmission resources $C_{\mathrm{R,MAX}}$, where the UL transmission time $\tau_{\mathrm{UL},vb}[i]$ of vehicle v on beam b at communication round i is given by

$$\tau_{\mathrm{UL},vb}[i] = \frac{Z}{R_{vb}[i]}, \tag{6.14}$$

where Z is the size of the model in MB.

The vehicles selected to participate in communication round i, should train and upload their local model within a time interval $\tau_{\mathrm{T+UL}} = \tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}$. When the vehicles have different training times $\tau_{\mathrm{T},v}$, as defined in (5.7), the local training time duration of each vehicle should be taken into consideration. Specifically, each BS beam becomes active, i.e., receives UL data, for the first time when the vehicle with the shortest training time that is assigned to that beam finishes its local training. To that end, we define the column vector $\boldsymbol{\tau}_{\mathbf{L}} = [\tau_{\mathrm{L},1}, \cdots, \tau_{\mathrm{L},B_{\mathrm{TOT}}}]^{\intercal}$ to indicate the start time of the UL transmissions per beam. Also, we introduce the $V \times B_{\mathrm{TOT}}$ auxiliary matrix $\hat{\mathbf{T}}_{\mathbf{L}} = [\hat{\boldsymbol{\tau}}_{\mathbf{L},1}, \cdots, \hat{\boldsymbol{\tau}}_{\mathbf{L},B_{\mathrm{TOT}}}]$ to associate the training time $\tau_{\mathrm{T},v}$ of each vehicle v to its assigned beam, where $\hat{\boldsymbol{\tau}}_{\mathbf{L},b}$ is a $V \times 1$ vector indicating the training times of the vehicles assigned to beam b

$$\hat{\mathbf{T}}_{\mathbf{L}}[i] = (\boldsymbol{\tau}_{\mathbf{T}} \cdot \mathbb{1}_{1\times B_{\mathrm{TOT}}}) \circ \mathbf{A}[i], \tag{6.15}$$

where $\boldsymbol{\tau}_{\mathbf{T}} = [\tau_{\mathrm{T},1}, \cdots, \tau_{\mathrm{T},V}]^{\intercal}$ holds the training time $\tau_{\mathrm{T},v}$ of every vehicle v and $\circ$ is the Hadamard product. Then, the start time of the UL transmissions at beam $b \in \mathcal{B}_{\mathrm{TOT}}$ is given by

$$\tau_{\mathrm{L},b}[i] = \min_{1\le v\le V} \hat{\boldsymbol{\tau}}_{\mathbf{L},b}[i]. \tag{6.16}$$

Selected vehicles can be co-scheduled on the same beam if they can jointly finish their UL transmissions within a time interval which varies per beam and depends on the training time of each vehicle assigned to the given beam. Hence, the condition for co-scheduling is

$$\mathbb{1}_{1\times V}\cdot(\mathbf{T_{UL}}[i]\circ\mathbf{A}[i])\preccurlyeq(\tau_{\text{APP,MAX}}-\tau_{\text{DL}})\,\mathbb{1}_{1\times B_{\text{TOT}}}-\boldsymbol{\tau}_{\mathbf{L}}{}^{\mathsf{T}}(\mathbf{A}[i]),\tag{6.17}$$

where $\mathbf{T_{UL}}[i]=[\boldsymbol{\tau}_{\mathbf{UL},1}[i],\cdots,\boldsymbol{\tau}_{\mathbf{UL},B_{\text{TOT}}}[i]]$ is the $V\times B_{\text{TOT}}$ upload time duration matrix, where $\boldsymbol{\tau}_{\mathbf{UL},b}[i]$ is a $V\times 1$ vector with the upload time duration $\tau_{\text{UL},vb}$ of each vehicle v on beam b, $^{\mathsf{T}}$ denotes the transpose operator, $\preccurlyeq$ denotes the element-wise inequality and $\boldsymbol{\tau}_{\mathbf{L}}(\mathbf{A}[i])$ denotes the dependence on $\mathbf{A}[i]$. Figures 6.2(a) and 6.2(b) show two examples of assigning two vehicles to one beam. In both examples, the start time of the UL transmissions is equal to $\tau_{\text{L},b}=\min\{\tau_{\text{T},1},\tau_{\text{T},2}\}=\tau_{\text{T},1}$ and thus the transmissions from both vehicles should jointly be performed in a time interval of duration $\tau_{\text{APP,MAX}}-\tau_{\text{DL}}-\tau_{\text{T},1}$. Figure 6.2(a) illustrates that the two vehicles cannot be co-scheduled on the same beam because $\tau_{\text{UL},1}+\tau_{\text{UL},2}>\tau_{\text{APP,MAX}}-\tau_{\text{DL}}-\tau_{\text{T},1}$ and thus the constraint in (6.17) is violated, whereas Figure 6.2(b) shows that the two vehicles can be co-scheduled because their joint UL transmission time is short enough to fulfill the constraint in (6.17).

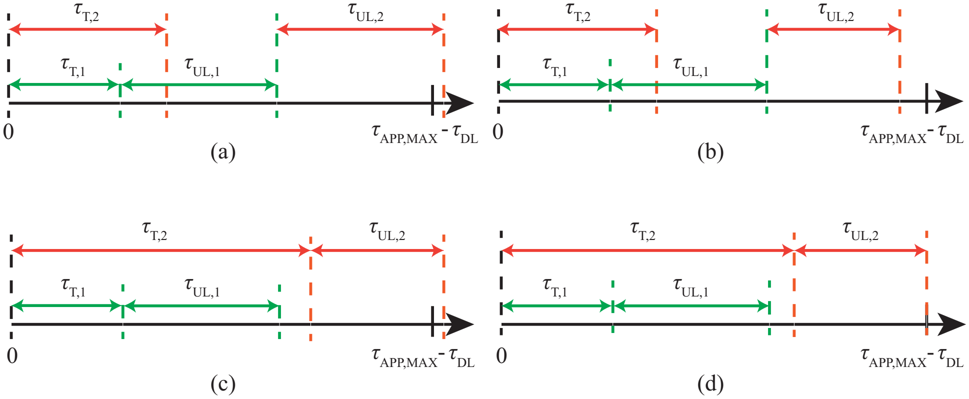


Figure 6.2: Examples of assigning two vehicles to one beam, where each example considers different combinations of local training latency and uplink transmission duration.

The fulfilment of (6.17) alone does not guarantee that vehicles, assigned to a given beam, can train and transmit within the time interval $\tau_{\text{T+UL}}=\tau_{\text{APP,MAX}}-\tau_{\text{DL}}$. The beam may be idle i.e., not actively receiving UL data, during the UL transmission interval, that also needs to be taken into account. Consider again the example of assigning two vehicles to one beam and that $\tau_{\text{L},b}=\min\{\tau_{\text{T},1},\tau_{\text{T},2}\}=\tau_{\text{T},1}$, as also shown in Figures 6.2(c) and 6.2(d). Therefore, the UL transmissions should be completed within a time interval of duration $\tau_{\text{APP,MAX}} - \tau_{\text{DL}} - \tau_{\text{T},1}$. Even though (6.17) is fulfilled, the example in Figure 6.2(c) illustrates that the two vehicles should not be co-scheduled because there is an unaccounted time interval, between the time that vehicle 1 finished its UL transmission and the time when vehicle 2 finished its local training, when the beam is idle. Figure 6.2(d) illustrates an example where both vehicles can be co-scheduled, regardless of the beam idle time. The key difference between the examples in Figures 6.2(c) and 6.2(d) is that in Figure 6.2(c) the time needed by vehicle 2 to train and upload its data $\tau_{\text{T},2} + \tau_{\text{UL},2}$ exceeds the scheduling interval $\tau_{\text{T+UL}} = \tau_{\text{APP,MAX}} - \tau_{\text{DL}}$. Therefore, an additional constraint is needed to ensure that selected vehicles can train and upload their model within the scheduling interval $\tau_{\text{T+UL}} = \tau_{\text{APP,MAX}} - \tau_{\text{DL}}$, which is given by

$$\boldsymbol{\tau}_{\mathbf{T}} + (\mathbf{T}_{\mathbf{UL}}[i] \circ \mathbf{A}[i]) \cdot \mathbb{1}_{B_{\text{TOT}} \times 1} \preccurlyeq (\tau_{\text{APP,MAX}} - \tau_{\text{DL}}) \mathbb{1}_{B_{\text{TOT}} \times 1}. \tag{6.18}$$

6.3.3. Problem Formulation

For a given communication round i, we formulate the following joint vehicle selection and resource allocation optimisation problem to maximise the total vehicle importance:

$$\max_{\mathbf{s}[i], \mathbf{A}[i]} \quad \text{Tr}(\mathbf{Q}[i] \cdot \mathbf{A}^{\mathsf{T}}[i]) \tag{6.19a}$$

$$\text{subject to} \quad \mathbf{A}[i] \cdot \mathbb{1}_{B_{\text{TOT}} \times 1} = \mathbf{s}[i], \tag{6.19b}$$

$$\mathbb{1}_{1 \times V} \cdot (\mathbf{T}_{\mathbf{UL}}[i] \circ \mathbf{A}[i]) \preccurlyeq (\tau_{\text{APP,MAX}} - \tau_{\text{DL}})\, \mathbb{1}_{1 \times B_{\text{TOT}}} - \boldsymbol{\tau}_{\mathbf{L}}{}^{\mathsf{T}}(\mathbf{A}[i]), \tag{6.19c}$$

$$\boldsymbol{\tau}_{\mathbf{T}} + (\mathbf{T}_{\mathbf{UL}}[i] \circ \mathbf{A}[i]) \cdot \mathbb{1}_{B \times 1} \preccurlyeq (\tau_{\text{APP,MAX}} - \tau_{\text{DL}}) \mathbb{1}_{B_{\text{TOT}} \times 1}, \tag{6.19d}$$

$$\mathbf{s}[i] \in \{0,1\}^{V \times 1}, \tag{6.19e}$$

$$\mathbf{A}[i] \in \{0,1\}^{V \times B_{\text{TOT}}}, \tag{6.19f}$$

where Tr(·) denotes the trace of a matrix i.e. the sum of the elements in the diagonal. The binary optimization variable $\mathbf{s}[i]$ indicates whether a vehicle is selected and the binary optimization matrix $\mathbf{A}[i]$ indicates the beam on which the selected vehicles are assigned to. Constraint (6.19b) indicates that the vehicles participating in the learning process at communication round i should be associated with exactly one beam in one cell. Constraint (6.19c) shows that vehicles can be co-scheduled on the same beam only if they can jointly finish their UL transmissions within the related time interval. Finally, constraint (6.19d) indicates that all selected vehicles should be able to train and transmit their local models within the scheduling interval $\tau_{\text{T+UL}} = \tau_{\text{APP,MAX}} - \tau_{\text{DL}}$.

It is worth mentioning that the optimisation problem in (6.19) resembles the agent selection framework in (5.9) in Chapter 5. Specifically, constraints (6.19c) and (6.19d) are a generalisation of constraints (5.9b) and (5.9c). This shows the usefulness of the agent selection framework in Chapter 5 and that it can be easily extended to the joint vehicle selection and resource allocation problem.

In the optimisation problem in (6.19), the selection of vehicles $\mathbf{s}[i]$ is defined based on the beam associations $\mathbf{A}[i]$ using (6.19b). Therefore, we can reduce the optimisation parameters by combining (6.19b) and (6.19e), which leads to the following optimisation problem

$$\begin{aligned}
\max_{\mathbf{A}[i]} \quad & \mathrm{Tr}(\mathbf{Q}[i] \cdot \mathbf{A}^{\mathsf{T}}[i]) && \text{(6.20a)}\\
\text{subject to} \quad & \mathbf{A}[i] \cdot \mathbb{1}_{B_{\mathrm{TOT}} \times 1} \preccurlyeq \mathbb{1}_{V \times 1}, && \text{(6.20b)}\\
& \mathbb{1}_{1 \times V} \cdot (\mathbf{T_{UL}}[i] \circ \mathbf{A}[i]) \preccurlyeq (\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}})\, \mathbb{1}_{1 \times B_{\mathrm{TOT}}} - \boldsymbol{\tau_{\mathrm{L}}}^{\mathsf{T}}(\mathbf{A}[i]), && \text{(6.20c)}\\
& \boldsymbol{\tau_{\mathrm{T}}} + (\mathbf{T_{UL}}[i] \circ \mathbf{A}[i]) \cdot \mathbb{1}_{B_{\mathrm{TOT}} \times 1} \preccurlyeq (\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}) \mathbb{1}_{V \times 1}, && \text{(6.20d)}\\
& \mathbf{A}[i] \in \{0,1\}^{V \times B_{\mathrm{TOT}}}. && \text{(6.20e)}
\end{aligned}$$

The optimisation problem (6.20) is non-linear, because the vector $\boldsymbol{\tau_{\mathrm{L}}}^{\mathsf{T}}(\mathbf{A}[i])$ requires the evaluation of a $\min(\cdot)$ function, as shown in (6.16). To that end, we define a relaxed version of the optimisation problem (6.20) by setting the same training time τ_{T} for all vehicles, which eliminates the need of the $\min(\cdot)$ function. Such a relaxation is achieved by considering that all vehicles have the same data set size K_v and the same processing capabilities. Additionally, the constraint (6.20d) is not needed in the relaxed problem because it is already satisfied with constraint (6.20c). Then, the relaxed optimisation problem where the vehicles have the same training times is given by

$$\begin{aligned}
\max_{\mathbf{A}[i]} \quad & \mathrm{Tr}(\mathbf{Q}[i] \cdot \mathbf{A}^{\mathsf{T}}[i]) && \text{(6.21a)}\\
\text{subject to} \quad & \mathbf{A}[i] \cdot \mathbb{1}_{B_{\mathrm{TOT}} \times 1} \preccurlyeq \mathbb{1}_{V \times 1}, && \text{(6.21b)}\\
& \mathbb{1}_{1 \times V} \cdot (\mathbf{T_{UL}}[i] \circ \mathbf{A}[i]) \preccurlyeq (\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}} - \tau_{L})\, \mathbb{1}_{1 \times B_{\mathrm{TOT}}}, && \text{(6.21c)}\\
& \mathbf{A}[i] \in \{0,1\}^{V \times B_{\mathrm{TOT}}}. && \text{(6.21d)}
\end{aligned}$$

For problems with a small number of variables and constrains, the optimal solution of the relaxed optimisation problem in (6.21) can be given with integer linear programming. In this chapter, we invoke the COIN-OR branch and cut (CBC) solver [124] for the solution of the problem in (6.21). Furthermore, we design a heuristic algorithm, namely the VBI algorithm, to approximate the solution of problem (6.20) and consequently, the VBI algorithm can also approximate the solution of problem (6.21).

6.3.4. Vehicle-Beam-Iterative (VBI) Algorithm

The VBI algorithm indicates which vehicles are selected for training at each communication round and assigns the selected vehicles to an appropriate BS beam. Hence, the

VBI algorithm forms vehicle-beam pairs with the aim to maximise the total vehicle importance. First, the VBI algorithm finds the best serving beam for each vehicle. Then, based on the results, the vehicle, which is best served on each beam is assigned to that beam. Therefore, at most one vehicle is assigned to each beam at a given iteration and the optimization matrix **A**, holding the vehicle-beam pairs, is updated. The process repeats for multiple iterations until the matrix **A** is fully defined. For the calculations, the VBI algorithm takes as input the training times τ_{T} of the vehicles, the upload times τ_{UL} of the vehicles per beam, the importance $\mathbf{Q}$ of vehicles per beam and the time period $\tau_{\mathrm{T+UL}} = \tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}$. The outputs of the VBI algorithm are the total vehicle importance Q_{TOT} and the matrix **A**. The VBI algorithm is described in Algorithm 1 and explained in detail as follows.

First, in lines 1-4, the initialisation steps are performed. In line 1, the vehicle-beam matrix **A** is initialised by setting $A_{vb} = 0$ to all vehicle-beam pairs that do not satisfy constraint (6.20d), i.e., $\tau_{\mathrm{T},v} + \tau_{\mathrm{UL},vb} \leq \tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}$. In line 2, a vector $\tau_{\mathbf{B}}$, holding the upload latency budget for every beam is defined and it is initialised with the time period $\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}$. In line 3, the vector $\tau_{\mathbf{L}}$ holding the training time per beam, i.e. the start time of the UL transmissions, is initialised to zero. Finally, in line 4, the total vehicle importance Q_{TOT} is initialised to zero.

After the initialisation step, from line 5 onwards, the algorithm repeats continuously until the matrix **A** is fully defined. As previously mentioned, at most one vehicle is assigned to each beam per iteration and hence at most V iterations are performed. Each iteration consists of the following five steps:

- **STEP 1 (lines 6-7)**: For each vehicle $v \in \mathcal{V}$ that has not already been selected for training, the beam b^* that maximises the importance q_{vb} of the vehicle is found, assuming that $A_{vb} \neq 0$.

- **STEP 2 (lines 8-10)**: From line 8 onwards, the algorithm iterates over all beams to define per beam $b \in \mathcal{B}_{\mathrm{TOT}}$ whether or not a vehicle will be assigned to it and which vehicle that will be. Depending on whether or not a vehicle is assigned to the beam, different steps are followed later on. Therefore, in line 9 a decision variable is initialised to False. Then, in line 10, based on the derived potential vehicle-beam pairs from step 1 (line7), the vehicle v^* that has the highest importance q_{vb} on each beam b is selected.

- **STEP 3 (lines 11-16)**: In this step a decision is taken on whether or not the selected vehicle v^* can be scheduled on beam b. In line 11, it is checked whether or not vehicle v^* is the first vehicle to be scheduled on beam b. If it is the first one, line 12 sets the training time $\tau_{\mathrm{L},b}$ at beam b equal to the training time τ_{T,v^*} of vehicle v^*

Algorithm 1 The Vehicle-Beam-Iterative (VBI) Algorithm

```
Input: Training time $\tau_{\mathbf{T}}$ of vehicles, upload time $\mathbf{T_{UL}}$ of vehicles per beam, importance
    $\mathbf{Q}$ of vehicles per beam and time period $\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}$
Output: Vehicle selection and beam allocation $\mathbf{A}$ and total vehicle importance $Q_{\mathrm{TOT}}$
 1: Set $A_{vb} = 0$ if $\tau_{\mathrm{T},v} + \tau_{\mathrm{UL},vb} > \tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}$ for each vehicle $v \in \mathcal{V}$ and beam $b \in \mathcal{B}_{\mathrm{TOT}}$
 2: Set $\tau_{\mathrm{B},b} = \tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}$ for each beam $b \in \mathcal{B}_{\mathrm{TOT}}$
 3: Set $\tau_{\mathrm{L},b} = 0$ for each beam $b \in \mathcal{B}_{\mathrm{TOT}}$
 4: Set $Q_{\mathrm{TOT}} = 0$
 5: while matrix A not fully defined do
 6:     for every vehicle $v \in \mathcal{V}$ not yet selected do
 7:         Find $b^* = \mathrm{argmax}_{b \in \mathcal{B}_{\mathrm{TOT}}}(q_{vb})$, given that $A_{vb} \neq 0$
 8:     for every beam $b \in \mathcal{B}_{\mathrm{TOT}}$ do
 9:         Set scheduled = False
10:         Find $v^* = \mathrm{argmax}_{v \in \mathcal{V}_b}(q_{vb})$, where $\mathcal{V}_b$ holds the vehicles selecting beam $b$ as
            their $b^*$
11:         if $v^*$ is the first vehicle scheduled on beam $b$, i.e. $\tau_{\mathrm{L},b} = 0$ then
12:             Set training time $\tau_{\mathrm{L},b} = \tau_{\mathrm{T},v^*}$
13:             Set scheduled = True
14:         else if $\tau_{\mathrm{UL},v^*b} \leq \tau_{\mathrm{B},b} - \min(\tau_{\mathrm{L},b}, \tau_{\mathrm{T},v^*})$ then
15:             Set training time $\tau_{\mathrm{L},b} = \min(\tau_{\mathrm{L},b}, \tau_{\mathrm{T},v^*})$
16:             Set scheduled = True
17:         if scheduled = True then
18:             Set $A_{v^*b} = 1$
19:             Set $A_{v^*\hat{b}} = 0$ for all other beams $\hat{b} \in \mathcal{B}_{\mathrm{TOT}} \setminus b$
20:             Update $\tau_{\mathrm{B},b} -= \tau_{\mathrm{UL},v^*b}$
21:             Update total importance $Q_{\mathrm{TOT}} += q_{v^*b}$
22:     for every vehicle $v \in \mathcal{V}$ not yet selected do
23:         for every beam $b \in \mathcal{B}_{\mathrm{TOT}}$ do
24:             if $\tau_{\mathrm{UL},vb} > \tau_{\mathrm{B},b} - \tau_{\mathrm{L},b}$ then
25:                 $A_{vb} = 0$
26: return A, $Q_{\mathrm{TOT}}$
```

and line 13 sets the decision variable to True. If vehicle v^* is not the first vehicle to be scheduled on beam b, line 14 evaluates according to constraint (6.20c) whether or not vehicle v^* can be co-scheduled with the other vehicle(s) already scheduled on beam b. If vehicle v^* can be co-scheduled, in line 15, the training time $\tau_{\mathrm{L},b}$ at beam b is set to the minimum time between the training time $\tau_{\mathrm{L},b}$ set in a previous iteration, when scheduling a different vehicle, and the training time τ_{T,v^*} of the newly scheduled vehicle v^*. Also, line 16 sets the decision variable to True.

- **STEP 4 (lines 17-21)**: If vehicle v^* is scheduled on beam b, i.e. the decision variable is True, line 18 sets the appropriate new entry in the vehicle-beam $\mathbf{A}$ matrix, i.e. $A_{v^*b} = 1$. Then, line 19 sets $A_{v^*\hat{b}} = 0$ for all other beams $\hat{b} \in \mathcal{B}_{\mathrm{TOT}} \setminus b$ because each vehicle can only be assigned to one beam. Next, line 20 updates the uploading latency budget $\tau_{\mathrm{B},b}$ accordingly and line 21 increases the total vehicle importance Q_{TOT} with the importance q_{v^*b} of the newly scheduled vehicle. In case that vehicle v^* is not scheduled, i.e. the decision variable is False, no action is taken and the vehicle can be re-considered for scheduling in the next iteration.

- **STEP 5 (lines 22-25)**: After iterating over all beams and before starting a new iteration as a result of line 5, an update step takes place. Specifically, lines 22-25 discard vehicle-beam pairs, i.e. $A_{vb} = 0$, that cannot fulfil constraint (6.20d) due to the newly scheduled vehicles of the given algorithm iteration.

Finally, once all iterations are completed, line 26 returns the vehicle-beam matrix $\mathbf{A}$ and the total vehicle importance Q_{TOT}.

The complexity of the algorithm can be split in two parts. The first part relates to the initialization steps, which have complexity $\mathcal{O}(VB_{\mathrm{TOT}})$ due to the calculations in line 1. The second part relates to STEPS 1-5, which also have complexity $\mathcal{O}(VB_{\mathrm{TOT}})$, for a single beam iteration. As previously mentioned, the algorithm performs at most V iterations and hence, the complexity is $\mathcal{O}(V^2B_{\mathrm{TOT}})$.

Because the VBI algorithm depends on the importance q_{vb}, the value of the tuning parameter ρ influences the behaviour of the algorithm. Specifically, when $\rho = 0$, the importance q_{vb} depends only on the resource consumption $C_{R,vb}$, which essentially depends on the bit rate R_{vb} and hence varies per beam. Therefore, to maximise the total vehicle importance, the algorithm selects the vehicles with the strongest wireless channels and assigns them to the beams that they experience the highest bit rate R_{vb}. This way, the VBI algorithm maximises the number of selected vehicles and it is expected that it provides a close to optimal solution.

On the other hand, when the tuning parameter $\rho = 1$, the importance q_{vb} depends only on the training loss $F(\mathbf{W}_{G,v})$, which is independent of the beam. Therefore, when $\rho = 1$, the VBI algorithm gives priority to the vehicles with high training loss $F(\mathbf{W}_{G,v})$ and they may be assigned on a sub-optimal beam, in terms of the resource consumption $C_{R,vb}$. However, the algorithm does take the latency constraints into account and hence, the shorter the latency budget $\tau_{\mathrm{APP,MAX}}$ is, the more likely is for a vehicle to be assigned on the beam it experiences the lowest resource consumption $C_{R,vb}$ (and highest bit rate

R_{vb}). If a vehicle is assigned to a sub-optimal beam, in terms of the resource consumption $C_{R,vb}$, more resources will be consumed, which in return will limit the total number of selected vehicles.

6.4. Scenario Configuration

This section presents the considered scenarios to evaluate the performance of the VBI algorithm. For the evaluations, we consider the object classification task on the ETSD, using the same CNN architecture and configuration as in Section 5.5.1. Next, the baseline algorithms, which will be compared against the VBI algorithm are presented. Finally, the wireless scenario is presented.

6.4.1. Learning Scenarios

In the analysis we consider scenarios with $V = 50$ vehicles and both IID and non-IID data. Moreover, we perform evaluations for the relaxed problem in (6.21), where all vehicles have the same training data set size and the original problem in (6.20), where vehicles have different training data set sizes. Hence, we consider in total four learning scenarios. The configuration of each learning scenario is summarised in Table 6.1 and explained in more detail as follows:

- **Scenario 1: Same data set size with IID data**: All vehicles have the same number of training samples $K_v = 150$, which are evenly distributed over the ten classes.
- **Scenario 2: Same data set size with non-IID data**: All vehicles have the same number of training samples $K_v = 150$, which are unevenly split over two classes such that on average all classes are equally represented in the training data set $\mathcal{K}$.
- **Scenario 3: Different data set sizes with IID data**: The vehicles have a different number of training samples K_v, which are unevenly split over the ten classes. The number of samples per class per vehicle is drawn from a Poisson distribution with a rate of 15.
- **Scenario 4: Different data set sizes with non-IID data**: The vehicles have different number of training samples K_v, which are unevenly split over two classes. The number of samples for each of the two classes per vehicle is drawn from a Poisson distribution with a rate of 75.

Additionally, in all four scenarios, for the calculation of the loss $F(\mathbf{W}_{G,v}[i])$ of vehicle v at communication round i, the categorical cross-entropy loss function is applied on the testing data set $\mathcal{K}_{T,v}$, which is unique for every vehicle and three times smaller than the training data set $\mathcal{K}_v$. Moreover, the split of the testing data set among the vehicles

Table 6.1: Configuration of the four learning scenarios.

Parameter	Same data set sizes		Different data set sizes	
	IID	non-IID	IID	non-IID
Number of classes per vehicle	10	2	10	2
Training samples K_v per vehicle	150	150	150 (on average)	150 (on average)
Training samples per class per vehicle	15	75 (on average)	15 (on average)	75 (on average)
Testing samples $K_{T,v}$ per vehicle	50	50	50 (on average)	50 (on average)
Testing samples per class at FL server	100	100	100	100

and the classes is similar to the split of the testing data set. Finally, in all scenarios, the accuracy of the global model is measured at the FL server based on its specific testing data set, which consists of 100 samples per class.

For the training, in all four scenarios, the vehicles invoke the SGD optimiser with learning rate $\eta = 0.05$, batch size $s_B = 64$ and with each vehicle performing $n_{\mathrm{LE}} = 2$ local epochs. The number of FLOPs required from the vehicles to train the CNN for a batch size $s_B = 64$ is measured by the Keras library, in Python, which is $n_{\mathrm{FLOP},G} = 6.55$ GFLOPs. Regarding the hardware of the vehicles, in all four learning scenarios we consider the processing capabilities $g_v = 64$ GFLOPs per second. Therefore, the training time $\tau_{\mathrm{T,v}}$ of vehicle v, as given by (5.7) depends on the number of training K_v samples at vehicle v, which depends on the learning scenario.

6.4.2. BASELINE ALGORITHMS

To evaluate the performance of the VBI algorithm we consider two baseline algorithms, namely the `max-loss-rate` and the `random-rate` algorithms, which are based on the `max-loss` and `random` policies, respectively, in Chapter 5, in regards to vehicle selection. The indication of rate in the name of the algorithms relates to the beam assignment. Specifically, the `max-loss-rate` algorithm aims to maximise the sum of the losses over all selected agents based on a rate-based beam assignment. First, it sorts the vehicles in descending order based on their loss $F(\mathbf{W}_{G,v})$ and selects as many vehicles as possible until the constraints in (6.20) are violated. Each selected vehicle v is assigned to the beam b^* that it experiences the lowest resource consumption C_{R,vb^*}. If a vehicle

v cannot be assigned to its best beam b^*, the vehicle v is not selected for training. The `random-rate` algorithm is implemented similarly with the only difference that it iterates over the vehicle list in a random order.

6.4.3. Wireless Scenario

For the wireless communication scenario, we consider an urban macro environment at $f_C = 3.5$ GHz and a bandwidth of $B = 50$ MHz [30, 125]. For the wireless propagation we assume a path loss exponent $\gamma = 3.7$ and shadowing with $\sigma_{\text{SF}} = 8$ dB, which are typical values for outdoor dense urban environments [40]. The considered area is covered by $M = 7$ three-sectorised BSs that are placed on a hexagonal grid with inter-site distance of $500m$ at a height of $25m$ [30]. Each sector is equipped with a 4×4 UPRA that is capable of simultaneously forming $B_M = 12$ beams.

In urban macro deployments a high number of vehicles is expected, which drive around an urban grid consisting of three $433m$ x $250m$ blocks and thus an area of size $433m$ x $750m$ [30]. Each street around the block has a total of 4 lanes and there are 2 lanes per driving direction. The lane width is $3.5m$ [30]. Moreover, the vehicles are driving with a speed of 60 km per hour and their antennas are placed at a height of $1.6m$ [125]. At the intersections the vehicles have a probability of 0.5 to keep driving straight ahead, 0.25 to go left and 0.25 to go right [125]. Finally, each vehicle is equipped with a 2×2 UPRA which can steer the beam to the direction of the beam formed at the respective sector.

Based on the considered UPRA model, as defined in Section 6.2.2, an analysis is carried out in Appendices A.1.1 and A.1.2 to define the beam directions of the GoB. The derived beam directions are the same at all sectors and they are the following: $-45°, -15°, 15°$ and $45°$ in the azimuth plane and $17°, 47°$ and $77°$ in the elevation plane. Based on the derived beam directions, a coverage analysis is performed in Appendix A.2.1 and it revealed that all roads are covered, i.e. $RSRP > -120$ dBm, by the beams pointing at the cell edge. Therefore, for our evaluation we only consider the four cell edge beams. The rest of the UPRA parameters for both the BSs and the vehicles, are calculated in Appendices A.1.2 and A.1.4 and they are summarised in Table 6.2.

Previously it was assumed that during a given communication round, the vehicles stay connected to one and the same beam. Therefore, in Appendix A.1.3 we approximate the time that a vehicle stays connected to a single beam. From the analysis, it is estimated that vehicles stay connected to a cell edge beam for up to $10.4 - 17.0$ seconds. This time interval serves as an upper bound to the latency budget $\tau_{\text{APP,MAX}}$ to ensure that the assumption that vehicles stay connected to one and the same beam is not violated.

In Appendix A.2.2 we calculate the bit rate at the cell edge at 105 Mbps, which can also serve as the broadcast bit rate. Considering that the FL model size is $Z \approx 13.4$ MB, the broadcast time duration is $\tau_{\text{DL}} \approx 1.02$ seconds. Finally, we set the latency budget

Table 6.2: Parameters of the UPRAs at the BSs and the vehicles.

Parameter	Value	Parameter	Value
$P_{\mathrm{M,MAX}}$	49 dBm	$P_{\mathrm{V,MAX}}$	23 dBm
NF_{M}	5 dB	NF_{V}	9 dB
$G_{\mathrm{M,MAX}}$	12 dBi	$G_{\mathrm{V,MAX}}$	6 dBi
$\Delta\varphi_M$	25°		
SLL_M	19.1		

$\tau_{\mathrm{APP,MAX}} = 2.5$ seconds, which allows to schedule at least one vehicle per beam.

6.5. Evaluation

This section presents the evaluation of the VBI algorithm. First, in Section 6.5.1 the relative performance of the VBI algorithm to the optimal solution of the problem in (6.21) is studied. Then, we evaluate the VBI algorithm against the baseline algorithms presented in Section 6.4.2, in terms of the accuracy of the global model. For this evaluation, we consider the four learning scenarios as presented in Section 6.4.1. Specifically, Sections 6.5.2 and 6.5.3 show the results for the scenarios where vehicles have the same and different data set sizes, respectively. Finally, in Section 6.5.4, we compare the four learning scenarios in terms of how long it takes to reach a certain accuracy level. We present the results of the four learning scenarios as an average of 15 independent simulations and the source code generating all data is available in [114].

6.5.1. VBI Algorithm Relative Performance

We evaluate the VBI algorithm in regards to the problem in (6.21), i.e. when vehicles have the same training data set sizes. For the evaluation, we compare the performance of the VBI algorithm, in terms of the total vehicle importance Q_{TOT}, to the one of the optimal solution, which is given by the CBC solver. For the comparison, three different values of the tuning parameter ρ are considered: the two extreme cases of $\rho = 0$ and $\rho = 1$ and the case of $\rho = 0.8$, which leads to a similar value range for the local loss $F(\mathbf{W}_{G,v})$ and the resource consumption $C_{R,vb}$. Moreover, the number of vehicles V in the network is also varied. Finally, the comparison is performed after communication round 0 for the scenario with non-IID data and the obtained results are averaged over 1000 independent simulations.

Figure 6.3 shows that for the extreme case of $\rho = 0$, the VBI algorithm provides a close to optimal solution, regardless of the number of vehicles in the network, as it was expected from the qualitative analysis in Section 6.3.4. That is because, under configura-

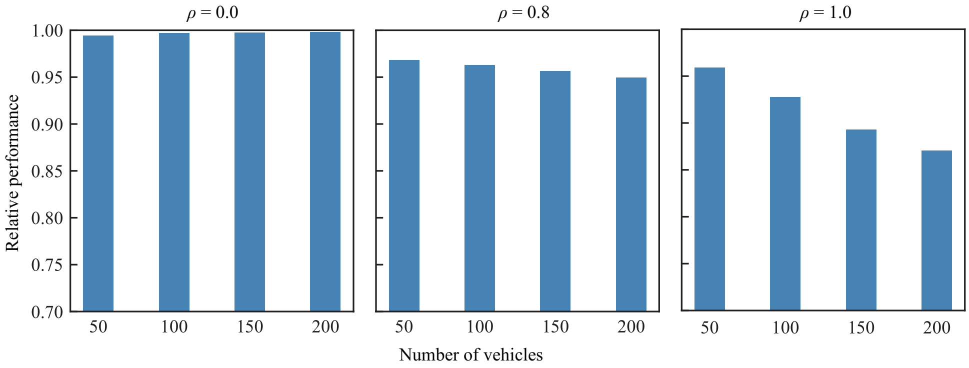


Figure 6.3: Relative performance of the VBI algorithm to the optimal solution, for the problem where vehicles have the same training data set sizes.

tion $\rho = 0$, the VBI algorithm selects the vehicles with the strongest wireless channels and assigns them to the beam they experience the lowest resource consumption $C_{R,vb}$. Consequently, the VBI algorithm maximises the number of scheduled vehicles, which leads to a close to optimal total vehicle importance Q_{TOT}. For the remainder of the evaluation section, we refer to the VBI algorithm with $\rho = 0$ as `VBI-rate`.

At the other extreme case of $\rho = 1$, the VBI algorithm prioritises the selection of vehicles with high loss $F(\mathbf{W}_{G,v})$, which might be assigned to a sub-optimal beam, as it was explained in Section 6.3.4. Figure 6.3 shows that the relative performance of the VBI algorithm when $\rho = 1$ is lower than when $\rho = 0$, which is a result of the sub-optimal beam assignment. Specifically, the sub-optimal beam assignment leads to a higher resource consumption, which in return limits the total number of vehicles that can be scheduled and consequently the total vehicle importance Q_{TOT}. Figure 6.3 also shows that the performance further decreases with the number of vehicles in the network, because there is a higher chance that a vehicle will be assigned to a sub-optimal beam. For the remainder of the evaluation section, we refer to the VBI algorithm with $\rho = 1$ as `VBI-loss`.

When $\rho \in (0,1)$, and specifically $\rho = 0.8$ in this comparison, Figure 6.3 shows that the VBI algorithm performs worse than when $\rho = 0$ but better than when $\rho = 1$. This is because the tuning parameter ρ configures the vehicle importance q_{vb} to take almost equally into account the resource consumption $C_{R,vb}$ and the loss $F(\mathbf{W}_{G,v})$. Therefore, during beam assignment there is some distinction among the beams to define which is the best serving beam in terms of the resource consumption $C_{R,vb}$ but this distinction is not as prominent as with $\rho = 0$. Hence, the larger the ρ is, the less distinction there is among the beams, which consequently leads to a sub-optimal beam assignment. For the remainder of the evaluation section, we refer to the VBI algorithm with $\rho = 0.8$ as `VBI-0.8`.

Based on the above observations, it is concluded that the VBI algorithm provides a closer to optimal solution when $\rho \to 0$, because it can then leverage the distinction of beams in terms of the resource consumption $C_{R,vb}$. However, the relative performance of the VBI algorithm is not directly related to the accuracy of the global model. Hence, in the following sections, we study the performance of the VBI algorithm in terms of the accuracy of the global model.

6.5.2. SAME DATA SET SIZES

We evaluate the performance of the VBI algorithm in terms of the accuracy of the global model for the problem where the vehicles have the same training data set size, as defined in (6.21). To that end, we compare the `VBI-rate`, `VBI-loss`, `max-loss-rate` and `random-rate` algorithms for the scenarios with IID and non-IID data. The `VBI-0.8` algorithm is only considered in a sensitivity analysis for the scenario with non-IID data.

IID DATA

Figure 6.4 shows the accuracy over time and illustrates that all four algorithms have a similar performance. The similar performance of the `VBI-loss` and `max-loss-rate` algorithms is expected because both algorithms select the vehicles with the highest loss $F(\mathbf{W}_{G,v})$. Additionally, both algorithms select approximately the same number of vehicles per communication round. This implies that the two algorithms are almost identical and that the `VBI-loss` algorithm mostly assigns to the selected vehicles, the beam that leads to the lowest resource consumption $C_{R,vb}$. This resource efficient beam assignment is a result of the short latency budget $\tau_{\text{APP,MAX}}$, which enforces that only vehicles with very good wireless channels can participate in the learning process, as previously explained. Because of the vehicles' mobility, the channel quality of the vehicles varies over time and hence the channel quality limitation due to the latency budget $\tau_{\text{APP,MAX}}$ applies to a different set of vehicles per communication round. Therefore, all four algorithms perform resource efficient beam assignment and all vehicles have fair chances, over time, of getting selected.

Moreover, Figure 6.4 shows that even though the `VBI-rate` and `random-max` algorithms do not take the loss $F(\mathbf{W}_{G,v})$ into account, they perform similarly to the loss-aware `VBI-loss` and `max-loss-rate` algorithms. This is because all vehicles have samples from all classes and hence the choice vehicles is not crucial for the learning. This result is in line with Result 5.3 in Chapter 5. Therefore, the main result is the following:

Result 6.1 *When vehicles have the same data set size and IID data are considered, the choice of vehicles is not crucial, considering that resource efficient beam assignment is performed.*

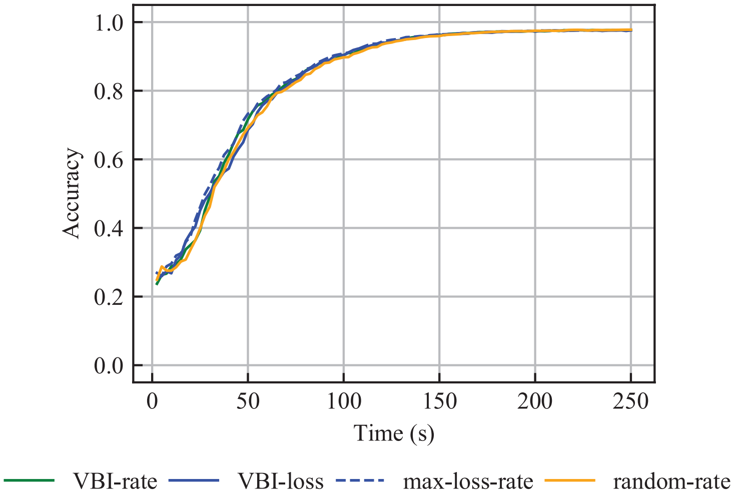


Figure 6.4: Accuracy over time for the relaxed problem with IID data, averaged over 15 independent simulations.

Non-IID Data

When non-IID data are considered, the vehicles have samples from only two classes and therefore, the selection of vehicles in a given communication round is more crucial than in scenarios with IID data, as also concluded in Result 5.2. Figure 6.5 illustrates that with non-IID data, the loss-aware `VBI-loss` and `max-loss-rate` algorithms outperform the loss-unaware `VBI-rate` and `random-rate` algorithms. That is because the former algorithms take both learning and channel aspects into account. The learning aspect ensures that the vehicles with samples that contribute more to the learning process are more often selected than other vehicles whereas the channel aspect ensures resource efficient beam assignment. Recall that the `VBI-loss` algorithm implicitly takes the channel quality into account via the latency budget $\tau_{\text{APP,MAX}}$. Additionally, Figure 6.5 shows that the `VBI-loss` and `max-loss-rate` algorithms behave almost identically, for the same reason as explained for the scenario with IID data. Therefore, it is again concluded that all four algorithms perform resource efficient beam assignment.

Moreover, Figure 6.5 illustrates that even though the four considered algorithms behave differently, they eventually all converge to the same accuracy level. Specifically, the 96% accuracy level is reached within 250s. The convergence to the same accuracy level is a result of the algorithms selecting many vehicles per communication round and hence eventually training enough on the most appropriate samples. Moreover, recall from Chapter 5 (Table 5.1) that in a time interval of 300s, no configuration with non-IID data lead to a higher than 90% accuracy level. The use of MU-MIMO plays a key role in

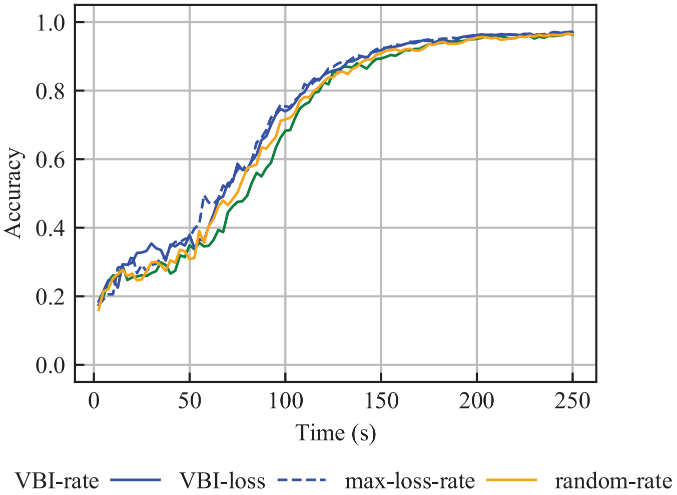


Figure 6.5: Accuracy over time for the relaxed problem with non-IID data, averaged over 15 independent simulations.

achieving a higher accuracy because it enhances the quality of the wireless channels and allows the selection of more vehicles per communication round. Therefore, MU-MIMO enables training on a bigger sample set per communication round and it also allows vehicles on the cell edge to also potentially participate in the learning process. These benefits of MU-MIMO eventually improve the convergence time of the global model. Therefore, the key takeaway results are:

Result 6.2 *When vehicles have the same data set sizes and non-IID data are considered, loss-aware algorithms provide higher accuracy during the initial learning phase, assuming that resource efficient beam assignment is performed.*

Result 6.3 *MU-MIMO capable BSs improve the attainable accuracy level within a given time interval. This is a result of enhancing the quality of the wireless channels and selecting many vehicles per communication round.*

To further study the performance of the VBI algorithm, we compare the `VBI-0.8` algorithm to the two extreme configurations of the VBI algorithm, i.e. `VBI-rate` and `VBI-loss`. Figure 6.6 shows that the accuracy with the `VBI-0.8` algorithm is bounded by the accuracy of the `VBI-rate` and `VBI-loss` algorithms. This result verifies that the parameter ρ can successfully tune the relative importance of the loss $F(\mathbf{W}_{G,v})$ and the resource consumption $C_{R,vb}$ in the vehicle importance q_{vb} calculation. This tuning of the vehicle importance q_{vb} is then reflected in terms of accuracy. Therefore, for the remainder of the evaluation section, we only show the performance of the `VBI-rate` and

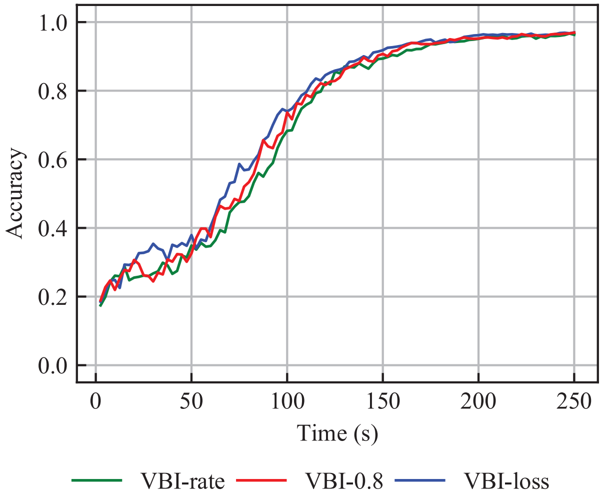


Figure 6.6: Accuracy over time for the relaxed problem with non-IID data, when considering different values of ρ for the VBI algorithm. The accuracy is averaged over 15 independent simulations.

`VBI-loss` algorithms.

6.5.3. Different Data Set Sizes

We now evaluate the performance of the VBI algorithm in terms of the accuracy of the global FL model for the problem where vehicles have different training data set sizes, and thus different training times $\tau_{\mathrm{T},v}$, as described in problem (6.20). Even though the VBI algorithm does not explicitly select vehicles based on their training time $\tau_{\mathrm{T},v}$, vehicles that have shorter training times $\tau_{\mathrm{T},v}$, have a higher chance of getting selected for training. This behaviour of the algorithm is due to the constraint that the selected vehicles need to train and upload their local model within the given latency budget $\tau_{\mathrm{APP,MAX}}$. Therefore, vehicles with a short training time $\tau_{\mathrm{T},v}$ can be selected even if their channel quality is not very good. On the other hand, vehicles with a high training time $\tau_{\mathrm{T},v}$ can only be selected when they have a very good channel quality.

IID data

Figure 6.7 shows the accuracy over time for the four considered algorithms and illustrates that the `VBI-rate`, `VBI-loss` and `max-loss-rate` algorithms perform similarly, while the `random-rate` algorithm underperforms. The performance of the `VBI-loss` and `max-loss-rate` algorithms is similar due to the applied short latency budget $\tau_{\mathrm{APP,MAX}}$, as previously explained in Section 6.5.2. Moreover, the design of the two loss-based algorithms and the `VBI-rate` algorithm allows the algorithms to more often select vehicles with a high training time $\tau_{\mathrm{T},v}$ compared to the `random-rate` algorithm. Thus, the

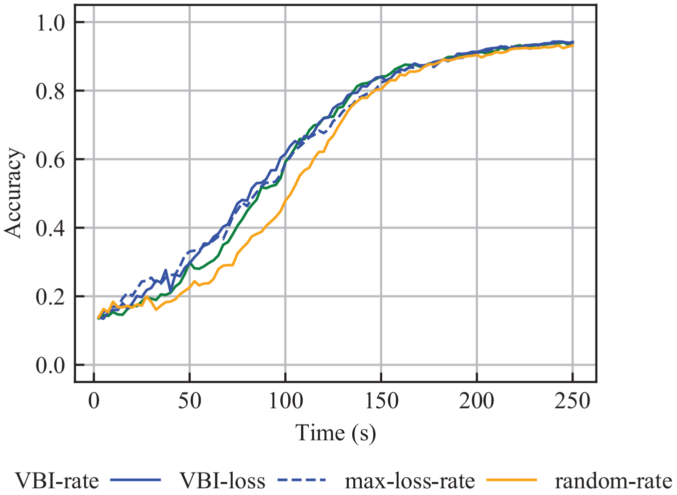


Figure 6.7: Accuracy over time for the original problem with IID data, averaged over 15 independent simulations.

`random-rate` algorithm, trains more often on a specific set of samples and hence has a slower convergence than the other three algorithms.

Specifically, the `VBI-rate` algorithm gives priority to the vehicles with good channel quality. Therefore, when vehicles with high training times $\tau_{\mathrm{T},v}$ experience good channels, they are likely to be selected. Additionally, the loss-based algorithms give priority to the vehicle with a high loss $F(\mathbf{W}_{G,v})$. Consequently, if vehicles have a high loss $F(\mathbf{W}_{G,v})$, they will be selected for training, once their channel quality allows for it. Figure 6.7 shows that the three algorithms perform similarly. It can then be concluded that the choice of vehicles in scenarios with IID data is not crucial, as long as all vehicles contribute to the learning process, which is a conclusion in line with Result 6.1.

Moreover, vehicles with short training times $\tau_{\mathrm{T},v}$, have few training samples K_v. Thus, their contribution to the global model is not significant. Considering that vehicles with short training times $\tau_{\mathrm{T},v}$ are often selected, the global model does not change significantly per communication round. Therefore, the convergence time is longer compared to the scenario where all vehicles have the same training times $\tau_{\mathrm{T},v}$. The main message in this configuration is:

Result 6.4 *When vehicles have different data set sizes and IID data are considered, it is important to ensure that all vehicles can participate in the learning process.*

Non-IID Data

Figure 6.8 shows the accuracy over time for the scenario with non-IID data and it illustrates that all four algorithms perform similarly. This result shows that the considered algorithms are insensitive to this specific scenario. Previously, in Section 6.5.2 it was concluded that with non-IID data, the loss-based `VBI-loss` and `max-loss-rate` algorithms provide performance gains (Result 6.2). However, when vehicles have different data set sizes, the loss-based algorithms do not provide any performance gains. The fact that some vehicles have small testing data set size $K_{T,v}$, implies that those vehicles calculate their loss $F(\mathbf{W}_{G,v})$ inaccurately. Hence, the loss-based algorithms cannot distinguish which vehicles are important to the learning process. This results to selecting vehicles more evenly, which is beneficial only in the scenario with IID data. We can therefore conclude that the loss $F(\mathbf{W}_{G,v})$ is not a good metric to indicate the importance of a vehicle in the learning process when some vehicles have small testing data set sizes $K_{T,v}$.

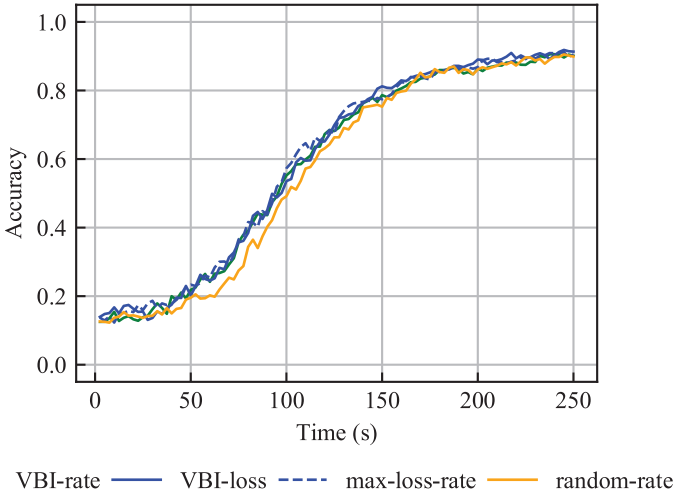


Figure 6.8: Accuracy over time for the original problem with non-IID data, averaged over 15 independent simulations.

To mitigate this problem, we performed simulations while considering both the loss $F(\mathbf{W}_{G,v})$ and the testing data set sizes $K_{T,v}$ in the learning importance q_{vb}. However, the obtained results showed no accuracy improvements. When we consider the testing data set sizes $K_{T,v}$ as a metric for selecting vehicles, we implicitly prioritize the selection of vehicles with long training times $\tau_{\mathrm{T},v}$. Therefore, the uploading time is limited and consequently fewer vehicles are scheduled compared to the scenario where the testing data set size $K_{T,v}$ is not incorporated in the learning importance q_{vb}. This reduction on the number of scheduled vehicles hinders the learning process and does not allow for faster learning. Thus, it is highlighted that further research is necessary to identify a good metric describing the importance of a vehicle in the learning process. The key take away message is:

Result 6.5 *When vehicles have different data set sizes and non-IID data are considered, the loss $F(\mathbf{W}_{G,v})$ cannot provide a clear distinction of which vehicles are important to the learning process.*

6.5.4. Comparison of Learning Scenarios

Some applications require to train the global model until a specific accuracy target is met. Therefore, we compare the four algorithms, in terms of how much time is needed to reach the 85% and 90% accuracy levels. To average out simulation noise, we consider that the accuracy level is reached if the average accuracy is above the accuracy target for 30 seconds. Table 6.3 shows the time in seconds to reach each accuracy level, where a hyphen indicates that the accuracy level could not be reached within the simulated 250 seconds.

Table 6.3: Time, in seconds, needed to reach the 85% and 90% accuracy levels for every algorithm in each learning scenario.

Algorithm	Same data set sizes and IID data		Same data set sizes and non-IID data		Different data set sizes and IID data		Different data set sizes and non-IID data	
	85%	90%	85%	90%	85%	90%	85%	90%
`VBI-rate`	95	115	145	170	170	208	198	-
`VBI-loss`	95	112	137	157	172	206	190	243
`max-loss-rate`	92	110	137	155	175	208	193	248
`random-rate`	97	117	145	162	180	212	195	-

Table 6.3 shows that all algorithms reach the accuracy levels approximately at the same time. For the scenario addressing the problem with different data set sizes and non-IID data, all four algorithms reach the 90% accuracy target after about 250 seconds. The hyphen for the `VBI-rate` and `random-rate` algorithms implies that they have not reached yet the target but it can be seen from Figure 6.8 that all four algorithms have approximately the same accuracy after 250 seconds. The reason why all algorithms behave similarly after the initial learning phase, regardless of the learning scenario, is attributed to MU-MIMO, which improves the quality of the wireless channels and allows to select many vehicles per communication round, as also highlighted in Result 6.3.

Moreover, Table 6.3 shows that it takes longer to reach the accuracy targets when vehicles have different data set sizes compared to when they have the same data set size. As mentioned in Section 6.5.3, in scenarios with different data set sizes, the vehicles with short training times are more often selected for training, which then requires more communication rounds to reach a certain accuracy target. Finally, when comparing the results in Table 6.3 to the results in Table 5.2, we conclude that MU-MIMO significantly increases the accuracy level within a given time interval, as also highlighted in Result 6.3.

6.6. Concluding Remarks

This chapter addressed the joint vehicle selection and resource allocation problem for FL, considering MU-MIMO capable BSs in vehicular networks. Specifically, we described the related optimisation problem in two scenarios; when vehicles have the same and different data set sizes. We approximated the solution of the optimisation problems with the proposed VBI algorithm, which we then evaluated in different learning scenarios. The results showed that MU-MIMO capable BSs increase the convergence time of the global model. Moreover, it was shown that the global model accuracy target is achieved faster in scenarios where vehicles have the same data set sizes than in scenarios where vehicles have different data set sizes. Finally, it was concluded that the loss-aware algorithms provide accuracy gains in non-IID scenarios, only when vehicles have the same data set sizes.

III

Deployment of an Drone Swarm

7

Random Geometric Graphs for Drone Networks

In this chapter, we address the connectivity of a swarm of drones. First, we model the swarm as a RGG with a distance-based connection function. Then we derive the link density in D dimensions, and for the 2D and 3D spaces we show that the link density is accurately approximated by the Fréchet distribution, for any rectangular space. We derive expressions, in terms of the link density, for the minimum number of nodes needed in the 2D and 3D spaces to ensure network connectivity. These results provide first-order estimates for the deployment of a swarm of drones to provide coverage in a disaster or crowded area.

This chapter is based on a published paper [126].

7.1. Introduction

Random graphs are created from a set $\mathcal{N}$ of N nodes, placed in a space $V \in \mathbb{R}^D$, where each pair of nodes is connected by a link with probability p, independently of the existence of any other link [127]. If the node i at position r_i and the node j at position r_j are connected with probability $p_{ij} = f(|r_i - r_j|)$, where $f(r)$ is a real function of the distance r, then we talk about a *random geometric graph* (RGG). If $f(r) = 1_{r<r_0}$, where 1_x is the indicator function[1], then all nodes at distance smaller than r_0 are connected almost surely [128, 129]. Moreover, the position r_i of each node i itself can be either deterministic or stochastic. In the latter case, the link existence is doubly stochastic and depends both on the distance function $f(r)$ and on the random placement of nodes described by a probability distribution $\Pr[r_1 \leq x_1, \cdots, r_N \leq x_N]$.

There is extensive work in literature on the properties of RGGs and their applications. RGGs can model transportation networks such as wireless [130, 131] and airline [132] networks as well as infrastructural networks like power grids [133]. Also, RGGs can be applied in analysing the structure of large data sets [134] and in modelling ad hoc networks, which are decentralised networks that do not rely on a fixed infrastructure. Applications of ad hoc networks include vehicular, disaster relief, sensor and flying swarm robotics networks [135, 136].

7

In this chapter, we focus on the *link density* and the *connectivity* of 2D and 3D RGGs, with an application to wireless networks. We define the link density as the ratio of the expected number of links over the maximum possible number of links in an undirected graph and the connectivity as the probability that a path exists between any pair of nodes in the graph. Bettstetter [131] studies the number of nodes needed to provide connectivity in a 2D RGGs and Dall and Christensen [137] provide the critical connectivity threshold in D dimensions. Van Mieghem [138] presents the exact solution for the link density and the average number of paths between any two nodes, when the graph is randomly generated in a square. Erba et al. [134] compare the average number of subgraphs in highly dimensional RGGs characterised by indicator-based and exponential distance functions. Moreover, multiple approximations to the nodal degree in bounded spaces are performed [139–141], however they are related to the 2D space and to indicator-based distance functions.

Focusing on connectivity in wireless communications, Hekmat and Van Mieghem [142] derive the giant component size for 2D RGGs with a log-normal distance function and show that it is a good measure for connectivity. Ng et al. [143] provide upper and lower bounds for the critical density of 2D and 3D RGGs with a log-normal distance function and under the unit disc model. By distributing the nodes inside or on the surface

[1] $1_x = 1$ if condition x is true, else $1_x = 0$

of a sphere, Khalid and Durrani [144] provide exact expressions for the mean node degree and the node isolation probability. They leave as an open problem the derivation of these expressions when the nodes are distributed in a cube. Finally, Dettmann and Georgiou [145] derive the full connection probability in 2D and 3D convex domains for various distance functions.

The main contributions of this chapter are the following:

- We derive an exact expression for the link density for an RGG in a D-dimensional prism and any distance function $f(r)$ allowing its graph properties to be elegantly and accurately deduced from an Erdős-Rényi random graph $G_p(N)$, whose theory is well developed [127].

For 2D and 3D RGGs modelling wireless networks that are characterised by a simple distance-based path loss model and Rayleigh fading:

- We derive an approximation of the link density in a hypercube that illustrates the importance of the nodes placed in its corners.
- We analytically demonstrate how the link density depends on the path loss exponent and on the prism size and shape. We further show that the link density in the 3D space is smaller than or equal to that in the 2D space.
- We show that the complementary distribution function of the Fréchet distribution accurately approximates the link density for any path loss exponent, prism size and prism shape.
- We deduce a general closed-form expression in terms of the link density to approximate the minimum density of nodes to ensure a connected network.

The remainder of this chapter is organised as follows. Section 7.2 describes the network model. The link density in the D-dimensional space is derived in Section 7.3, which also presents an approximation of the link density in a hypercube. The impact of the wireless environment and the hyperprism's shape and volume on the link density is assessed in Section 7.4. Additionally, Section 7.4 illustrates the high accuracy of the link density approximation with the Fréchet distribution as well as a brief motivation on the approximation accuracy. In Section 7.5, we derive the minimum number of nodes needed for connectivity based on the link density. Finally, Section 7.6 concludes the chapter with a summary and the future work.

7.2. Network Model

A graph $G(N, L)$ consists of a set $\mathcal{N}$ of N nodes and a set $\mathcal{L}$ of L links. We assume that nodes are placed uniformly at random inside a hyperprism in D dimensions, with one vertex at the origin and with length Z_d in the d-th orthogonal direction of the coordinate axes. The distance function $f(r)$ provides the connection probability between two nodes placed at $r_i = (r_{i_1}, \cdots, r_{i_D})$ and $r_j = (r_{j_1}, \cdots, r_{j_D})$, where $r = |r_i - r_j|$ denotes their mutual distance. Figure 7.1 draws an example of the considered graph with $f(r) = e^{-0.07r^2}$ in the 3D space.

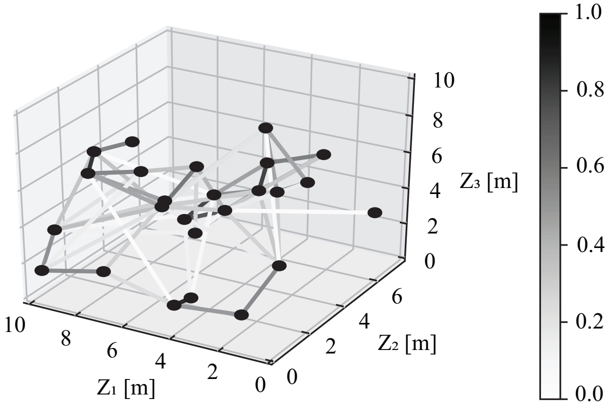


Figure 7.1: Example realisation of a RGG with $f(r) = e^{-0.07r^2}$ and $N = 25$ in a 3D rectangular prism, where the link colour refers to the connection probability between two nodes. For visualisation purposes, only links with $f(r) > 10^{-3}$ are shown.

In wireless networks, the distance function $f(r)$ is influenced by the wireless channel between the nodes. The RGG assumption on independent link existence for distinct node pairs, is approximate for wireless networks, because wireless transmissions interfere with each other, thus creating dependency between the node pairs. Wireless networks that operate on a dedicated frequency band or in isolation from other wireless networks and ensure orthogonal transmissions in e.g. frequency or time, can be exactly modelled by RGGs.

The impact of the wireless channel is reflected by the received signal power $P_{\text{RX},ij}$, which depends on the distance r between the transmitter i at location r_i and the receiver j at location r_j. Assuming that the *path loss* is approximately characterised by a power law and that G_i and G_j denote the antenna gains at the transmitter and receiver, respectively, the received signal power $P_{\text{RX},ij}$ is [40]

$$P_{\text{RX},ij}(r) = P_{\text{TX},ij} G_i G_j K \left(\frac{r_c}{r}\right)^{\gamma} \tag{7.1}$$

with

$$K = \left(\frac{\lambda}{4\pi r_c}\right)^2 < 1,$$

where $P_{\mathrm{TX},ij}$ is the transmit power for communication from i to j in Watts, λ is the wavelength in meters, r_c is the reference distance for the antenna far field in meters, γ is the path loss exponent and $r > r_c$ is given in meters. The reference distance $r_c \in [1, 100]$ depends on the propagation environment and on the antenna characteristics [40] while it further holds that $r_c \gg \lambda$, which implies $K < 1$.

Ignoring interference and thus assuming independent links, the SNR Γ_{ij} that the receiver j experiences from the transmitter i is given by

$$\Gamma_{ij}(r) = \frac{P_{\mathrm{RX},ij}(r)}{P_{\mathrm{NOISE}}} = \frac{P_{\mathrm{TX},ij} G_i G_j K}{P_{\mathrm{NOISE}}} \left(\frac{r_c}{r}\right)^{\gamma}, \tag{7.2}$$

where P_{NOISE} is the thermal noise power. Assuming a fixed transmission power $P_{\mathrm{TX},ij} = P_{\mathrm{TX},ji} = P_{\mathrm{TX}}$ at every node, two nodes i and j are connected if and only if the $\Gamma_{ij} = \Gamma_{ji}$ is greater than the SNR threshold Γ_{MIN}. Therefore,

$$f(r) = \Pr[\Gamma_{ij}(r) > \Gamma_{\mathrm{MIN}}],$$

where $\Pr[\cdot]$ indicates the probability. Using (7.2), this can be rewritten as

$$f(r) = \Pr\left[r < r_0\right],$$

where

$$r_0 = r_c \left(\frac{P_{\mathrm{TX}} G_i G_j K}{P_{\mathrm{NOISE}}\, \Gamma_{\mathrm{MIN}}}\right)^{1/\gamma}, \tag{7.3}$$

and thus $r_0 \geq r_c$, denotes the *maximum allowed distance* between node i and node j such that they are connected. Since the received power $P_{\mathrm{RX},ij}$ reduces with distance r, node i has a spherical coverage area and thus any node j located within a sphere with radius r_0 is connected to the node i. Hence, the distance function can be written as a step function $f(r) = 1_{r<r_0}$.

When considering *multipath fading*, the received signal power $P_{\mathrm{RX},ij}$ varies randomly due to signal reflections and it is given at time t and frequency ν by

$$P_{\mathrm{RX},ij}(r, t, \nu) = P_{\mathrm{TX}} G_i G_j K \left(\frac{r_c}{r}\right)^{\gamma} ||H_{ij}(t, \nu)||^2, \tag{7.4}$$

where $H_{ij}(t, \nu)$ is the channel response to multipath fading on the channel between transmitter i and receiver j. With (7.4), the distance function $f(r) = \Pr[\Gamma_{ij}(r, t, \nu) > \Gamma_{\mathrm{MIN}}]$ can now be written as

$$f(r) = \Pr\left[||H_{ij}(t, \nu)||^2 > \frac{\beta}{K}\left(\frac{r}{r_c}\right)^{\gamma}\right], \tag{7.5}$$

where β is the *minimum required channel gain* given by

$$\beta = \frac{P_{\mathrm{NOISE}}\, \Gamma_{\mathrm{MIN}}}{P_{\mathrm{TX}} G_i G_j}. \tag{7.6}$$

Because generally $P_{\text{NOISE}} \ll P_{\text{TX}}$, we have $\beta < 1$. For a 'typical' drone-to-drone application, it is estimated that $\beta < 10^{-10} = -100$ dB.

Assuming that $H_{ij}(t,\nu)$ is Rayleigh distributed [134], then $||H_{ij}(t,\nu)||^2$ is exponentially distributed with a mean of 1 and (7.5) becomes

$$f(r) = e^{-\frac{\beta}{K}\left(\frac{r}{r_C}\right)^{\gamma}}. \tag{7.7}$$

The distance function in (7.7) is commonly considered in the literature with $r_c = 1$ m for wireless networks [145], while it also appears in studies on the properties of data sets and of machine learning algorithms [134] in highly dimensional RGGs. It is convenient to rewrite (7.7) with respect to the maximum allowed distance r_0, as derived for the case without multipath fading, using (7.3) and (7.6):

$$f(r) = e^{-\left(\frac{r}{r_0}\right)^{\gamma}} \tag{7.8}$$

for which $\lim_{\gamma\to\infty} f(r) = 1_{r<r_0}$ and $f(r_0) = \frac{1}{e} \approx 0.3678$.

Figure 7.2 illustrates the distance function $f(r)$ for different values of γ in terms of the normalised distance, defined as the ratio of the distance r over the maximum allowed distance r_0. A wide value range of γ is considered to understand the behaviour of the distance function $f(r)$ for all real, positive numbers. Figure 7.2 exemplifies that for a given γ, the distance function $f(r)$ reduces with distance r. Furthermore, the distance function $f(r)$ increases in γ for $r < r_0$, but decreases in γ for $r > r_0$ since with a higher γ, the received signal power $P_{\text{RX},ij}$ attenuates more quickly over distance r.

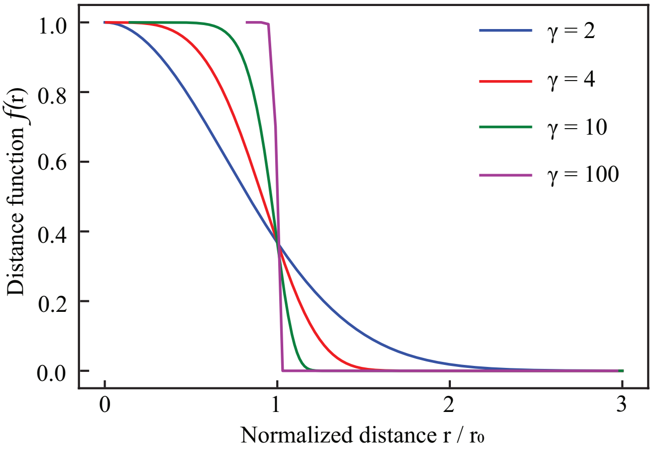


Figure 7.2: Distance function $f(r)$ with respect to the normalised distance $\frac{r}{r_0}$ when $\beta = -127\ dB$, $\lambda = 0.08\ m$ and $r_c = 1\ m$.

7.3. LINK DENSITY IN D DIMENSIONS

We define the link density $p = \frac{E[L]}{L_{\text{MAX}}}$ as the ratio of the expected number $E[L]$ of links over the maximum possible number $L_{\text{MAX}} = \frac{N(N-1)}{2}$ of links in an undirected graph. In this section, we derive the link density of a RGG in a rectangular hyperprism in D dimensions and provide an approximation to the link density in a hypercube.

7.3.1. LINK DENSITY ANALYSIS

The number of links $L[\{\mathcal{R}\}]$ in an RGG, where $\{\mathcal{R}\} = \{r_1, r_2, \cdots r_N\}$ is the set with the nodal positions, in space $V \in \mathbb{R}^D$ is given by [138]

$$L[\{\mathcal{R}\}] = \sum_{i=1}^{N} \sum_{j=i+1}^{N} f\left(|r_i - r_j|\right),$$

where $f(\cdot)$ is the function generating an RGG with N nodes. The expected number of links $E[L]$ is given by

$$E[L] = \int_V \Pr[\{\mathcal{R}\}]\, L[\{\mathcal{R}\}]\, d[\{\mathcal{R}\}],$$

where $\Pr[\{\mathcal{R}\}] = g_{\{r_1,\cdots,r_N\}}(x_1,\cdots,x_N)$ is the probability density function (PDF) of the position of the set of nodes, given by

$$\Pr[\{\mathcal{R}\}] = \frac{d\Pr[r_1 \le x_1, \cdots, r_N \le x_N]}{dx_1 \cdots dx_N}$$

resulting in

$$E[L] = \int_V dr_1 \cdots dr_N\, g_{\{r_1,\cdots,r_N\}}(r_1,\cdots,r_N) \sum_{i=1}^{N} \sum_{j=i+1}^{N} f\left(|r_i - r_j|\right). \tag{7.9}$$

We can proceed further, if we assume independence in nodal positions, i.e. $\Pr[r_1 \le x_1, \cdots, r_N \le x_N] = \prod_{n=1}^{N} \Pr[r_n \le x_n]$, with corresponding PDF

$$g_{\{r_1,\cdots,r_N\}}(x_1,\cdots,x_N) = \prod_{n=1}^{N} g_{r_n}(x_n).$$

Then, the expected number of links in (7.9) reduces to

$$\begin{aligned} E[L] &= \int_V \prod_{n=1}^{N} dr_n g_{r_n}(r_n) \sum_{i=1}^{N} \sum_{j=i+1}^{N} f\left(|r_i - r_j|\right) \\ &= \sum_{i=1}^{N} \sum_{j=i+1}^{N} \int_V dr_i \int_V dr_j g_{r_i}(r_i) g_{r_j}(r_j)\, f\left(|r_i - r_j|\right). \end{aligned}$$

Assuming identical distributions $\Pr[r_n \le x] = \Pr[r \le x]$ and a same PDF $g_{r_n}(x) = g_r(x)$ for any node $n \in \mathcal{N}$, the corresponding link density p is

$$p = \frac{E[L]}{L_{\text{MAX}}} = \int_V dq \int_V ds\, g_r(q) g_r(s)\, f\left(|q - s|\right). \tag{7.10}$$

7

When the nodes are placed uniformly at random inside a D-dimensional rectangular hyperprism with edge lengths $Z_1, Z_2, \cdots, Z_D$ and volume $v = \prod_{d=1}^{D} Z_d$, so that $g_r(x) = \frac{1}{v}$, the integral in (7.10) can be analytically evaluated. Numerical evaluation of the more general expression given in (7.10) is rather straightforward, given that the IID nodal location density $g_r(x)$ is known. Choosing the uniform density $g_r(x) = \frac{1}{v}$ and a square of size Z in 2D, the link density p for an arbitrary distance function $f(r)$ is derived in [138]. Appendix B.1 generalises the link density p to D dimensions in a rectangular hyperprism. Because in Cartesian coordinates the distance between nodes r_i and r_j is given by $|r_i - r_j|^2 = \sum_{d=1}^{D}(r_{i_d} - r_{j_d})^2$, we denote $f(|r_i - r_j|) = h(|r_i - r_j|^2)$ to simplify the notation and (7.10) becomes

$$p = 2^D \int_0^{Z_1} du_1 \cdots \int_0^{Z_D} du_D \prod_{d=1}^{D} \frac{(Z_d - u_d)}{Z_d^2} h\left(\sum_{d=1}^{D} u_d^2\right), \tag{7.11}$$

where u_d is the location variable in dimension d. The analytical derivation of (7.11) with $D = 2$ for general distance function $f(r) = h(r^2)$ is presented in Appendix B.1. Analytical derivation of (7.11) in higher dimensions $D > 2$ is cumbersome.

Instead of integrating over the positions u_d in D-dimensions, we can integrate over the distance between two nodes and (7.11) is rewritten as the expectation of the distance function $f(R)$

7

$$p = E[f(R)] = \int_0^{r_{\text{MAX}}} f(r) g_R(r) dr \tag{7.12}$$

where the random variable $R \in [0, r_{\text{MAX}}]$ of the distance has PDF $g_R(r)$. Even though (7.11) and (7.12) are the same when nodes are independently placed at random inside a hyperprism, (7.12) is implicit and assumes the knowledge of the PDF $g_R(r)$.

7.3.2. Link Density Approximation in a Hypercube

A number of papers, e.g. [140, 141, 145], study the boundary effects of the considered space on the nodal degree and connectivity. In this work, the boundary effects are captured in the derivation of the link density p in (7.11). We approximate here the link density p in (7.11) for the case of a hypercube with $Z_1 = \cdots = Z_D = Z$ to study the effects of the nodes located at the corners of the hyperprism. Assuming that one vertex of the hypercube is at the origin, we consider a part of a hypersphere of radius Z and centre at the origin, which is entirely enclosed by the hypercube. Thus, in the 2D space, the link density $p_{\text{square-2D}}$ in a square is approximated by the link density $p_{\text{circle/4}}$ in a quarter of a circle while in 3D, the link density $p_{\text{cube-3D}}$ in a cube is approximated by the link density $p_{\text{sphere/8}}$ in an octant of a sphere. In Appendix B.2.1 and B.2.2 we derive the link densities

$p_{\text{circle}/4}$ and $p_{\text{sphere}/8}$, respectively, leading to

$$p_{\text{square-2D}} = \int_0^1 h(Zx)\left(2\pi x - 8x^2 + 2x^3\right) dx + p_{\text{error-2D}} \tag{7.13}$$

$$p_{\text{cube-3D}} = \int_0^1 h(Zx)\left(4\pi x^2 - 6\pi x^3 + 8x^4 - x^5\right) dx + p_{\text{error-3D}} \tag{7.14}$$

where $p_{\text{error-2D}}$ and $p_{\text{error-3D}}$ denote the errors introduced by the approximations in the 2D and 3D spaces, respectively.

Appendix B.2.3 solves $p_{\text{circle}/4}$ and $p_{\text{sphere}/8}$ for any value of γ of the distance function (7.7). Comparing the link densities $p_{\text{circle}/4}$ and $p_{\text{sphere}/8}$ with the exact link densities $p_{\text{square-2D}}$ and $p_{\text{cube-3D}}$ derived from (7.11), the errors $p_{\text{error-2D}}$ and $p_{\text{error-3D}}$ are determined and shown in Figure 7.3.

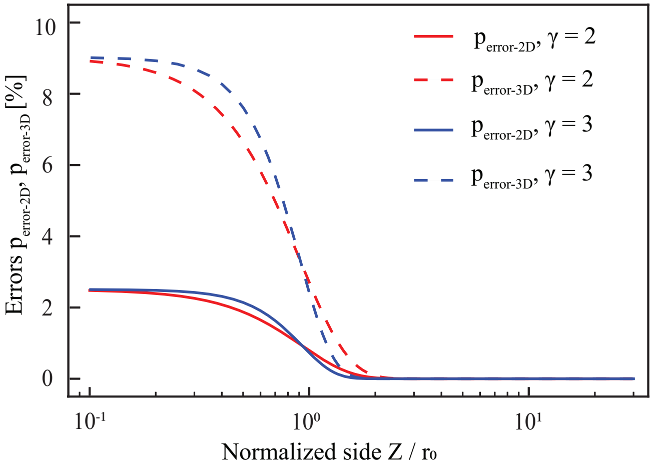


Figure 7.3: Error from approximate solution of link density in a square and a cube for $r_c = 1$ m, $K = 4.65 \cdot 10^{-5}$ and $\beta = -103.3$ dB.

Figure 7.3 shows that the approximation of the link density $p_{\text{square-2D}}$ in the 2D space, as shown in (7.13), is more accurate than the link density $p_{\text{cube-3D}}$ in the 3D space, as shown in (7.14), regardless of γ. Indeed, the partial circle/sphere does not cover the whole area/volume of the square/cube and thus the nodes located in the distant corner are neglected. In general, the volume ratio of the inscribed hypersphere over the hypercube is equal to $v_D = \frac{\pi^{\frac{D}{2}}}{\Gamma(\frac{D}{2}+1)2^D}$, which is independent of the size Z and rapidly tends to zero with D. For example, $v_2 = \frac{\pi}{4} = 0.7854$, $v_3 = \frac{\pi}{6} = 0.5236$, $v_4 = \frac{\pi^2}{32} = 0.3084$, $v_5 = \frac{\pi^2}{60} = 0.1644$ and $v_6 = \frac{\pi^3}{384} = 0.0807$. In other words, the higher the number of dimensions D, the worse the approximation and the larger the ratio $1 - v_D$ of the neglected 'corner' volume. This explains why the error $p_{\text{error-3D}}$ is larger than the error $p_{\text{error-2D}}$.

Figure 7.3 also shows that even though the fraction of the uncovered area/volume is fixed for any Z, the approximations are accurate when $\frac{Z}{r_0}$ is greater than about 2 and 2.8, for $\gamma = 3$ and $\gamma = 2$, respectively or more specifically when the link densities $p_{\text{square-2D}}$ and $p_{\text{cube-3D}}$ are greater than about 0.8 and 0.5, respectively and regardless of γ. Apparently, the corner nodes that are neglected from the approximate link densities $p_{\text{circle}/4}$ and $p_{\text{sphere}/8}$, have negligible influence on the link densities $p_{\text{square-2D}}$ and $p_{\text{cube-3D}}$, respectively, for large Z. Typically, links involving a corner node are characterised by a large distance r and, consequently, a small connection probability that tends to zero for large R. Furthermore, beside the size Z of the hypercube, the accuracy also depends on γ. Specifically, the impact of the corner nodes on the link density is less prominent for large γ, because the distance function $f(r)$ decreases in γ (for $r > r_0$).

When the nodes are placed independently and uniformly at random in a square of size Z, the PDF of the distance between two nodes $g_R(r)$ in (7.12) is equal to [146]:

$$g_R(r) = \begin{cases} \dfrac{2\pi Z^2 r - 8Zr^2 + 2r^3}{Z^4}, & \text{for } 0 < r \le Z, \quad (7.15) \\[2ex] \dfrac{-2r^3 + 8Zr\sqrt{r^2 - Z^2} + 2Z^2 r\left(4\arcsin\left(\frac{Z}{r}\right) - 2 - \pi\right)}{Z^4}, & \text{for } Z < r \le \sqrt{2}Z. \quad (7.16) \end{cases}$$

After the transformation $r = Zx$, the $p_{\text{circle}/4}$ and $p_{\text{error-2D}}$ terms in (7.13) are again found, using (7.15) and (7.16) in (7.12), respectively. Therefore, the approximation (7.13) indeed neglects all links between nodes located at the corner of the square. Similar conclusions apply for any number of dimensions D.

7.4. Evaluation with Simulations

For the distance function $f(r)$ in (7.8), the link density p in (7.11) is simulated and the influence of the path loss exponent γ and of the geometry of the prism in 2D and 3D is studied. We also show that the link density p is *accurately approximated* by a Fréchet distribution. We denote the side ratio $\omega = \frac{Z_2}{Z_1}$ and the height ratio $\delta = \frac{Z_3}{Z_1}$ and assume that $Z_1 \ge Z_2$ and $Z_1 \ge Z_3$, implying that $0 < \omega \le 1$ and $0 < \delta \le 1$.

7.4.1. Impact of Environment

Figure 7.4 shows the link density p_{2D} and p_{3D} in the 2D and 3D spaces versus the normalised length $\frac{Z_1}{r_0}$ of side Z_1 w.r.t. the maximum allowed distance r_0, when varying γ. Figure 7.4 shows that the link density p increases with the loss exponent γ when $\frac{Z_1}{r_0}$ is less than a threshold, that depends on the dimension and prism's shape and size. For $Z_1 < r_0$, the majority of distances between two nodes obeys $r < r_0$ and thus the link density p behaves similarly to the distance function $f(r)$ for $r < r_0$ ($\frac{1}{e} \le f(r) \le 1$). When Z_1

is sufficiently larger than r_0, the distance between two nodes r can be much greater than r_0 and thus the distance function $f(r)$ can take any value between zero and one. This is also the reason why after a $\frac{Z_1}{r_0}$ threshold value, the link density p behaves the same for any γ.

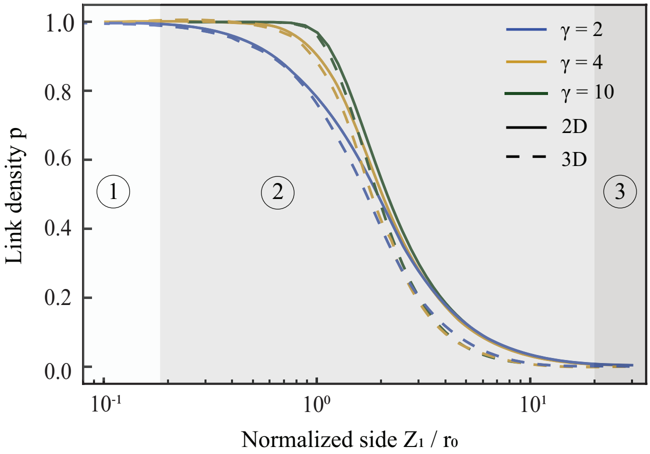


Figure 7.4: Impact of the path loss exponent γ on the link density p in the 2D and 3D spaces for $\omega = 0.75$, $\delta = 0.5$, $r_c = 1$ m, $K = 4.65 \cdot 10^{-5}$ and $\beta = -103.3$ dB.

When $Z_1 \sim r_0$, border effects play a role, as previously explained. Equation (7.3) demonstrates that r_0 decreases with γ and the border effects influence the link density when γ decreases, for a particular value of Z_1. However, when Z_1 is sufficiently larger than r_0, the borders have no impact on the link density. Moreover, $\lim_{\gamma\to\infty} f(r) = 1_{r<r_0}$ and for $\gamma \to \infty$ in (7.3), it holds that $r_0 = r_c$. Thus, the limit of $\gamma \to \infty$ in (7.11), results in $p = 0$, due to the restriction $r > r_c$ in wireless networks. Additionally, for $\frac{Z_1}{r_0} \to 0$ the link density $p \to 1$ because either the distance r between any two nodes is very small ($r \to 0$), as an effect of $Z_1 \to 0$, and thus $\lim_{r\to 0} f(r) = 1$, or because $\lim_{r_0\to\infty} f(r) = 1$, as a result of $r_0 \gg Z_1$.

The link density in Figure 7.4 versus $x = \frac{Z_1}{r_0}$ is fitted by

$$p(x) = 1 - e^{-\left(\frac{x-a}{b}\right)^{-c}}, \tag{7.17}$$

where $F_X(z) = \Pr[X \le z] = e^{-\left(\frac{z-a}{b}\right)^{-c}} 1_{z\ge a}$ is a scaled Fréchet distribution of r.v. $X \ge 0$ and the parameters $a \in (-\infty,\infty)$, $b \in (0,\infty)$ and $c \in (0,\infty)$ are the location of the minimum, scale and shape of the Fréchet distribution, respectively. The values of (a_{2D}, b_{2D}, c_{2D}) and (a_{3D}, b_{3D}, c_{3D}) fitting the link density p curves in the 2D and 3D spaces, respectively, as shown in Figure 7.4, are given in Table 7.1 along with their standard error. The root mean square error (RMSE) of each fit is less than 0.01 and emphasises the remarkably

Table 7.1: Fit values for (7.17) with $\omega = 0.75$, $\delta = 0.5$, $r_c = 1$ m, $K = 4.65 \cdot 10^{-5}$ and $\beta = -103.3$ dB.

γ	2	4	10
$a_{2\text{D}}$	-1.09 ± 0.02	-0.40 ± 0.02	0.00 ± 0.03
$b_{2\text{D}}$	2.54 ± 0.02	2.05 ± 0.02	1.79 ± 0.03
$c_{2\text{D}}$	2.28 ± 0.01	2.18 ± 0.02	1.99 ± 0.03
$a_{3\text{D}}$	-2.00 ± 0.02	-1.27 ± 0.03	-0.29 ± 0.01
$b_{3\text{D}}$	3.31 ± 0.03	2.80 ± 0.03	1.94 ± 0.01
$c_{3\text{D}}$	3.50 ± 0.02	3.62 ± 0.03	2.82 ± 0.01

high accuracy of the Fréchet approximation (7.17). The dependence of the fitting parameters a, b, c on γ, ω and δ , shown in Appendix B.3, highlights that the parameter $c \approx D$ approximately equals the dimensions D, when the border effects are minimal.

In summary, the Fréchet distribution in (7.17) very accurately approximates the link density with a distance function $f(r) = e^{-\left(\frac{r}{r_0}\right)^{\gamma}}$ for any physically interesting γ and all prism geometries. Therefore, this new insight motivates the use of (7.17), instead of (7.11), in applications.

7.4.2. Hard RGG in a Square and the Fréchet Distribution

To motivate the accurate fitting with the Fréchet distribution, we consider the special case of a hard RGG with $\lim_{\gamma\to\infty} f(r) = 1_{r<r_0}$. Then, the link density in (7.12) for a square of size Z is equal to $p_{\text{inf-2D}} = G_R(r) = \Pr[R \leq r_0]$. Setting $x = \frac{Z}{r_0}$, (7.12) leads to, for $0 < r_0 \leq Z$:

$$p_{\text{inf-2D}}(x) = \frac{x^{-4}}{2} - \frac{8x^{-3}}{3} + \pi x^{-2}, \tag{7.18}$$

for $Z < r_0 \leq \sqrt{2}Z$:

$$p_{\text{inf-2D}}(x) = -\frac{x^{-4}}{2} + \frac{8(x^{-2}-1)^{3/2}}{3} + 4\sqrt{x^{-2}-1} + 4x^{-2}\arcsin x - (2+\pi)x^{-2} + \frac{1}{3} \tag{7.19}$$

and for $r_0 > \sqrt{2}Z$, the link density $p_{\text{inf-2D}} = 1$.

The link density $p_{\text{inf-2D}}$ can be approximated by a Poisson point process (PPP), where N nodes are uniformly distributed in a circle with radius Z. The probability in a PPP to have an isolated node equals $p_{\text{iso}} = e^{-\rho\pi r_0^2}$, where $\rho = \frac{N}{\pi Z^2}$ is the node density. Hence, the probability to have a link is

$$\Pr[R \leq r_0] = 1 - e^{-\rho\pi r_0^2}.$$

Because $p_{\text{inf-2D}} = \Pr[R \leq r_0]$ we can write using $x = \frac{Z}{r_0}$:

$$p_{\text{inf-2D-PPP}}(x) = 1 - e^{-\left(\frac{x}{b}\right)^{-2}} \tag{7.20}$$

where $b = \sqrt{N}$. Thus, $p_{\text{inf-2D-PPP}}(x)$ satisfies (7.17) with parameters $(0, \sqrt{N}, 2)$ of the Fréchet distribution.

Figure 7.5 shows the link density for a square and for $\gamma \to \infty$ as derived *(i)* via simulations from (7.11), *(ii)* by fitting (7.11) with the Fréchet distribution in (7.20) and *(iii)* by (7.18) and (7.19). Figure 7.5 illustrates that the Fréchet distribution approximates the link density in (7.11) *remarkably well.* Specifically, the fit of the Fréchet distribution with parameters $(0, 1.65, 2)$ yields an RMSE of 0.005 and the difference on a plot is hardly visible. The Fréchet distribution is only slightly inaccurate when $r_0 \sim Z$, which is due to border effects that are not captured in (7.20). Additionally, Figure 7.5 shows that the link density in (7.18) and (7.19) are the exact solutions of (7.11). The simplicity of the Fréchet distribution compared to the complexity of the exact link density (7.11) is remarkable and motivates its use in applications.

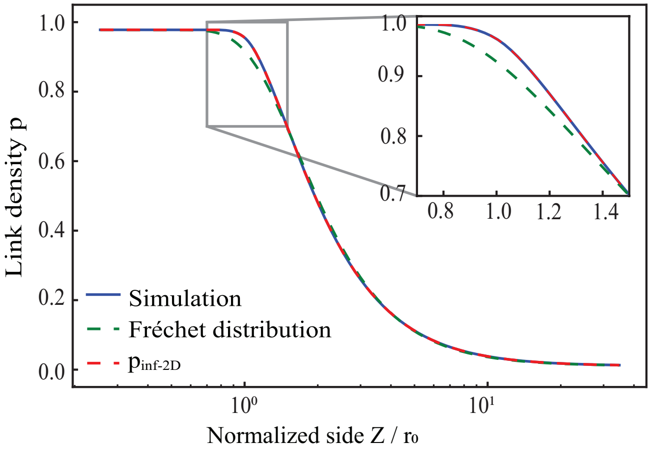


Figure 7.5: Link density comparison between the simulation of (7.11) (blue), the fitting of a Fréchet (green) and the polynomial solution from (7.18) and (7.19) for a square of size Z and $\gamma \to \infty$.

7.4.3. Difference in Dimensions

Figure 7.4 shows that the link density $p_{3D} \le p_{2D}$, regardless of the value of γ. We identify three regions for $\frac{Z_1}{r_0}$, indicated by the encircled numbers in Figure 7.4:

1. $p_{2D} = p_{3D} = 1$: the distance between any two nodes is short enough to provide a link.

2. p_{2D} upper bounds p_{3D}: the 3D distance between any two nodes is always larger than or equal to its projection in the 2D space.

3. $p_{2D} \to 0$ and $p_{3D} \to 0$: the distance between any two nodes is too large to provide a link.

Figure 7.6 shows the link density difference $p_{2D} - p_{3D}$ of the Fréchet approximation (7.17). The difference $p_{2D} - p_{3D}$ behaves similarly for any γ and it is maximised at around $\frac{Z_1}{r_0} = 2.75$, which is dependent on the geometry given by ω and δ.

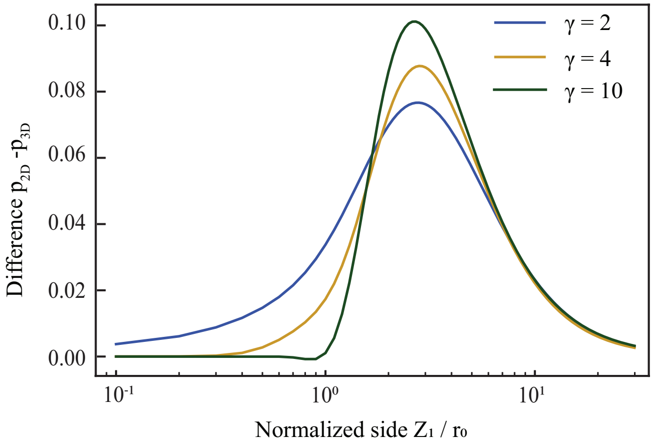


Figure 7.6: Difference $p_{2D} - p_{3D}$ of link density between the 2D and 3D spaces, for different values of γ when $\omega = 0.75$, $\delta = 0.5$, $r_c = 1$ m, $K = 4.65 \cdot 10^{-5}$ and $\beta = -103.3$ dB.

7.4.4. IMPACT OF SHAPE AND VOLUME

The maximum distance between two nodes is

$$\begin{aligned} r_{2D,MAX} &= \sqrt{Z_1^2 + Z_2^2} = Z_1\sqrt{1+\omega^2}, \\ r_{3D,MAX} &= \sqrt{Z_1^2 + Z_2^2 + Z_3^2} = Z_1\sqrt{1+\omega^2+\delta^2}. \end{aligned} \tag{7.21}$$

For a given value of the side Z_1 and the side length Z_2 (and thus ω), an increase of the height Z_3 (and thus δ), reduces the link density p_{3D} because the maximum distance $r_{3D,MAX}$ between two nodes increases, as also shown in (7.21). Figure 7.7 illustrates the difference $p_{2D} - p_{3D}$ in link density between the 2D and 3D spaces, which increases with δ. For $\delta \ll \omega$, e.g. $\omega = 1$ and $\delta = 0.1$, the effect of Z_3 (and thus δ) in (7.21) becomes negligible and hence $p_{2D} - p_{3D} \approx 0$. Additionally, an increase of δ shifts the maximum difference $p_{2D} - p_{3D}$ to a smaller $\frac{Z_1}{r_0}$ value because the shape of the prism becomes more symmetrical.

Similarly, based on (7.21), when considering a constant side Z_1 and height Z_3 (and thus δ), an increase of the side length Z_2 (and thus ω) increases the size of the rectangle

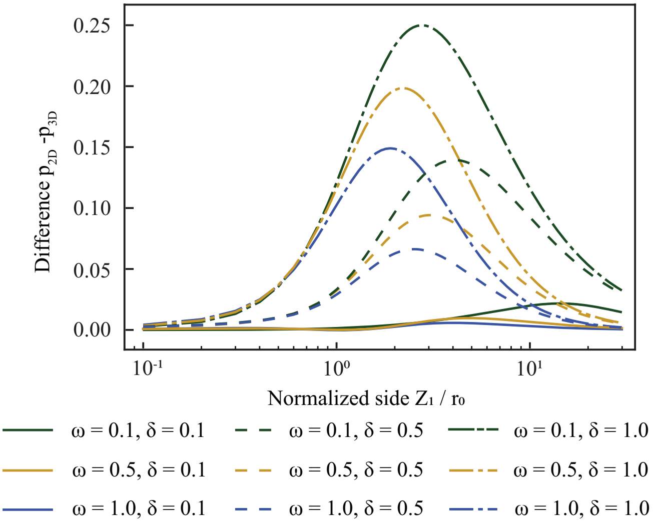


Figure 7.7: Difference $p_{2D} - p_{3D}$ of link density between the 2D and 3D spaces for different combinations of ω and δ when $\gamma = 2$, $r_c = 1$ m, $K = 4.65 \cdot 10^{-5}$ and $\beta = -103.3$ dB.

and prism in the horizontal plane, given by Z_1 and Z_2. Thus, the maximum distances $r_{2D,MAX}$ and $r_{3D,MAX}$ between two nodes increase and hence both the link densities p_{2D} and p_{3D} decrease. Figure 7.7 depicts that the difference $p_{2D} - p_{3D}$ also decreases with an increase of ω, which implies that p_{2D} reduces faster than p_{3D}.

7.5. Application to Drone Networks

In future telecommunication networks, drones are expected to provide coverage in a disaster area or when a ground base station fails or to serve incidental traffic hot spots. When a swarm of drones is deployed, the drones in the swarm are expected to communicate with each other in order to avoid collisions and exchange necessary information for collaborative tasks. Thus, any drone should be able to reach any other drone in the swarm to establish a connected network. While many studies in literature focus on deploying a swarm of drones to provide coverage and/or capacity to the access network, the connectivity among the drones is usually ignored [147][148]. In this section, the minimum number N_{MIN} of drones that need to be deployed for a connected network is computed, based on the link density p. We model the drone network with a RGG. Because drones can be deployed at the same altitude or at different altitudes, e.g. for scenarios where both terrestrial users and users in high-rise buildings are considered, the 2D and the 3D spaces are considered.

Previously we have shown that the link density p depends on γ, $\omega = \frac{Z_2}{Z_1}$, $\delta = \frac{Z_3}{Z_1}$ as well as on r_0 and thus on β. To evaluate the impact of each parameter, we refer to a baseline scenario S_0, which can describe a realistic drone network and we propose a set of scenarios by unilaterally varying the parameters of the baseline scenario to an extreme value, as shown in Table 7.2. We simulate 10000 realisations for each scenario and for each prism's size Z_1 and derive the link density p and the minimum number of nodes N_{MIN}, such that the network is connected. We measure connectivity via the giant component size, which equals the number of nodes in the largest cluster of the network divided by the total number of nodes N in the network. When the giant component size is equal to 1, the network is connected. The number of nodes in the largest cluster are found after first creating N clusters, where the n-th cluster contains the n-th node. Then, we merge the clusters that have at least one common node. We repeat the cluster mergers until no cluster shares a common node with another cluster. The largest cluster is the one that contains the most nodes. We regard the network as connected when the giant component size is greater than or equal to 0.99.

Table 7.2: Scenario configurations.

Scenario	γ	β	ω	δ
S_0	2	-127 dB	0.75	0.5
S_γ	4	-127 dB	0.75	0.5
S_β	2	-80 dB	0.75	0.5
S_ω	2	-127 dB	0.1	0.5
S_δ	2	-127 dB	0.75	0.1

Table 7.2 shows the chosen parameters for the considered scenarios. Scenario S_0 describes a network with a good propagation environment (e.g. in high altitude), with a typical minimum required channel gain β and drones placed in a space where the horizontal plane is larger than the vertical plane. In Scenario S_γ, the propagation environment is worse (i.e. the path loss exponent γ is higher) than the one in Scenario S_0, which can indicate that the drones are flying closer to the ground where there are many obstacles. The effects of a higher minimum required channel gain β compared to Scenario S_0 are captured in Scenario S_β, implying an increase of the noise power and/or a reduction of the SNR threshold and/or a reduction of the transmission power. Scenarios S_ω and S_δ capture the impact of the space where the drones are located. Specifically, in Scenario S_ω, the horizontal plane is narrower than the vertical plane while in Scenario S_δ, the vertical plane is much narrower than the horizontal plane.

Figure 7.8 shows the link density p and the minimum number of nodes N_{MIN} as derived from simulating the above-mentioned scenarios. The fitted curves in Figure 7.8

are deduced from all scenarios and follow a power law $N_{\mathrm{MIN}} = \alpha p^{\delta}$ where $(\alpha_{2\mathrm{D}}, \delta_{2\mathrm{D}})$ and $(\alpha_{3\mathrm{D}}, \delta_{3\mathrm{D}})$ are the fitting parameters with their standard error, in the 2D and 3D spaces, respectively:

$$\begin{aligned} \alpha_{2\mathrm{D}} &= 5.14 \pm 0.32, \delta_{2\mathrm{D}} = -1.12 \pm 0.04, \\ \alpha_{3\mathrm{D}} &= 4.35 \pm 0.27, \delta_{3\mathrm{D}} = -1.23 \pm 0.04, \end{aligned} \tag{7.22}$$

The RMSE of the fitting is 2.96 and 2.73 for the 2D and 3D spaces, respectively.

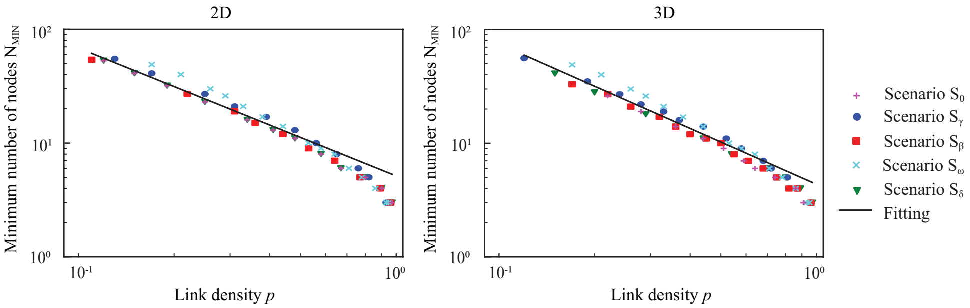


Figure 7.8: Minimum number of nodes N_{MIN} needed for connectivity in respect to the link density p, for the scenarios shown in Table 7.2, as well as the fitting curves given by (7.22).

Figure 7.8 shows that the minimum number of nodes N_{MIN} needed for connectivity does not depend on the dimension D of the space (2D or 3D), but only on the value of the link density p. However, N_{MIN} varies per scenario when the link density is less than about 0.5. For example, when comparing Scenarios S_0 and S_ω, a reduction of the parameter ω from 0.75 to 0.1, increases the minimum number of nodes N_{MIN} by about 10, when the link density is equal to 0.2. Also, for link density $p > 0.7$, the fitted curves are overestimating the minimum number of nodes N_{MIN}. Therefore, we can conclude that (7.22) provides a rough approximation of the minimum number of nodes N_{MIN}.

7.6. CONCLUDING REMARKS

We have computed the link density in D-dimensional RGGs, generated by a general distance function $f(r)$ and we have demonstrated its remarkably accurate approximation by the Fréchet distribution function (7.17) for any path loss exponent γ and any prism geometry. Also, we indicated that the link density $p_{2\mathrm{D}}$ in the 2D space upper bounds that in the 3D space, when the same propagation environment and size of the horizontal plane are considered. Finally, based on the giant component size, we have found that the minimum number of nodes N_{MIN} needed for connectivity is a power law of the link density p. The above-mentioned insights can be helpful in applications requiring the deployment of a swarm of drones. For example, when the size of and the propagation conditions in a disaster or crowded area that require coverage or extra capacity are known, N_{MIN} provides an estimation on the minimum number of drones that should be deployed to have a connected swarm of drones.

8

Conclusions

"The more I learn, the more I realise how much I don't know."

Albert Einstein.

In this dissertation, the challenging task of resource management in the RAN has been addressed in three distinct areas. Specifically, we modelled and simulated realistic 5G networks and provided better understanding on how different technology features deal with the problem of serving traffic with diverse requirement. Moreover, we focused on performing federated learning in resource-constrained wireless networks. We proposed and assessed a general agent selection framework and then extended it to address the joint agent selection and resource allocation problem in vehicular networks. Finally, we provided an approximation on the number of drones in a swarm, by modelling the drone swarm with a RGG. In this final chapter, the main contributions of this dissertation and the directions for future work are presented.

8.1. MAIN CONTRIBUTIONS

In the first part of this dissertation, we focused on serving traffic with diverse requirements. RAN slicing is a novel concept introduced in 5G networks, which allows to optimally configure network slices. MNOs can then configure a slice per customer or per traffic class based on its required QoS. In Chapter 3 we evaluated the performance of RAN slicing in terms of the traffic handling capacity. For the evaluation, we modelled an Industry 4.0-inspired scenario with realistic traffic models and requirements, as defined by 3GPP. We then compared a sliced network with optimally configured isolated slices to an optimally configured non-sliced network. Simulations showed that 20% more traffic can be handled by the non-sliced network. This result highlights that the inherent trunking loss incurred when configuring slices with dedicated resources can harm the traffic handling capacity.

To improve the performance of RAN slicing, in Chapter 4, we allowed slices to use the idle resources of other slices. Moreover, we assessed the merit of flexible numerology and mini-slots, in both sliced and non-sliced networks. For the evaluations, we modelled a challenging smart city environment, again based on traffic models and requirements from 3GPP. The results revealed that when RAN slicing is appropriately configured with flexible numerology and mini-slots, it provides better service performance than non-sliced networks. Moreover, it was shown that the slice-specific numerology should not be configured based on solely the service type of the respective slice. There is no doubt that RAN slicing enables the MNOs to configure slices for specific services and customers and improves network manageability. However, to provide good service performance in a sliced RAN, effective radio resource management should be performed in terms of e.g. sharing idle resources among slices and appropriately configuring complementary RAN features.

In the second part of the dissertation we focused on FL in resourced-constrained wireless networks. In Chapter 5 we proposed a metric to characterize an agent based on its importance in the learning process and its resource consumption in terms of transmission, processing and energy resources. We then proposed a general agent selection framework, in terms of an optimisation problem, that can be adapted to the needs and constraints of the network and the agents. Focusing on the trade-off between learning and wireless communication performance measures, we compared the agent selection policies derived as solutions of the optimization problem to other baseline selection policies. We showed that the proposed agent selection policies, which consider both learning and wireless channel aspects can provide higher global model accuracy. Moreover, we showed that pure learning-based policies can perform well in scenarios where the agents have a limited number of training samples or where the latency requirement is very stringent. Additionally, the results showed that the local loss is a good metric to characterise the importance of an agent in the learning process, in scenarios with non-IID data. Finally, it was revealed that the total energy consumption is dominated by the training-related consumption, rather than the transmission-related consumption.

In Chapter 6 we extended the previously proposed agent selection framework to address the joint agent selection and resource allocation problem. Specifically, we addressed this problem in the context of vehicular communications and assuming that the BSs are MU-MIMO capable. We then proposed the VBI algorithm to approximate the solution of the defined problem. For the evaluation of the VBI algorithm we considered a vehicular scenario based on road and mobility models from 3GPP and on the learning task of object classification of European traffic signs. The results showed that MU-MIMO capable BSs improve the convergence time of the global model because they enhance the quality of the wireless channels and they increase the number of vehicles that can be selected for training in a given communication round. Moreover, it was demonstrated that the convergence time increases when vehicles have different data set sizes. Furthermore, it was shown that the local loss is a good metric to characterise the importance of a vehicle in the learning process, only in the scenario where all vehicles had the same data set sizes. This was due to having vehicles with very small data set sizes, which then provide an inaccurate calculation of their local loss.

Chapter 7 addressed the number of drones that need to be deployed in a swarm to ensure that all drones in the swarm can communicate with each other. Considering RGGs with a distance-based connection function, we derived an expression for the link density in a D-dimensional prism. Focusing on the 2D and 3D spaces, we showed that the complementary distribution function of the Fréchet distribution approximates remarkably well the link density, regardless of the path loss exponent and the prism geometry. Moreover, we showed that the link density in the 2D space upper bounds that in the 3D space, considering the same environment and horizontal plane size. Finally, we modelled the drone swarm with a RGG and we approximated the minimum number of drones to ensure a connected swarm in both 2D and 3D spaces.

8.2. DIRECTIONS FOR FUTURE WORK

Considering that RAN slicing will be adopted by the MNOs, it is crucial to further investigate its performance in realistic scenarios. In our analysis in Chapter 4 we considered traffic with variability on a very fine time scale, which is handled by the scheduler and/or by the idle resource sharing feature. However, traffic with high variability over time should be considered and the concept of dynamic resource assignment between slices should be investigated. Potentially, dynamic resource assignment can be performed with input from a traffic forecasting algorithm.

In Chapter 5 we showed that the optimal agent selection policy depends on the scenario and communication round. Therefore, a promising direction for future work is to design adaptive agent selection policies to further improve the convergence time and level of the global model. Additionally, deeper understanding of the learning model is needed. For example, by analysing the CNN (or any other neural network architecture), we can identify which weights are important for the learning. Then, the importance of an agent in the learning process can be expressed in terms of how its data set impacts those specific weights. Moreover, the metric describing the importance of an agent in the learning process should also be robust when an agent has a very small data set size. In Chapter 6 we performed periodic resource allocation from a higher time scale perspective. However, it is important that resource allocation is also considered on the ms-time scale, especially in scenarios with mobile agents. Additionally, further work is also needed from the application level perspective. Even though many applications of FL in vehicular networks are mentioned in literature, specific accuracy targets and latency constraints need to be defined per application. Finally, more real-world data sets and scenarios should be considered for investigation. For example, in both Chapters 5 and 6 we considered that agents/vehicles maintain the same data set over time. However, in reality, the data sets change over time, e.g. because the vehicles collect data via sensors while driving.

In Chapter 7 we showed that the Fréchet distribution (7.17) very accurately approximates the link density. However, a method to easily derive the fitting values for different scenarios is left for further research. Moreover, in our analysis we assumed that the fixed nodes are independent and uniformly distributed in the D-dimensional space. To better model wireless networks, the node mobility and the spatial distribution should be considered, which will eventually influence the minimum number of nodes needed for connectivity. Additionally, in most wireless networks, the links are dependent due to interference. Thus, the impact of interference and potential ways to mitigate interference should also be addressed.

A

Appendix of Chapter 6

A.1. Analysis on Antenna Arrays

A.1.1. Cell Edge Beam Downtilt

Based on the angle of the beam on the azimuth and elevation planes, a certain area on the ground can be served. A simplified model to calculate the area on the ground which is served by a beam, is to assume a trapezoid ground area, whose size depends on the HPBW $\Delta\varphi_M$ and the downwards elevation angle θ of the beam [149]. Figure A.1 illustrates on the left hand side the azimuth projection of the trapezoid ground area, traced with a blue colour. The right hand side of Figure A.1 shows with a solid blue line the direction θ of a given beam and the distances d_1 and d_2 defining the size of the trapezoid. Specifically, distance d_1 denotes the distance from the BS until the small base of the trapezoid, hence $d_1 \geq 0$, and distance d_2 denotes the distance from the BS until the big base of the trapezoid. Both the distances d_1 and d_2 are a function of the beam direction $\frac{\Delta\varphi_M}{2} \leq \theta \leq 90° - \frac{\Delta\varphi_M}{2}$. Additionally, the angles θ_{d_1} and θ_{d_2} defining the distances d_1 and d_2 are equal to $\theta_{d_1} = \theta + \frac{\Delta\varphi_M}{2}$ and $\theta_{d_3} = \theta - \frac{\Delta\varphi_M}{2}$. Then, the distances d_1 and d_2, for the range $\frac{\Delta\varphi_M}{2} \leq \theta \leq 90° - \frac{\Delta\varphi_M}{2}$, are given in degrees by

$$d_1 = \frac{h_M}{\tan\left(\theta + \frac{\Delta\varphi_M}{2}\right)}, \tag{A.1}$$

$$d_2 = \frac{h_M}{\tan\left(\theta - \frac{\Delta\varphi_M}{2}\right)}. \tag{A.2}$$

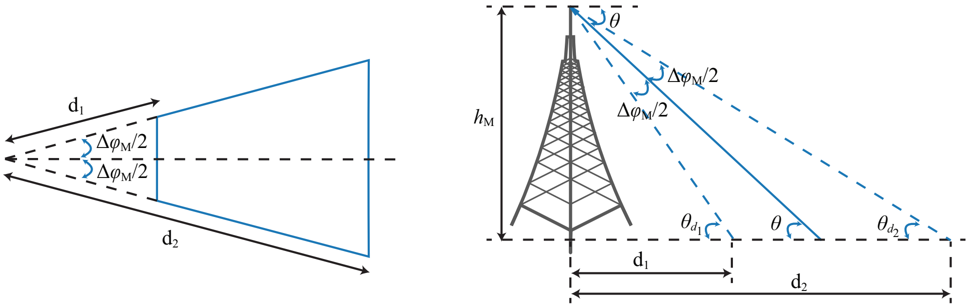


Figure A.1: The trapezoid ground area served by a beam with a downwards elevation angle θ and with HPBW $\Delta\varphi_M$, is given by the distances d_1 and d_2.

The cell edge is served by the beam with the smallest downtilt in the elevation plane. Using the method of the trapezoid ground area, we approximate the downtilt θ_{CE} of the beam serving the cell edge. Using a hexagonal layout, the distance d_c from the BS until the cell edge is given by

$$d_c = \frac{2}{3}\text{ISD}, \tag{A.3}$$

where ISD is the inter-site distance. Setting $d_2 = d_c$ in (A.2), the downtilt θ_{CE} is given by

$$\theta_{\text{CE}} = \arctan\left(\frac{h_M}{d_c}\right) + \frac{\Delta\varphi_M}{2}. \tag{A.4}$$

A.1.2. BEAM DIRECTIONS

In this work we consider 4 × 4 UPRAs for the BSs and thus $N_{E,M} = 16$. Then, the HPBW $\Delta\varphi_M$, using (6.4), and the FNBW $\varphi_{0,M}$, using (6.5), are given by:

$$\Delta\varphi_M \approx \sqrt{\frac{3}{16}} \approx 0.43\text{ rad} \approx 25°,$$

$$\varphi_{0,M} = 2\left[\frac{\pi}{2} - \cos^{-1}\left(\frac{2}{4}\right)\right] = \frac{\pi}{3}\text{ rad} = 60°.$$

Then, the angular resolution $\phi_{B,M} = \varphi_{0,M}/2 = 30°$ in both the azimuth and elevation planes. Considering three-sectorised cells, the antenna arrays need to cover a range of 120° in the azimuth plane. Consequently, there are $B_{A,M} = 120°/\phi_{B,M} = 4$ beams in the azimuth plane. On the left hand side of Figure A.2, the four beams covering the azimuth plane are shown, pointing at angles $-45°$, $-15°$, $15°$ and $45°$.

In the elevation plane, the antenna array needs to cover the vehicles on the ground and thus cover a range of θ less than 90°. Because each BS needs to cover a specific area on the ground, the beams need to point close to the cell edge to ensure that they do not introduce significant interference to the adjacent cells. Based on an ISD= $500m$

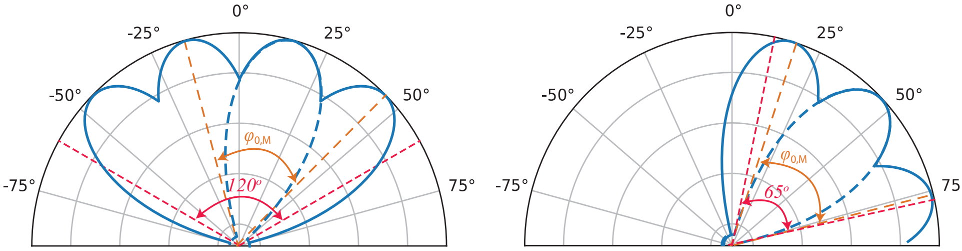


Figure A.2: Direction of beams in the GoB for an 4 × 4 UPRA, in the [left] azimuth and [right] elevation planes.

and antenna height $h_M = 25m$, the angle to the cell edge is equal to 4.3°. Therefore, the beams in the elevation plane need to cover a range of 85.7°. Consequently, there are $B_{E,M} = \lceil 85.7° / 30° \rceil = 3$ beams in the coverage range. As previously introduced, the beam direction for the cell edge is given by (A.4), and therefore the cell edge beam is at $\theta_{\text{CE}} \approx 17°$. Based on the angular resolution $\phi_{B,M} = 30°$, the other two beam directions are 47° and 77°. The right hand side of Figure A.2 illustrates the three beams and their direction.

Based on the derived beam directions, the UPRA at the BSs can simultaneously form and transmit $B_M = B_{A,M} B_{E,M} = 12$ beams. Furthermore, using (6.7), the maximum beam gain $G_{M,\text{MAX}} = 10\log(16) = 12$ dBi and using (6.8) and (6.9) we calculate $\text{FBR}_M = \text{SLL}_M = 19.1$ dB.

A.1.3. Beam Connection Time

A vehicle stays connected to the same beam when moving in the area covered by the given beam. The area served by a beam can be calculated using (A.1) and (A.2) when the HPBW $\Delta\varphi_M$ is substituted by the angular resolution $\phi_{B,M}$. Figure A.3 shows for a given azimuth direction, the ground coverage area of each of the three elevation beams calculated in Appendix A.1.2. Each beam coverage area is a trapezoid and it is described by the distances d_1 and d_2, as defined in Appendix A.1.1. Hence, the two bases and the height of the trapezoid can be calculated. Table A.1 shows the length of the distances d_1 and d_2 as well as the length of the long base and the height of the trapezoid, considering the angular resolution $\phi_{B,M} = 30°$. [1].

Table A.1 also shows a rough approximation of up to how much time a vehicle will stay connected to the same beam, which depends on the elevation angle of the beam. The maximum connection time interval is calculated as the length of the long base and

[1] The distance d_2 for the beam pointing to the cell edge was set equal to the cell edge distance d_c, as derived in (A.3).

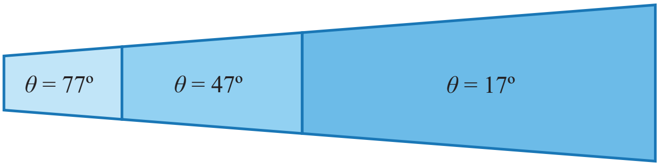


Figure A.3: Ground coverage area for every beam in the elevation plane (not in scale), at a given azimuth direction.

Table A.1: Geometry description of the trapezoid coverage areas and beam connection time derivation, for the given beam directions.

Beam direction θ	Distance d_1 [m]	Distance d_2 [m]	Long base [m]	Height [m]	Maximum connection time [s]
16.8°	40.3	333.3	172.5	283.0	10.4 – 17.0
46.8°	13.4	40.3	20.9	26.0	1.3 – 1.6
76.8°	0.0	13.4	6.9	12.9	0.4 – 0.8

the height of the trapezoid divided by the speed of the car, which is assumed to be 60 km/h.

A.1.4. Vehicle Beam

Each vehicle is equipped with a 2 × 2 UPRA and thus $N_{E,V} = 4$ antenna elements. Thus, the HPBWs $\Delta\varphi_V \approx 50°$, as given by (6.4). Moreover, using (6.11), the maximum beam gain $G_{V,\text{MAX}} = 10\log(4) = 6$ dBi. Finally, we assume that the beam can be steered in any direction.

A.2. Simulation Setup

A.2.1. Coverage Analysis

Based on the beam directions, derived in Appendix A.1.2, a coverage analysis is carried to ensure that all locations in the study area are covered in the DL channel. Using the previously described antenna model for the BSs and propagation environment from Section 6.4, we calculate the SSB RSRP on beam b for every location v as

$$RSRP_{vb} = P_{\text{SSB}} + G_{M,vb} + G_{T,vb}, \tag{A.5}$$

where P_{SSB} denotes the transmit power per resource element in dBm, which depends on the maximum transmit power $P_{\text{M,MAX}}$ and power P_{SSB} is given by (2.7). We assume that the reference signals are pairwise code-multiplexed on common time-frequency resources [33] and hence equally share the transmit power P_{SSB}. Moreover, we consider that coverage is provided to a location v if for at least one beam $RSRP_{vm} \geq -120$ dBm.

We consider a bandwidth of $B = 50$ MHz and hence $N_{\text{PRB}} = 270$ PRBs, considering numerology $\mu = 0$. Also, the maximum transmit power $P_{\text{M,MAX}} = 49$ dBm at the BSs. Using (2.7) we calculate $P_{\text{SSB}} = 10.89$ dBm. For a grid of $5m \times 5m$ pixels, Figure A.4 shows the RSRP values for each pixel in the considered area. The black grid on the RSRP map represents the modelled roads and hence the potential vehicle locations. Figure A.4 also shows that the RSRP level at every pixel is above the -120 dBm threshold and thus coverage is achieved. Finally, all of the roads are covered by the beams pointing at the cell edge which allows to only consider four beams per cell in the optimisation problem in Section 6.3.

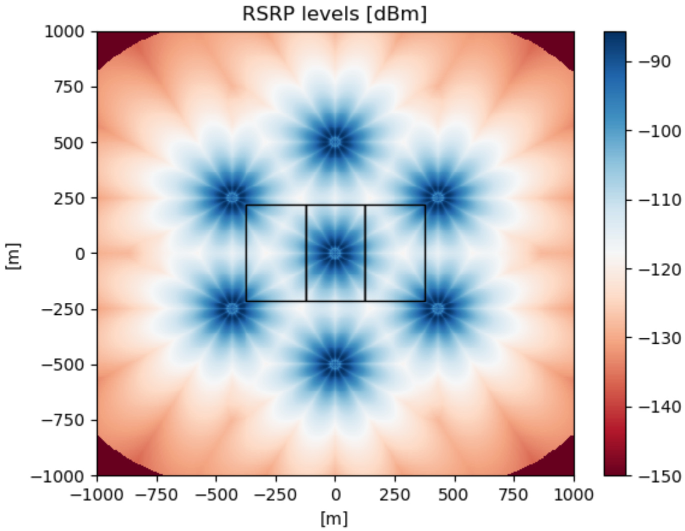


Figure A.4: RSRP levels in dBm at the considered area, assuming a grid of $5m \times 5m$ pixels. The black grid represents the roads.

A.2.2. BROADCAST BIT RATE

The broadcast bit rate is chosen such that the associated SNR requirement can be met. In this work, we associate the broadcast bit rate to the cell edge bit rate and hence to the cell edge SNR Γ_{CE}

$$\Gamma_{\text{CE}} = P_B + G_T + G_{V,\text{MAX}} + G_M - P_{\text{NOISE}} - NF_V, \tag{A.6}$$

where P_B denotes the broadcast beam transmit power and and $NF_V = 9$ dB is the noise figure at the vehicles [125]. Assuming that all four beams per cell are used for the broadcast and that the maximum transmit power $P_{M,\mathrm{MAX}} = 49$ dBm is equally shared among the four beams, we calculate the power $P_B = 43$ dBm. The distance to the cell edge $d_c = 333m$ and thus the path gain $G_T = -137.8$ dB. Moreover, from Appendix A.1.4, the maximum beam gain $G_{V,\mathrm{MAX}} = 6$ dBi. For the transmit antenna gain G_M we assume the worst case scenario that the angle off the boresight direction of the beam is equal to $\varphi = \Delta\varphi_M/2$ and $\vartheta = \varphi_{0,M}/2$ in the elevation and azimuth planes, respectively. Using 6.10, the transmit antenna gain $G_M = 6$ dB. The noise power is given by

$$P_N = -174 + 10\log_{10}(B)\,, \tag{A.7}$$

and for a bandwidth of $B = 50$ MHz, the noise power $P_{\mathrm{NOISE}} = -97$ dBm. Then, using (A.6), the cell edge SNR $\Gamma_{\mathrm{CE}} = 5.2$ dB and hence the boadcast bit rate is calculated, using (6.2), at 105 Mbps.

B

Appendix of Chapter 7

B.1. Link density in a rectangular hyperprism

When nodes are placed uniformly at random inside the prism, the link density p, given by

$$p = \frac{E[L]}{L_{\text{MAX}}} = \int_V dq \int_V ds\, g_r(q) g_r(s)\, f\left(|q-s|\right),$$

can be written as

$$p = \int_V \frac{dq}{v} \int_V \frac{ds}{v} f(|q-s|), \tag{B.1}$$

where $q = (x_1, x_2, \ldots, x_D)$ and $s = \left(y_1, y_2, \ldots, y_D\right)$ are the coordinates of two random nodes and $f(|q-s|)$ is the probability that two nodes at distance $|q-s|$ are connected by a link. In Cartesian coordinates, the distance $|x-y|^2 = \sum_{d=1}^{D}(x_d - y_d)^2$. To simplify the notation, we denote $h\left(|x-y|^2\right) = f(|x-y|)$ and the integral in (B.1) becomes

$$p = \int_0^{Z_1} dx_1 \int_0^{Z_1} dy_1 \cdots \int_0^{Z_D} dx_D \int_0^{Z_D} dy_D \frac{h\left(\sum_{d=1}^{D}(x_d - y_d)^2\right)}{\prod_{d=1}^{D} Z_d^2} \tag{B.2}$$

We use symmetry to reduce the $2D$-fold integral to a D-fold integral. We concentrate on the integration over the d dimension and denote $w_d^2 = \sum_{k=1;k\neq d}^{D}(x_k - y_k)^2$ that is independent of dimension d (i.e. of x_d and y_d),

$$\int_0^{Z_d} dx_d \int_0^{Z_d} dy_d\, h\left(\left(x_d - y_d\right)^2 + \sum_{\substack{k=1\\k\neq d}}^{D} (x_k - y_k)^2\right) \\
= \int_0^{Z_d} dx_d \int_0^{Z_d} dy_d\, h\left(\left(x_d - y_d\right)^2 + w_d^2\right).$$

After substitution $u_d = x_d - y_d$, where y_d is kept constant, followed by partial integration, we obtain

$$\int_0^{Z_d} dx_d \int_0^{Z_d} dy_d\, h\left((x_d-y_d)^2 + \sum_{\substack{k=1\\k\neq d}}^{D}(x_k-y_k)^2\right)$$
$$= 2\int_0^{Z_d} du_d\ (Z_d-u_d)\, h\left(u_d^2 + \sum_{\substack{k=1\\k\neq d}}^{D}(x_k-y_k)^2\right).$$

Any other dimension can be treated similarly and the integral in (B.2) is reduced to the integral

$$p = 2^D \int_0^{Z_1} du_1 \cdots \int_0^{Z_D} du_D \prod_{d=1}^{D} \frac{(Z_d-u_d)}{Z_d^2}\, h\left(\sum_{d=1}^{D} u_d^2\right). \tag{B.3}$$

We can calculate the link density (B.3) numerically for any dimension D and analytically for $D = 2$,

$$p_{\text{rect-2D}} = \frac{4}{(Z_1Z_2)^2}\int_0^{Z_1} du_1 \int_0^{Z_2} du_2\ (Z_1-u_1)(Z_2-u_2)\ h\left(u_1^2+u_2^2\right) \tag{B.4}$$

Transformed to polar coordinates $p_{\text{rect-2D}} = \frac{4(p_{\text{A}}+p_{\text{B}}+p_{\text{C}})}{(Z_1Z_2)^2}$ where

$$p_{\text{A}} = \int_0^{Z_2} h(r)\, r\, dr \int_0^{\frac{\pi}{2}} (Z_1 - r\cos\theta)\,(Z_2 - r\sin\theta)\, d\theta,$$
$$p_{\text{B}} = \int_{Z_2}^{Z_1} h(r)\, r\, dr \int_0^{\arcsin\left(\frac{Z_2}{r}\right)} (Z_1 - r\cos\theta)\,(Z_2 - r\sin\theta)\, d\theta,$$
$$p_{\text{C}} = \int_{Z_1}^{\sqrt{Z_1^2+Z_2^2}} h(r)\, r\, dr \int_{\arccos\left(\frac{Z_1}{r}\right)}^{\arcsin\left(\frac{Z_2}{r}\right)} (Z_1 - r\cos\theta)\,(Z_2 - r\sin\theta)\, d\theta.$$

Solving the θ-integrals of p_{A}, p_{B} and p_{C} separately, with

$$\begin{aligned} Q(\theta) &= \int (Z_1 - r\cos\theta)\,(Z_2 - r\sin\theta)\, d\theta \\ &= Z_1Z_2\theta + Z_1 r\cos\theta - Z_2 r\sin\theta - \frac{r^2}{4}\cos(2\theta) + c \end{aligned}$$

where c is an integration constant, leads to the link density in a rectangle

$$p_{\text{rect-2D}} = \frac{4}{Z_1^2Z_2^2}\left(p_{\text{A}}+p_{\text{B}}+p_{\text{C}}\right) \tag{B.5}$$

with

$$
\begin{aligned}
p_{\mathrm{A}} &= \int_0^{Z_2} h(r)\, r \left[\frac{Z_1 Z_2 \pi}{2} - (Z_1 + Z_2) r + \frac{r^2}{2} \right] dr, \\
p_{\mathrm{B}} &= \int_{Z_2}^{Z_1} h(r)\, r \left[\frac{Z_1 Z_2 \pi - Z_2^2}{2} - Z_1 r Z_1 Z_2 \arccos\left(\frac{Z_2}{r}\right) + Z_1 \sqrt{r^2 - Z_2^2} \right] dr, \\
p_{\mathrm{C}} &= \int_{Z_1}^{\sqrt{Z_1^2+Z_2^2}} h(r)\, r \left[\frac{Z_1 Z_2 \pi - Z_1^2 - Z_2^2}{2} - Z_1 Z_2 \arccos\left(\frac{Z_1}{r}\right) - Z_1 Z_2 \arccos\left(\frac{Z_2}{r}\right) \right. \\
&\quad \left. + Z_1 \sqrt{r^2 - Z_2^2} + Z_2 \sqrt{r^2 - Z_1^2} - \frac{r^2}{2} \right] dr.
\end{aligned}
$$

B.2. LINK DENSITY APPROXIMATION IN A HYPERCUBE

For $Z_1 = Z_2 = \cdots = Z_D = Z$, (7.11) becomes

$$p_{\mathrm{Dcube}} = \frac{2^D}{Z^{2D}} \int_0^Z du_1 \cdots \int_0^Z du_D \prod_{d=1}^{D} (Z - u_d)\, h\left(\sum_{d=1}^{D} u_d^2 \right).$$

We transform the integral from Cartesian to polar coordinates using the transformation in [150]. Because the boundaries of the hypercube are a bit more involved, we consider the integral over a part of the hypersphere of radius Z and centre at the origin, that is entirely enclosed by the hypercube,

$$
\begin{aligned}
p_{\mathrm{Dcircle}} = &\frac{2^D}{Z^{2D}} \int_0^R h(r)\, r^{D-1} dr \int_0^{\frac{\pi}{2}} d\varphi_1 \cdots \int_0^{\frac{\pi}{2}} d\varphi_{D-1} \prod_{d=1}^{D-1} \left(Z - r \cos\varphi_d \prod_{j=1}^{d-1} \sin\varphi_j \right) \\
&\times \left(Z - r \sin\varphi_{D-1} \prod_{j=1}^{D-2} \sin\varphi_j \right) \prod_{d=1}^{D-1} \sin^{D-1-d} \varphi_d
\end{aligned}
\tag{B.6}
$$

and provides a lower bound for p_{Dcube} of the hypercube.

B.2.1. TWO DIMENSIONS

Setting $D = 2$ and using $\varphi_1 = \theta$ in (B.6), assuming that the coordinates of the circle with radius Z are given in (r, θ), we find for the 2D space

$$p_{\mathrm{circle/4}} = \frac{4}{Z^4} \int_0^Z h(r)\, r\, dr \int_0^{\frac{\pi}{2}} (Z - r\cos\theta)(Z - r\sin\theta)\, d\theta. \tag{B.7}$$

Using (B.5) and the transformation $r = Zx$, (B.7) becomes

$$p_{\mathrm{circle/4}} = \int_0^1 h(Zx) \left(2\pi x - 8x^2 + 2x^3 \right) dx. \tag{B.8}$$

B.2.2. THREE DIMENSIONS

In the 3D space, we use $\varphi_1 = \theta$ and $\varphi_2 = \phi$, assuming that the coordinates of the sphere with radius Z are given in (r, θ, ϕ). Using (B.6) we find

$$\begin{aligned} p_{\text{sphere}/8} =& \frac{8}{Z^6} \int_0^Z h(r)\, r^2 dr \int_0^{\frac{\pi}{2}} \sin\theta\, (Z - r\cos\theta)\, d\theta \\ & \times \int_0^{\frac{\pi}{2}} \left(Z - r\cos\phi \sin\theta\right)\left(Z - r\sin\phi\sin\theta\right) d\phi. \end{aligned} \tag{B.9}$$

The ϕ-integral becomes:

$$\int_0^{\frac{\pi}{2}} (Z - r\sin\theta\cos\phi)(Z - r\sin\theta\sin\phi) d\phi = \frac{\pi}{2} Z^2 - 2rZ\sin\theta + \frac{r^2\sin^2\theta}{2}. \tag{B.10}$$

Substituting (B.10) in (B.9), we get

$$p_{\text{sphere}/8} = \frac{8}{Z^6} \int_0^Z h(r)\, r^2 dr \int_0^{\frac{\pi}{2}} \sin\theta\, (Z - r\cos\theta) \left(\frac{\pi}{2} Z^2 - 2rZ\sin\theta + \frac{r^2\sin^2\theta}{2}\right) d\theta.$$

After substitution of the θ-integral

$$\int_0^{\frac{\pi}{2}} \sin\theta (Z - r\cos\theta) \left(\frac{\pi}{2} Z^2 - 2rZ\sin\theta + \frac{r^2\sin^2\theta}{2}\right) d\theta = \frac{\pi Z^3}{2} - \frac{3\pi Z^2 r}{4} + Zr^2 - \frac{r^3}{8}$$

and letting $r = Zx$, we arrive at

$$p_{\text{sphere}/8} = \int_0^1 h(Zx) \left(4\pi x^2 - 6\pi x^3 + 8x^4 - x^5\right) dx. \tag{B.11}$$

B.2.3. FORMAL SOLUTION OF p_{DCIRCLE} IN (B.6) IN HIGHER DIMENSIONS

Both integrals (B.8) and (B.11) are of the form

$$p_{\text{Dcircle}} = \int_0^1 h(Zx)\, p_n(x)\, dx,$$

where $p_n(x) = \sum_{j=0}^n a_j x^j$ is a polynomial of degree n in x. The integral can be elegantly solved if $h(z)$ is an entire function[1]. Here, we confine to $h(z) = e^{-\beta z^\gamma}$, which is not an entire function if γ is not an integer. With $\alpha = \beta R^\gamma$, the general integral above becomes[2]

$$p_{\text{Dcircle}} = \sum_{j=0}^n a_j \int_0^1 e^{-\alpha r^\gamma} r^j dr. \tag{B.12}$$

[1] An entire (also called integral) function is a complex function without singularities in the finite complex plane (see [151, Chapter VIII]).

[2] We can transform the integral by letting $x = \alpha r^\gamma$ and $r = \left(\frac{x}{\alpha}\right)^{\frac{1}{\gamma}} = \alpha^{-\frac{1}{\gamma}} x^{\frac{1}{\gamma}}$, thus $dr = \alpha^{-\frac{1}{\gamma}} \frac{1}{\gamma} x^{\frac{1}{\gamma}-1} dx$ and

$$\int_0^1 e^{-\alpha r^\gamma} r^j dr = \frac{\alpha^{-\frac{j+1}{\gamma}}}{\gamma} \int_0^\alpha e^{-x} x^{\frac{j+1}{\gamma}-1} dx.$$

The right-hand side integral can be written in terms of the incomplete Gamma integral is $\Gamma(a,z) = \int_z^\infty e^{-x} x^{a-1} dx$, which is *not* an entire function.

From the definitions in [152, pp. 6.5.3, 6.5.4], $\Gamma(a,z) = \Gamma(a)\left(1 - z^a \gamma^*(a,z)\right)$ and [152, p. 6.5.29], it follows that

$$\gamma^*(a,z) = \frac{1}{\Gamma(a)} \sum_{k=0}^{\infty} \frac{(-z)^k}{k!\,(a+k)}. \tag{B.13}$$

A second powerful series is

$$\gamma^*(a,z) = e^{-z} \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(a+1+k)}. \tag{B.14}$$

The entire incomplete Gamma function $\gamma^*(a,z)$ is an entire function so that (B.13) and (B.14) converge for all a and all z. With this preparation, we return to the integral (B.12) and find, after Taylor expansion of the exponential and invoking (B.13)

$$\begin{aligned} p_{D\text{circle}} &= \sum_{j=0}^{n} a_j \sum_{k=0}^{\infty} \frac{(-\alpha)^k}{k!\,(j+1+\gamma k)} \\ &= \frac{1}{\gamma} \sum_{j=0}^{n} a_j \gamma^*\left(\frac{j+1}{\gamma}, \alpha\right) \Gamma\left(\frac{j+1}{\gamma}\right). \end{aligned}$$

The other series (B.14) leads to

$$p_{D\text{circle}} = \frac{e^{-\alpha}}{\gamma} \sum_{j=0}^{n} a_j \sum_{k=0}^{\infty} \frac{\Gamma\left(\frac{j+1}{\gamma}\right)}{\Gamma\left(\frac{j+1}{\gamma} + 1 + k\right)} \alpha^k.$$

Using $\frac{\Gamma(x+m)}{\Gamma(x)} = \prod_{l=0}^{m-1}(x+l)$ then results into a factorial series (see e.g. [153]),

$$p_{D\text{circle}} = e^{-\alpha} \sum_{j=0}^{n} \frac{a_j}{j+1} \sum_{k=0}^{\infty} \frac{\alpha^k}{\prod_{l=1}^{k}(\frac{j+1}{\gamma} + l)}$$

that converges for all α.

B.3. Fréchet fitting of the link density

The highly accurate Fréchet distribution for the link density

$$p(x) = 1 - e^{-\left(\frac{x-a}{b}\right)^{-c}},$$

has parameters a, b, c, that depend upon the path loss exponent γ and the prism geometry.

For the 2D space, Figure B.1 shows the influence of the path loss exponent γ on the fitting parameters $(a_{2\text{D}}, b_{2\text{D}}, c_{2\text{D}})$ for different values of $\omega = \frac{Z_2}{Z_1}$. In particular, Figure B.1 shows that $-1.5 < a_{2\text{D}} \leq 0$, $1 < b_{2\text{D}} < 3$ and $1 < c_{2\text{D}} < 3$. Also, for a given ω and γ it holds $|\frac{a_{2\text{D}}}{b_{2\text{D}}}| < 0.5$. Additionally, the plot of c_{2D} versus the path loss exponent γ in Figure

B.1(c) roughly illustrates two regimes for $\omega \leq 0.3$ and for $\omega > 0.3$. The lower values of $\omega \leq 0.3$ indicate the convergence of the 2D space towards the 1D space in which the link density p is more confined by the border effects. Hence, the Fréchet fitting is slightly less accurate for $\omega \leq 0.3$ than for higher values of $\omega > 0.3$ and reflected by the RMSE in Figure B.1(d), although the maximum RMSE < 0.01 is still very low.

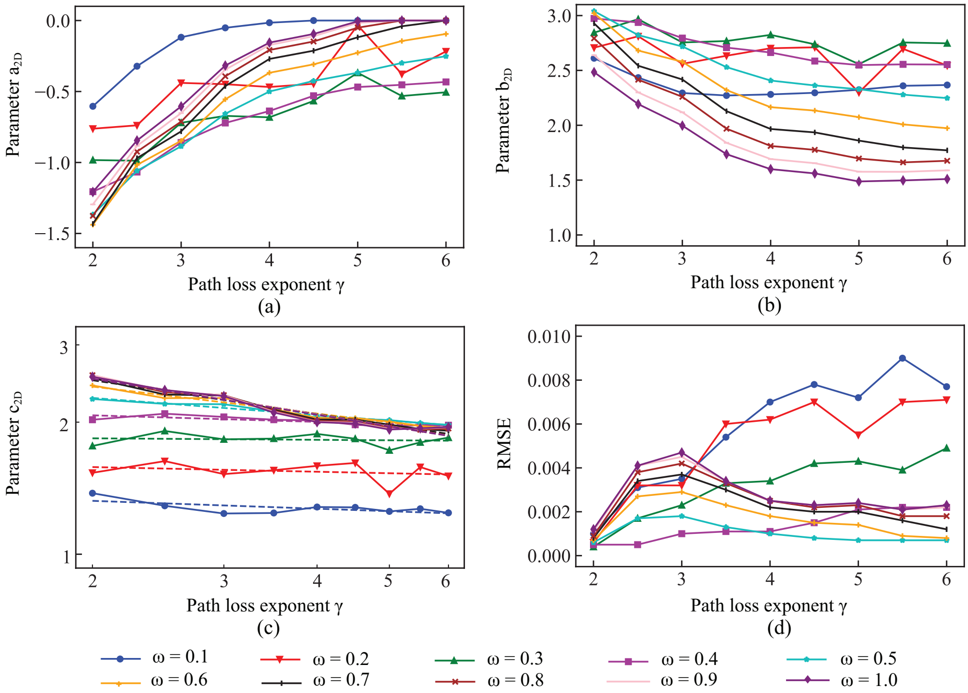


Figure B.1: Panels (a), (b) and (c) illustrate the fitting parameters $a_{2\text{D}}$, $b_{2\text{D}}$ and $c_{2\text{D}}$, respectively, for different values of ω and γ, while panel (d) illustrates the RMSE of each fit. Panel (c) is in log-log scale.

The parameter c_{2D} represents the shape of the Fréchet distribution and Figure B.1(c) exhibits that the parameter c_{2D} follows closely a power law $c_{2D}(\gamma) = \psi\gamma^{-\xi}$. The values and the standard error of the fitting parameters (ψ, ξ) and the RMSE of each fitting are given in Table B.1. Additionally, Figure B.1(c) illustrates that for $\omega > 0.3$, the parameter $c_{2D} \geq 2$ and approaches $c_{2D} \to 2$ for higher values of γ, because of the minimal border effects as explained in Section 7.3.2. Generally, when the border effects are minimal, the parameter c_{2D} is approximately equal to the dimensions $D = 2$.

A similar analysis for the 3D space in Figure B.2 illustrates the dependence of parameters $(a_{3\text{D}}, b_{3\text{D}}, c_{3\text{D}})$ on $\delta = \frac{Z_3}{Z_1}$ for $\gamma = 5$. Similarly to the 2D space, the parameter c_{3D} is approximately equal to the dimensions $D = 3$ for graphs where the border effects are minimal i.e. for large ω, δ and γ. Additionally, when $\omega \to 0$ and for large δ (to minimise

Table B.1: Fit values for parameter c_{3D}.

ω	ψ	ξ	RMSE
0.1	1.38 ± 0.05	0.06 ± 0.03	0.03
0.2	1.62 ± 0.11	0.04 ± 0.05	0.07
0.3	1.85 ± 0.08	0.01 ± 0.03	0.05
0.4	2.15 ± 0.04	0.05 ± 0.01	0.03
0.5	2.48 ± 0.03	0.13 ± 0.01	0.02
0.6	2.79 ± 0.05	0.21 ± 0.01	0.03
0.7	2.98 ± 0.07	0.26 ± 0.02	0.03
0.8	3.05 ± 0.09	0.27 ± 0.02	0.04
0.9	3.06 ± 0.10	0.28 ± 0.03	0.05
1.0	3.03 ± 0.10	0.27 ± 0.03	0.05

the border effects) the 3D space reduces to the 2D space and thus $c_{3D} \approx c_{2D} \approx 2$. Due to symmetry, when $\delta \to 0$ and for large ω, it again holds $c_{3D} \approx c_{2D} \approx 2$.

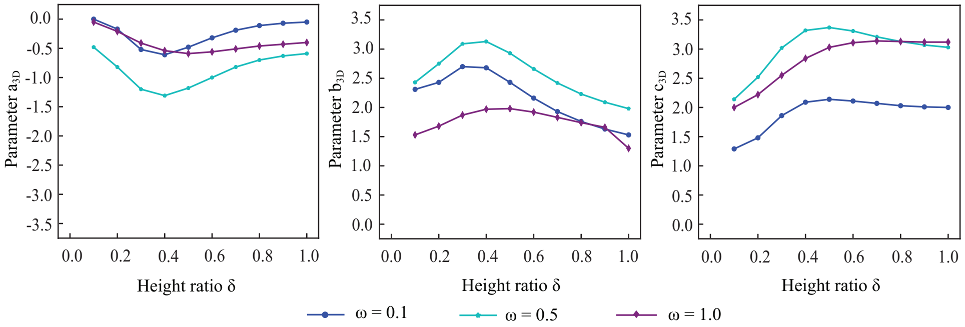


Figure B.2: Fitting parameters (a_{3D}, b_{3D}, c_{3D}) for different values of ω and δ and for $\gamma = 5$.

Nomenclature

Global

$\mathcal{B}_{\text{TOT}}$	Set of all beams in the network
$\mathcal{B}_m$	Set of beams at BS m
$\mathcal{M}$	Set of BSs in the network
$\Delta\varphi_k$	HPBW, where $k = M$ for the BSs and $k = V$ for the vehicles
Γ_{MIN}	Minimum SNR for connectivity
Γ_{ij}	SNR between a transmitter i and a receiver j
Γ_{jb}	SNR at receiver j for a transmission from beam b
$\hat{\Gamma}_{ij}$	SINR between a transmitter i and a receiver j
γ	Path loss exponent
λ	Wavelength
μ	Numerology
$\phi_{B,M}$	Angular resolution of the beams at the BSs
ψ_{SF}	Random variable for shadow fading
σ_{SF}	Standard deviation for normally distributed shadow fading
$\varphi_{0,k}$	FNBW, where $k = M$ for the BSs and $k = V$ for the vehicles
φ_{vb}	Angle off the boresight direction of a beam b, at receiver v
ϑ_{vb}	Negative elevation angle relative to the direction of beam b, at receiver v
B	System bandwidth
B_{TOT}	Total number of beams in the network, where $B_{\text{TOT}} = \lvert\mathcal{B}_{\text{TOT}}\rvert$
B_M	Number of beams at a BS, where $B_M = \lvert\mathcal{B}_m\rvert$

$B_{A,M}$	Number of beams in the azimuth plane at the BSs
$B_{E,M}$	Number of beams in the elevation plane at the BSs
C_{ij}	Channel capacity between a transmitter i and a receiver j
G_E	Gain per antenna element
G_i	Antenna gain at transmitter/receiver i
$G_{A,vb}$	Effective antenna gain in the azimuth plane for a transmission between vehicle v and beam b
$G_{E,vb}$	Effective antenna gain in the elevation plane for a transmission between vehicle v and beam b
$G_{k,\mathrm{MAX}}$	Maximum beam gain, where $k = M$ for the BSs and $k = V$ for the vehicles
$G_{M,vb}$	Antenna gain at the BS-side, for communication between vehicle v and beam b
$G_{S,M}$	Gain of the side lobe at the BSs
$G_{T,j}$	Transmission gain at receiver j
$G_{T,vb}$	Transmission gain for communication between vehicle v and beam b
$G_{V,vb}$	Antenna gain at the vehicle-side, for communication between vehicle v and beam b
H_{ij}	Channel response to multipath fading on the channel between transmitter i and receiver j
$L_{\mathrm{MF},ij}$	Experienced multipath fading loss between a transmitter i and a receiver j
$L_{\mathrm{PL},ij}$	Path loss between a transmitter i and a receiver j
$L_{\mathrm{SH},ij}$	Experienced shadow fading loss between a transmitter i and a receiver j
N_0	Noise power spectral density
N_{PRB}	Number of PRBs
N_{SUB}	Number of subcarriers per PRB
$N_{E,k}$	Number of antenna elements, where $k = M$ for the BSs and $k = V$ for the vehicles

NF_i	Noise figure at transmitter/receiver i
P_{NOISE}	Thermal noise power
$P_{\text{RX},ij}$	Received power of a transmission between a transmitter i and a receiver j
P_{SSB}	Transmit power per resource element in the SSB
$P_{\text{TX,MAX}}$	Maximum transmit power at an antenna
$P_{\text{TX},ij}$	Transmit power from a transmitter i to a receiver j
$P_{\text{TX},i}$	Transmit power of transmitter i
$P_{\text{V,MAX}}$	Maximum transmit power of vehicles
R_{jb}	Bit rate at receiver j for a transmission from beam b
R_j	Bit rate at receiver j
c	Speed of light
d_0	Reference distance for path loss
d_E	Distance between antenna elements
d_{ij}	3D distance between a transmitter i and a receiver j
d_j	3D distance between the BS and a receiver j
f_C	Carrier frequency

Related to Part I: Serving Traffic With Diverse Requirements

δ_n	Maximum allowed packet drop rate of QoS flow n
λ_{TO}	Arrival rate of TO flows
λ_{VR}	Arrival rate of VR sessions
$\lambda_{\text{BB},y}$	Arrival rate of BB sessions in channel direction $y \in \{\text{UL,DL}\}$
$\phi_S(\cdot)$	Weight function of scheduler S
τ_n	Latency budget of QoS flow n

N	Number of QoS flows in the network
N_{LC}	Number of LC flows in the network
N_x	Number of persistent sessions of service $x \in \{\text{VS,SM}\}$
$Q_{S,n}$	Scheduling metric of QoS flow n, considering a scheduler S
$\overline{R}_n$	Exponentially smoothed experienced bit rate of QoS flow n
S	Scheduler
S_{BB}	Packet size of BB sessions
S_{LC}	Packet size of LC flows
S_{TO}	Packet size of TO flows
$S_{x,y}$	Packet size of service $x \in \{\text{VS,SM}\}$ in channel direction $y \in \{\text{UL,DL}\}$
$T_{x,y}$	Arrival period of packets of service $x \in \{\text{VS,SM}\}$ in channel direction $y \in \{\text{UL,DL}\}$
W_n	Head-of-line packet latency of QoS flow n
t_c	Smoothing parameter

Related to Part II: Collaborative Learning

$\mathcal{K}$	Set of the training data across all agents/vehicles
$\mathcal{K}_v$	Set of the training data of agent/vehicle v
$\mathcal{K}_{T,v}$	Set of the testing data of agent/vehicle v
$\mathcal{V}$	Set of agents/vehicles in the network
$\mathcal{V}_G$	Set containing the selected agents/vehicles selected
δ_v	Deviation between the local and global model at agent/vehicle v
η	Learning rate
ν_v	CPU clock frequency at agent/vehicle v
ω_v	Number of FLOPs per cycle at agent/vehicle v
ρ	Constant tuning the importance of a vehicle on each beam

ρ_E	Constant tuning the relevant significance of the energy resource consumption
ρ_L	Constant tuning the relevant significance of the learning importance
ρ_R	Constant tuning the relevant significance of the transmission resource consumption
ρ_T	Constant tuning the relevant significance of the processing resource consumption
τ_{AGG}	Time to aggregate all local models at the FL server
$\tau_{\text{APP,MAX}}$	Application-specific latency budget
τ_{DL}	Time for broadcasting the global model
τ_{SCH}	Time period of scheduling and notification of agents/vehicles
$\tau_{\text{T+UL}}$	Time for all scheduled agents/vehicles to train and upload their local models
$\tau_{\text{L},b}$	Upload start time at beam b
$\tau_{\text{T},v}$	Training time of agent/vehicle v
$\tau_{\text{UL},vb}$	Upload time of agent/vehicle v on beam b
$\tau_{\mathbf{B}}$	A $B_{\text{TOT}} \times 1$ vector with upload latency budget for every beam
$\tau_{\mathbf{L}}$	A $B_{\text{TOT}} \times 1$ vector with the upload start time at each beam
$\tau_{\mathbf{T}}$	A $V \times 1$ vector with the training latency vehicles
$\mathbf{A}$	A binary $V \times B_{\text{TOT}}$ optimization matrix with the associations between the selected vehicles and the beams
$\mathbf{Q}$	A $V \times B_{\text{TOT}}$ matrix with the importance of each vehicle per beam
$\hat{\mathbf{T}}_{\mathbf{L}}$	A $V \times B_{\text{TOT}}$ auxiliary matrix
$\mathbf{T}_{\mathbf{UL}}$	A $V \times B_{\text{TOT}}$ matrix with the upload times of each vehicle at each beam
$\mathbf{W}_G$	The weights of the global model
$\mathbf{W}_v$	The weights of the model at agent/vehicle v
$\mathbf{X}_v$	The input data of agent/vehicle v
$\hat{\mathbf{Y}}_v$	The model output (predictions) of agent/vehicle v

$\mathbf{Y}_v$	The output data of agent/vehicle v
$\mathbf{s}_v$	A binary $V \times 1$ optimization vector for the selection of vehicles
$\mathbf{x}_{vk}$	The k^{th} input vector of agent/vehicle v
$\hat{\mathbf{y}}_{vk}$	The k^{th} predicted output vector of agent/vehicle v
$\mathbf{y}_{vk}$	The k^{th} output vector of agent/vehicle v
B_v	Transmission bandwidth allocated to agent/vehicle v
$C_{\mathrm{R,MAX}}$	Available transmission resources
$C_{E,v}$	Consumption of energy resources of agent/vehicle v
$C_{R,v}$	Consumption of transmission resources of agent/vehicle v
$C_{T,v}$	Consumption of processing resources of agent/vehicle v
E_v	Energy level at agent/vehicle v
$F(\cdot)$	The loss function of the model
K	Number of training data samples across all agents/vehicles, where $K = \lvert\mathcal{K}\rvert$
K_v	Number of training data samples at agent/vehicle v, where $K_v = \lvert\mathcal{K}_v\rvert$
$K_{T,v}$	Number of testing data samples at agent/vehicle v, where $K_{T,v} = \lvert\mathcal{K}_{T,v}\rvert$
M	Number of BSs in the network, where $M = \lvert\mathcal{M}\rvert$
Q_{TOT}	Total vehicle importance
V	Number of agents/vehicles in the network, where $V = \lvert\mathcal{V}\rvert$
Z	Size of the FL model
e_v	Energy consumption coefficient at agent/vehicle v
$f_k(\cdot)$	The loss function on sample k
g_{MIN}	Minimum required processing capabilities
g_v	Processing capabilities of agent/vehicle v
n_{LE}	Number of local iterations
n_C	Size of the output vector

n_X	Size of the input vector
$n_{\mathrm{CORES},v}$	Number of CPU codes at agent/vehicle v
$n_{\mathrm{FLOP,G}}$	Number of FLOPs to train the FL model for a batch size s_B
$q_{L,v}$	Importance of agent/vehicle v in the learning process
q_{vb}	Importance of agent/vehicle v on beam b
q_v	Importance of agent/vehicle v
s_v	Binary optimization variable for the selection of agent/vehicle v
s_B	Batch size

Related to Part III: Deployment of a Drone Swarm

$\mathcal{L}$	Set of links in a graph
$\mathcal{N}$	Set of nodes in a graph
$\mathcal{R}$	Set with the nodal positions
β	Minimum required channel gain
δ	Height ratio
ω	Side ratio
ρ	Node density
D	Dimension
$G(N,L)$	Graph with N nodes and L links
$G_p(N)$	Erdős-Rényi random graph with N nodes and link probability p
L	Number of link in a graph, where $L = \lvert\mathcal{L}\rvert$
L_{MAX}	Maximum number of link in a graph
N	Number of nodes in a graph, where $N = \lvert\mathcal{N}\rvert$
N_{MIN}	Minimum number of UAVs for a connected network
Z	Length of a hypercube in each dimension

Z_d	Length of a hyperprism in the d^{th} dimension
$f(r)$	Distance-based function for the connection probability of two nodes
p	Link density
p_x	Link density in a space described by x, e.g. x =2D for a 2D space
p_{ij}	Probability to have a link between nodes i and j
r	Distance between two nodes
r_0	Maximum allowed distance between to nodes such that they are connected
r_{MAX}	Maximum distance between two nodes
r_c	Reference distance for the antenna far field
r_i	Position of node i
v	Volume

Bibliography

[1] J. C. Maxwell, "A dynamical theory of the electromagentic field", *Philosophical Transactions of the Royal Society of London*, vol. 155, pp. 459–512, 1865. DOI: 10.1098/rstl.1865.0008.

[2] M. G. Raymer, *The silicon web: Physics for the Internet age*, 1st ed. CRC press, 2009.

[3] W. B. Carlson, *Tesla: Inventor of the electrical age*. Princeton university press, 2013.

[4] J. Q. Anderson, *Imagining the Internet: Personalities, predictions, perspectives*. Rowman & Littlefield, 2005.

[5] C. Moffett, "Marconi's wireless telegraph", *McClure's Magazine*, pp. 99–112, 1899.

[6] J. A. Poli, "Development and present trend of police radio communications", *Journal of Criminal Law and Criminology*, vol. 33, no. 2, pp. 193–197, 1942.

[7] D. Ring, "Mobile telephony: Wide area coverage", *Bell Laboratories Technical Memorandum*, 1947.

[8] M. Cooper, R. W. Dronsuth, A. J. Leitich, C. N. Lynk Jr., J. J. Mikulski, J. F. Mitchell, R. A. Richardson, and J. H. Sangster, "Radio telephone system", US Patent US-3906166-A, 1973.

[9] V. K. Garg and J. E. Wilkes, *Wireless and personal communications systems*, 1st ed. Prentice Hall, 1995.

[10] A. A. Huurdeman, *The worldwide history of telecommunications*. John Wiley & Sons, 2003.

[11] M. Sauter, *From GSM to LTE-advanced pro and 5G : An introduction to mobile networks and mobile broadband*, 4th ed. John Wiley & Sons, 2021.

[12] T. Chapman, E. Larsson, P. von Wryca, E. Dahlman, S. Parkvall, and J. Sköld, *HSPA evolution: The fundamentals for mobile broadband*. Academic Press, 2015.

[13] B. A. Bjerke, "LTE-advanced and the evolution of LTE deployments", *IEEE Wireless Communications*, vol. 18, no. 5, pp. 4–5, 2011. DOI: 10.1109/MWC.2011.6056684.

[14] E. Dahlman, S. Parkvall, and J. Sköld, *4G, LTE-advanced pro and the road to 5G*, 3rd ed. Academic Press, 2016.

[15] M. Massaro and S. Kim, "Why is South Korea at the forefront of 5G? Insights from technology systems theory", *Telecommunications Policy*, vol. 46, no. 5, 2022. DOI: 10.1016/j.telpol.2021.102290.

[16] S. Ahmadi, *5G NR: Architecture, technology, implementation, and operation of 3GPP new radio standards*, 1st ed. Academic Press, 2019.

[17] "European vision for the 6G network ecosystem", 5G IA, White Paper, 2021.

[18] A. Mourad, R. Yang, P. H. Lehne, and A. de la Oliva, "Towards 6G: Evolution of key performance indicators and technology trends", in *2nd 6G Wireless Summit (6G SUMMIT)*, Levi, Finland, 2020. DOI: 10.1109/6GSUMMIT49458.2020.9083759.

[19] "Ericsson mobility report", Ericsson, Tech. Rep., 2023.

[20] "KPN, T-Mobile and VodafoneZiggo acquire frequencies in Dutch mobile communications auction". (2020), [Online]. Available: https://www.government.nl/latest/news/2020/07/21/kpn-t-mobile-and-vodafoneziggo-acquire-frequencies-in-dutch-mobile-communications-auction.

[21] "Empowering vertical industries through 5G networks: Current status and future trends", 5G PPP & 5G IA, White Paper, 2020.

[22] "IMT vision: Framework and overall objectives of the fuuture development of IMT for 2020 and beyong", ITU-R, Recommendation, 2015.

[23] "Service requirements for cyber-physical control applications in vertical domains", 3GPP, TS 22.104 v17.4.0, 2020.

[24] S. E. Elayoubi, S. B. Jemaa, Z. Altman, and A. Galindo-Serrano, "5G RAN slicing for verticals: Enablers and challenges", *IEEE Communications Magazine*, vol. 57, no. 1, pp. 28–34, 2019. DOI: 10.1109/MCOM.2018.1701319.

[25] "NR; Physical channels and modulation", 3GPP, TS 38.211 v16.3.0, 2020.

[26] "Study on New Radio (NR) access technology", 3GPP, TS 38.912 v17.0.0, 2022.

[27] Q. H. Spencer, C. B. Peel, A. L. Swindlehurst, and M. Haardt, "An introduction to the multi-user MIMO downlink", *IEEE Communications Magazine*, vol. 42, no. 10, pp. 60–67, 2004. DOI: 10.1109/MCOM.2004.1341262.

[28] "6G use case and analysis", NGMN, White Paper, 2022.

[29] "European vision for the 6G network ecosystem", 5G IA, White Paper, 2021. DOI: 10.5281/zenodo.5007671.

[30] "5G; Study on scenarios and requirements for next generation access technologies", 3GPP, TR 38.913 v17.0.0, 2022.

[31] "5G; NR; Base station (BS) radio transmission and reception", 3GPP, TS 38.104 v16.4.0, 2020.

[32] C. A. Balanis, *Antenna theory: analysis and design*, 4th ed. John Wiley and Sons Inc, 2016.

[33] E. Dahlman, S. Parkvall, and J. Sköld, *5G NR: The next generation wireless access technology*, 1st ed. Elsevier Academic Press, 2018.

[34] A. A. Zaidi, R. Baldemair, V. Molés-Cases, N. He, K. Werner, and A. Cedergre, "OFDM numerology design for 5G new radio to support IoT, eMBB and MBSFN", *IEEE Communications Standards Magazine*, vol. 2, no. 2, pp. 78–83, 2018. DOI: 10.1109/MCOMSTD.2018.1700021.

[35] "NR, physical layer procedures for control", 3GPP, TS 38.213 v16.3.0, 2020.

[36] "Mixed numerology in an OFDM system", Ericsson, R1-165833 3GPP TSG-RAN WG1 meeting #85, 2016.

[37] L. Liu, R. Chen, S. Geirhofer, K. Sayana, Z. Shi, and Y. Zhou, "Downlink MIMO in LTE-advanced: SU-MIMO vs. MU-MIMO", *IEEE Communications Magazine*, vol. 50, no. 2, pp. 140–147, 2012. DOI: 10.1109/MCOM.2012.6146493.

[38] L. D. Nguyen, H. D. Tuan, T. Q. Duong, and H. V. Poor, "Multi-user regularized zero-forcing beamforming", *IEEE Transactions on Signal Processing*, vol. 67, no. 11, pp. 2839–2853, 2019. DOI: 10.1109/TSP.2019.2905833.

[39] "5G TDD Synchronisation", GSMA, White paper, 2020.

[40] A. Goldsmith, *Wireless communications.* Cambridge University Press, 2005.

[41] F. Jesus, M. Sousa, F. Freitas, P. Vieira, A. Rodrigues, and M. P. Queluz, "Evaluating 5G coverage in 3D scenarios under configurable antenna beam patterns", in *Proceedings of Symposium on WPMC*, Okayama, Japan, 2021. DOI: 10.1109/WPMC52694.2021.9700421.

[42] S. E. Elayoubi, M. Fallgren, P. Spapis, G. Zimmermann, D. Martín-Sacristán, C. Yang, S. Jeux, P. Agyapong, L. Campoy, Y. Qi, and S. Singh, "5G service requirements and operational use cases: Analysis and METIS II vision", in *Proceedings of EuCNC*, Athens, Greece, 2016. DOI: 10.1109/EuCNC.2016.7561024.

[43] M. Raftopoulou and R. Litjens, "Optimisation of numerology and packet scheduling in 5G networks: To slice or not to slice?", in *Proceedings of VTC-Spring*, Helsinki, Finland, 2021. DOI: 10.1109/VTC2021-Spring51267.2021.9448814.

[44] A. Kandoi, M. Raftopoulou, and R. Litjens, "Assessment of 5G RAN features for integrated services provisioning in smart cities", in *Proceedings of Workshop on WiMob*, Thessaloniki, Greece, 2022. DOI: 10.1109/WiMob55322.2022.9941612.

[45] "Requirements for support of radio resource management", 3GPP, TS 38.133 v18.4.0, 2024.

[46] "User Equipment (UE) radio transmission and reception; Part 4: Performance requirements", 3GPP, TS 38.101-4 v16.4.0, 2021.

[47] F. Capozzi, G. Piro, L. A. Grieco, G. Boggia, and P. Camarda, "Downlink packet scheduling in LTE cellular networks: Key design numbers and a survey", *IEEE Communications Surveys & Tutorials*, vol. 15, no. 2, pp. 678–700, 2013. DOI: 10.1109/SURV.2012.060912.00100.

[48] C.-Y. Chang and N. Nikaein, "RAN runtime slicing system for flexible and dynamic service execution environment", *IEEE Access*, vol. 6, 2018. DOI: 10.1109/ACCESS.2018.2847610.

[49] S. Khatibi and A. Jano, "Elastic slice-aware radio resource management with AI-traffic prediction", in *Proceedings of EuCNC*, Valencia, Spain, 2019. DOI: 10.1109/EuCNC.2019.8801995.

[50] K. I. Pedersen, G. Pocovi, J. Steiner, and S. R. Khosravirad, "Punctured scheduling for critical low latency data on a shared channel with mobile broadband", in *Proceedings of VTC-Fall*, Toronto, ON, Canada, 2017. DOI: 10.1109/VTCFall.2017.8287951.

[51] A. Akhtar and H. Arslan, "Downlink resource allocation and packet scheduling in multi-numerology wireless systems", in *Proceedings of WCNCW*, Barcelona, Spain, 2018. DOI: 10.1109/WCNCW.2018.8369012.

[52] N. Bechir, M. Nasreddine, A. Mahmoud, H. Walid, and M. Sofien, "Novel scheduling algorithm for 3GPP downlink LTE cellular network", *Procedia Computer Science*, vol. 40, pp. 116–122, 2014. DOI: 10.1016/j.procs.2014.10.038.

[53] V. Nair, R. Litjens, and H. Zhang, "Optimisation of NB-IoT deployment for smart energy distribution networks", *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, 2019. DOI: 10.1186/s13638-019-1485-2.

[54] "Study on channel model for frequencies from 0.5 to 100 GHz", 3GPP, TR 38.901 v16.1.0, 2020.

[55] S. Jaeckel, L. Raschkowski, K. Börner, L. Thiele, F. Burkhardt, and E. Eberlein, "QuaDRiGa - Quasi deterministic radio channel generator, user manual and documentation", Fraunhofer Heinrich Hertz Institute, Tech. Rep., version 2.2.0, 2017.

[56] "NR, physical layer procedures for data", 3GPP, TS 38.214 v16.3.0, 2020.

[57] S. Pratschner, B. Tahir, L. Marijanovic, M. Mussbah, K. Kirev, R. Nissel, S. Schwarz, and M. Rupp, “Versatile mobile communications simulation: The Vienna 5G link level simulator”, *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, 2018. DOI: 10.1186/s13638-018-1239-6.

[58] K. Sayana, J. Zhuang, and K. Stewart, “Link performance abstraction based on mean mutual information per bit (MMIB) for the LLR channel”, IEEE 802.16 Broadband Wireless Access Working Group, IEEE C802.16m-07/097, 2007.

[59] K. I. Pedersen, G. Monghal, I. Z. Kovacs, T. E. Kolding, A. Pokhariyal, F. Frederiksen, and P. Mogensen, “Frequency domain scheduling for OFDMA with limited and noisy channel feedback”, in *Proceedings of VTC-Fall*, Baltimore, MD, USA, 2007. DOI: 10.1109/VETECF.2007.378.

[60] S. N. Anbalagan, R. Litjens, K. Das, A. Chiumento, P. Havinga, and J. L. van den Berg, “A sensitivity analysis on the potential of 5G channel quality prediction”, in *Proceedings of VTC-Spring*, Helsinki, Finland, 2021. DOI: 10.1109/VTC2021-Spring51267.2021.9448944.

[61] “Summary of DL/UL scheduling and HARQ management”, Qualcomm Incorporated, Reno, USA, R1-1721652, 3GPP TSG-RAN WG1 Meeting #91, 2017.

[62] “New services and applications with 5G ultra-reliable low latency communications”, 5G Americas, White Paper, 2018.

[63] “UP latency in NR”, Ericsson, Prague, Czech Republic, R2-1711550, 3GPP TSG-RAN WG2 Meeting #99bis, 2017.

[64] M. Raftopoulou, *Slicesim: System-level simulator for RAN slicing*, version 1.0, 2023. DOI: 10.5281/zenodo.10204995.

[65] J. Navarro-Ortiz, P. Romero-Diaz, S. Sendra, P. Ameigeiras, J. J. Ramos-Munoz, and J. M. Lopez-Soler, “A survey on 5G usage scenarios and traffic models”, *IEEE Communications Surveys & Tutorials*, vol. 2, no. 22, pp. 905–929, 2020. DOI: 10.1109/COMST.2020.2971781.

[66] B. N. Silva, M. Khan, and K. Han, “Towards sustainable smart cities: A review of trends, architectures, components, and open challenges in smart cities”, *Sustainable Cities and Society*, vol. 38, pp. 697–713, 2018. DOI: 10.1016/j.scs.2018.01.053.

[67] H. Yin, L. Zhang, and S. Roy, “Multiplexing URLLC traffic within eMBB services in 5G NR: fair scheduling”, *IEEE Transactions on Communications*, vol. 69, no. 2, pp. 1080–1093, 2021. DOI: 10.1109/TCOMM.2020.3035582.

[68] T. Bag, S. Garg, Z. Shaik, and A. Mitschele-Thiel, "Multi-numerology based resource allocation for reducing average scheduling latencies for 5G NR wireless networks", in *Proceedings of EuCNC*, Valencia, Spain, 2019. DOI: 10.1109/EuCNC.2019.8802009.

[69] T. Ma, Y. Zhang, F. Wang, D. Wang, and D. Guo, "Slicing resource allocation for eMBB and URLLC in 5G RAN", *Hindawi Wireless Communications and Mobile Computing*, vol. 2020, 2020. DOI: 10.1155/2020/6290375.

[70] L. Feng, Y. Zi, W. Li, F. Zhou, P. Yu, and M. Kadoch, "Dynamic resource allocation with RAN slicing and scheduling for uRLLC and eMBB hybrid services", *IEEE Access*, vol. 8, 2020. DOI: 10.1109/ACCESS.2020.2974812.

[71] J. Li, W. Shi, P. Yang, Q. Ye, X. S. Shen, X. Li, and J. Rao, "A hierarchical soft RAN slicing framework for differentiated service provisioning", *IEEE Wireless Communications*, vol. 27, no. 6, pp. 90–97, 2020. DOI: 10.1109/MWC.001.2000010.

[72] A. Kandoi, "Assessment of key 5G RAN features for integrated services provisioning in a smart city environment", Delft University of Technology, MSc Thesis, 2022.

[73] F. Gunnarsson, M. N. Johansson, A. Furuskar, M. Lundevall, A. Simonsson, C. Tidestav, and M. Blomgren, "Downtilted base station antennas - A simulation model proposal and impact on HSPA and LTE performance", in *Proceedings of VTC-Fall*, Calgary, AB, Canada, 2008. DOI: 10.1109/VETECF.2008.49.

[74] "5G", NGMN, White Paper, version 1.0, 2015.

[75] J. Morais, S. Braam, R. Litjens, S. Kizhakkekundil, and J. L. van den Berg, "Performance modelling and assessment for social VR conference services in 5G radio networks", in *Proceedings of WiMob*, Bologna, Italy, 2021. DOI: 10.1109/WiMob52687.2021.9606263.

[76] "Perspectives on vertical industries and implications for 5G", NGMN, White Paper, 2016.

[77] P. Sarigiannidis, M. Louta, and A. Michalas, "On effectively determining the downlink-to-uplink sub-frame width ratio for mobile WiMax networks using spline extrapolation", in *Proceedings of 15th Panhellenic Conference on Informatics*, Kastoria, Greece, 2011. DOI: 10.1109/PCI.2011.44.

[78] "Traffic models for XR", Qualcomm Incorporated, R1-2101493 3GPP TSG-RAN WG1 e-meeting, 2021.

[79] S.-C. Tseng, Z.-W. Liu, Y.-C. Chou, and C.-W. Huang, "Radio resource scheduling for 5G NR via deep deterministic policy gradient", in *Proceedings of ICC Workshops*, Shanghai, China, 2019. DOI: 10.1109/ICCW.2019.8757174.

[80] "User Equipment (UE) radio transmission and reception", 3GPP, TS 38.101 v15.3.0, 2019.

[81] A. Kandoi and M. Raftopoulou, *System-level simulator for RAN feature evaluation*, version 1.0.0, 2023. DOI: 10.5281/zenodo.10340755.

[82] M. Raftopoulou, J. M. B. da Silva Jr., R. Litjens, H. V. Poor, and P. Van Mieghem, "Agent selection framework for federated learning in resource-constrained wireless networks", submitted, 2024.

[83] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data", in *Proceedings on AISTATS*, Fort Lauderdale, FL, USA, 2017. [Online]. Available: https://proceedings.mlr.press/v54/mcmahan17a/mcmahan17a.pdf.

[84] S. Niknam, H. S. Dhillon, and J. H. Reed, "Federated learning for wireless communications: Motivation, opportunities, and challenges", *IEEE Commununications Magazine*, vol. 58, no. 6, pp. 46–51, 2020. DOI: 10.1109/MCOM.001.1900461.

[85] Z. Du, C. Wu, T. Yoshinaga, K.-L. A. Yau, Y. Ji, and J. Li, "Federated learning for vehicular internet of things: Recent advances and open numbers", *IEEE Open Journal of the Computer Society*, vol. 1, pp. 45–61, 2020. DOI: 10.1109/OJCS.2020.2992630.

[86] A. Nilsson, S. Smith, G. Ulm, E. Gustavsson, and M. Jirstrand, "A performance evaluation of federated learning algorithms", in *Proceedings of 2nd Workshop on DIDL*, Rennes, France, 2018, pp. 1–8. DOI: 10.1145/3286490.3286559.

[87] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-IID data", arXiv preprint arXiv:1806.00582, 2018.

[88] Z. Charles, Z. Garrett, Z. Huo, S. Shmulyian, and V. Smith, "On large-cohort training for federated learning", in *Proceedings of NeurIPS*, 2021. [Online]. Available: https://openreview.net/pdf?id=Kb26p7chwhf.

[89] Y. J. Cho, J. Wang, and G. Joshi, "Towards understanding biased client selection in federated learning", in *Proceedings of AISTATS*, Valencia, Spain, 2022. [Online]. Available: https://proceedings.mlr.press/v151/jee-cho22a.html.

[90] F. Lai, X. Zhu, H. V. Madhyastha, and M. Chowdhury, "Oort: Efficient federated learning via guided participant selection", in *Proceedings of USENIX Symposium OSDI*, 2021, pp. 19–35. [Online]. Available: https://www.usenix.org/conference/osdi21/presentation/lai.

[91] H. T. Nguyen, V. Sehwag, S. Hosseinalipour, C. G. Brinton, M. Chiang, and H. V. Poor, "Fast-convergent federated learning", *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 1, pp. 201–218, 2021. DOI: 10.1109/JSAC.2020.3036952.

[92] W. Chen, S. Horváth, and P. Richtárik, "Optimal client sampling for federated learning", *Transactions on Machine Learning Research*, 2022. [Online]. Available: https://openreview.net/forum?id=8GvRCWKHIL.

[93] M. Ribero and H. Vikalo, "Communication-efficient federated learning via optimal client sampling", arXiv preprint arXiv:2007.15197, 2020.

[94] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge", in *Proceedings of IEEE ICC*, Shanghai, China, 2019. DOI: 10.1109/ICC.2019.8761315.

[95] H. H. Yang, Z. Liu, T. Q. S. Quek, and H. V. Poor, "Scheduling policies for federated learning in wireless networks", *IEEE Transactions on Communications*, vol. 68, no. 1, pp. 317–333, 2020. DOI: 10.1109/TCOMM.2019.2944169.

[96] M. M. Amiri, D. Gündüz, S. R. Kulkarni, and H. V. Poor, "Convergence of update aware device scheduling for federated learning at the wireless edge", *IEEE Transactions on Wireless Communications*, vol. 20, no. 6, pp. 3643–3658, 2021. DOI: 10.1109/TWC.2021.3052681.

[97] W. Shi, S. Zhou, Z. Niu, M. Jiang, and L. Geng, "Joint device scheduling and resource allocation for latency constrained wireless federated learning", *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 453–467, 2021. DOI: 10.1109/TWC.2020.3025446.

[98] W. Zhang, X. Wang, P. Zhou, W. Wu, and X. Zhang, "Client selection for federated learning with non-IID data in mobile edge computing", *IEEE Access*, vol. 9, pp. 24 462–24 474, 2021. DOI: 10.1109/ACCESS.2021.3056919.

[99] L. Huang, Y. Yin, Z. Fu, S. Zhang, H. Deng, and D. Liu, "LoAdaBoost: Loss-based AdaBoost federated machine learning with reduced computational complexity on IID and non-IID intensive care data", *PLOS ONE*, vol. 15, no. 4, 2020. DOI: 10.1371/journal.pone.0230706.

[100] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks", in *Proceedings of MLSys*, Austin, TX, USA, 2020. [Online]. Available: https://proceedings.mlsys.org/paper_files/paper/2020/file/1f5fe83998a09396ebe6477d9475ba0c-Paper.pdf.

[101] S. P. Karimireddy, S. Kale, M. Mohri, S. J. Reddi, S. U. Stich, and A. T. Suresh, "SCAFFOLD: Stochastic controlled averaging for federated learning", in *Proceedings of ICML*, 2020. [Online]. Available: https://proceedings.mlr.press/v119/karimireddy20a.html.

[102] D. A. E. Acar, Y. Zhao, R. Matas, M. Mattina, P. Whatmough, and V. Saligrama, "Federated learning based on dynamic regularization", in *Proceedings of ICLR*, Vienna, Austria, 2021. [Online]. Available: https://openreview.net/pdf?id=B7v4QMR6Z9w.

[103] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of FedAvg on Non-IID data", in *Proceedings of ICLR*, Addis Ababa, Ethiopia, 2020. [Online]. Available: https://openreview.net/forum?id=HJxNAnVtDS.

[104] J. Wang, R. Das, G. Joshi, S. Kale, Z. Xu, and T. Zhang, "On the unreasonable effectiveness of federated averaging with heterogeneous data", arXiv preprint arXiv:2206.04723, 2022.

[105] K. Janocha and W. M. Czarnecki, "On loss functions for deep neural networks in classification", arXiv preprint arXiv:1702.05659, 2017.

[106] K. P. Murphy, *Probabilistic Machine Learning: An introduction*. MIT Press, 2022.

[107] Q. Zeng, Y. Du, K. Huang, and K. K. Leung, "Energy-efficient resource management for federated edge learning with CPU-GPU heterogeneous computing", *IEEE Transactions on Wireless Communications*, vol. 20, no. 12, pp. 7947–7962, 2021. DOI: 10.1109/TWC.2021.3088910.

[108] B. Korte and J. Vygen, *Combinatorial Optimization, Theory and Algorithms*, 5th ed. Springer, 2012. DOI: 10.1007/978-3-642-24488-9.

[109] C. G. Serna and Y. Ruichek, "Classification of traffic signs: The european dataset", *IEEE Access*, vol. 6, 2018. DOI: 10.1109/ACCESS.2018.2884826.

[110] S. Chilamkurthy. "Keras tutorial-traffic sign recognition". (2017), [Online]. Available: https://chsasank.com/keras-tutorial.html.

[111] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", arXiv preprint arXiv:1409.1556, 2015.

[112] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, “A joint learning and communications framework for federated learning over wireless networks”, *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 269–283, 2021. DOI: 10.1109/TWC.2020.3024629.

[113] “User Equipment (UE) radio transmission and reception (TDD)”, 3GPP, TS 25.102 v17.0.0, 2022.

[114] M. Raftopoulou and J. M. B. da Silva Jr., *FLoverWireless: System-level simulator for FL over wireless networks*, version 1.0, 2023. DOI: 10.5281/zenodo.10342352.

[115] M. Raftopoulou, J. M. B. da Silva Jr., R. Litjens, H. V. Poor, and P. Van Mieghem, “Joint vehicle selection and resource allocation using beamforming for federated learning in vehicular networks”, unpublished, 2024.

[116] S. V. Balkus, H. Wang, B. D. Cornet, C. Mahabal, H. Ngo, and H. Fang, “A survey of collaborative machine learning using 5G vehicular communications”, *IEEE Communications Surveys & Tutorials*, vol. 24, no. 2, pp. 1280–1303, 2022. DOI: 10.1109/COMST.2022.3149714.

[117] H. Zhang, J. Bosch, and H. H. Olsson, “End-to-end federated learning for autonomous driving vehicles”, in *Proceedings of IJCNN*, Shenzhen, China, 2021. DOI: 10.1109/IJCNN52387.2021.9533808.

[118] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, “Federated learning in mobile edge networks: A comprehensive survey”, *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020. DOI: 10.1109/COMST.2020.2986024.

[119] H. Zhang, J. Bosch, and H. Olsson, “Real-time end-to-end federated learning: An automotive case study”, in *Proceedings of COMPSAC*, Madrid, Spain, 2021. DOI: 10.1109/COMPSAC51774.2021.00070.

[120] Q. Zeng, Y. Du, K. Huang, and K. K. Leung, “Energy-efficient radio resource allocation for federated edge learning”, in *Proceedings of ICC Workshops*, Dublin, Ireland, 2020. DOI: 10.1109/ICCWorkshops49005.2020.9145118.

[121] K. Fan, W. Chen, J. Li, X. Deng, X. Han, and M. Ding, “Mobility-aware joint user scheduling and resource allocation for low latency federated learning”, in *Proceedings of ICCC*, Dalian, China, 2023. DOI: 10.1109/ICCC57788.2023.10233347.

[122] D. Deveaux, T. Higuchi, S. Uçar, C.-H. Wang, J. Härri, and O. Altintas, “On the orchestration of federated learning through vehicular knowledge networking”, in *Proceedings of VNC*, New York, NY, USA, 2020. DOI: 10.1109/VNC51378.2020.9318386.

[123] K. Venugopal, M. C. Valenti, and R. W. Heath, "Device-to-device millimeter wave communications: Interference, coverage, rate and finite topologies", *IEEE Transactions on Wireless Communications*, vol. 15, no. 9, 2016. DOI: 10.1109/TWC.2016.2580510.

[124] J. Forrest *et al.*, *Coin-or/cbc: Release releases/2.10.10*, version releases/2.10.10, Apr. 2023. DOI: 10.5281/zenodo.7843975.

[125] "Study in evaluation methodology for new Vehicle-to-Everything (V2X) use cases for LTE and NR", 3GPP, TR 37.885 v15.3.0, 2019.

[126] M. Raftopoulou, R. Litjens, and P. Van Mieghem, "Fréchet distribution in geometric graphs for drone networks", *Physical Review E*, vol. 106, 2 2022. DOI: 10.1103/PhysRevE.106.024301.

[127] B. Bollobas, *Random graphs*, 2nd ed. Cambridge University Press, 2011.

[128] E. N. Gilbert, "Random plane networks", *Journal of the Society for Industrial and Applied Mathematics*, vol. 9, no. 4, pp. 533–543, 1961.

[129] M. Wilsher, C. P. Dettmann, and A. Ganesh, "Connectivity in one-dimensional soft random geometric graphs", *Physical Review E*, vol. 102, 2020. DOI: 10.1103/PhysRevE.102.062312.

[130] M. Haenggi, J. G. Andrews, F. Baccelli, O. Dousse, and M. Franceschetti, "Stochastic geometry and random graphs for the analysis and design of wireless networks", *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 7, pp. 1029–1046, 2009. DOI: 10.1109/JSAC.2009.090902.

[131] C. Bettstetter, "On the minimum node degree and connectivity of a wireless multihop network", in *Proceedings of ACM MobiHoc*, Lausanne, Switzerland, 2002. DOI: 10.1145/513800.513811.

[132] M. Barthelemy, "Spatial networks", *Physics Reports*, vol. 499, pp. 1–109, 2011. DOI: 10.1016/j.physrep.2010.11.002.

[133] G. A. Pagani and M. Aiello, "From the grid to the smart grid, topologically", *Physica A: Statistical Mechanics and its Applications*, vol. 449, pp. 160–175, 2016. DOI: 10.1016/j.physa.2015.12.080.

[134] V. Erba, S. Ariosto, M. Gherardi, and P. Rotondo, "Random geometric graphs in high dimension", *Physical Review E*, vol. 102, 2020. DOI: 10.1103/PhysRevE.102.012306.

[135] J. Hoebeke, I. Moerman, B. Dhoedt, and P. Demeester, "An overview of mobile and ad hoc networks: Applications and challenges", *Communications Network*, vol. 3, pp. 60–66, 2004.

[136] I. Bekmezci, O. K. Sahingoz, and S. Temel, “Flying ad-hoc networks (FANETs): A survey”, *Ad Hoc Networks*, vol. 11, no. 3, pp. 1254–1270, 2013. DOI: 10.1016/j.adhoc.2012.12.004.

[137] J. Dall and M. Christensen, “Random geometric graphs”, *Physical Review E*, vol. 66, 2002. DOI: 10.1103/PhysRevE.66.016121.

[138] P. Van Mieghem, “Paths in the simple random graph and the waxman graph”, *Probability in the Engineering and Informational Sciences*, vol. 15, pp. 535–555, 2001.

[139] P. Fan, G. Li, K. Cai, and K. B. Letaief, “On the geometrical characteristics of wireless ad-hoc networks and its application in network performance analysis”, *IEEE Transactions on Wireless Communications*, vol. 6, no. 4, pp. 1256–1264, 2007. DOI: 10.1109/TWC.2007.348322.

[140] B. Bakhshi and S. Khorsandi, “Node connectivity analysis in multi-hop wireless networks”, in *Proceedings of the 2010 IEEE Wireless Communications and Networking Conference*, Sydney, NSW, Australia, 2010. DOI: 10.1109/WCNC.2010.5506578.

[141] Z. Khalid, S. Durrani, and J. Guo, “A tractable framework for exact probability of node isolation and minimum node degree distribution in finite multihop networks”, *IEEE Transactions on Vehicular Technology*, vol. 63, no. 6, pp. 2836–2847, 2014. DOI: 10.1109/TVT.2013.2293580.

[142] R. Hekmat and P. Van Mieghem, “Connectivity in wireless ad-hoc networks with a log-normal radio model”, *Mobile Networks and Applications, Special number: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, vol. 11, no. 3, pp. 351–360, 2006.

[143] S. C. Ng, G. Mao, and B. D. O. Anderson, “Critical density for connectivity in 2D and 3D wireless multi-hop networks”, *IEEE Transactions on Wireless Communications*, vol. 12, no. 4, pp. 1512–1523, 2013. DOI: 10.1109/TWC.2013.021213.112130.

[144] Z. Khalid and S. Durrani, “Connectivity of three dimensional wireless sensor networks using geometrical probability”, in *Proceedings of AusCTW*, Adelaide, SA, Australia, 2013. DOI: 10.1109/AusCTW.2013.6510043.

[145] C. P. Dettmann and O. Georgiou, “Random geometric graphs with general connection functions”, *Physical Review E*, vol. 93, no. 3, 2016. DOI: 10.1103/PhysRevE.93.032313.

[146] S. Lellouche and M. Souris, “Distribution of distances between elements in a compact set”, *Stats*, vol. 3, no. 1, 2020. DOI: 10.3390/stats3010001.

[147] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage", *IEEE Communications Letters*, vol. 20, no. 8, pp. 1647–1650, 2016. DOI: 10.1109/LCOMM.2016.2578312.

[148] A. V. Savkin and H. Huang, "Asymptotically optimal deployment of drones for surveillance and monitoring", *Sensors*, vol. 19, no. 9, 2019. DOI: 10.3390/s19092068.

[149] B. Lin, W. Wang, J. Guo, and Z. Fei, "Outage performance for UAV communications under imperfect beam alignment: A stochastic geometry approach", in *Proceedings of ICCT*, Tianjin, China, 2021. DOI: 10.1109/ICCT52962.2021.9657883.

[150] L. Blumenson, "A derivation of n-dimensional spherical coordinates", *The American Mathematical Monthly*, vol. 67, no. 1, pp. 63–66, 1960.

[151] E. C. Titchmarsh, *The Theory of Functions*. Oxford University Press, 1964.

[152] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions*. Dover Publications, Inc., 1968.

[153] P. Van Mieghem, "Binet factorial series and extensions to Laplace transforms", Delft University of Technology, report20210202, 2021. [Online]. Available: http://arxiv.org/abs/2102.04891.

Acknowledgements

First of all, I would like to thank my promotor Prof. Piet Van Mieghem who gave me the opportunity to pursue my doctorate. I am thankful for his guidance and for giving me the freedom to explore different research ideas. I am grateful for his support, feedback and for always inspiring me with his enthusiasm for research and science. During my PhD journey I also had the opportunity to work with Dr. Remco Litjens, who was the person to first introduce me to the field of cellular networks and who also guided me through my MSc thesis. Remco, I am so grateful for all the knowledge you provided me and for helping me become a better researcher. Thank you for always being flexible, available for help and for teaching me the importance of being consistent and critical. Of course I cannot forget to thank you for giving me literally thousands of review comments on my reports. Piet and Remco, it was such a pleasure working with you.

I am again thankful to Prof. Van Mieghem for initiating our collaboration with Prof. Vincent Poor and Dr. José Mairton B. da Silva Jr. Thank you for introducing me to the challenging research topic of federated learning and for all the interesting discussions we had. Mairton, I am really grateful for being patient with me and for teaching me so many things about federated learning and optimisation. Apart from federated learning, I also had the chance to have many interesting discussions related to my research on RAN slicing. Dr. Eric Smeitink, Dr. Ramin Hekmat and Dr. Frank Mertz, thank you for our discussions and your valuable insights. During my PhD time I also had the pleasure to guide students through their MSc research. Chinedu Aguwamba and Ayushi Kandoi, I really enjoyed working with you.

I would also like to thank my colleagues at the Network Architectures and Services (NAS) group, who made working at the office enjoyable and inspiring. Prof. Rob Kooij, Dr. Edgar van Boven and Dr. Maksim Kitsak, thank you for all the inspiring discussions and words of encouragement. During the past four years, I have shared an office with Dr. Bastian Prasse, Dr. Long Ma, Dr. Qiang Liu, Dr. Massimo Achterberg, Zhihao Qiu and Yingue Ke. Thank you all for listening to my ideas and problems and for all the coffee breaks and fun discussions we had. I am also thankful to the rest group members Rogier Noldus, Gabriel Budel, Fenghua Wang, Dr. Ivan Jokic, Elizaveta Evmenova, Robin Persoons, Xinhan Liu, Brian Chang, Matteo D'Alessandro, David Almasan, Dr. Sergey Shvydun, Dr. Scott Dahlgren, Dr. Yuxuan Yang, Dr. Peng Sun and Dr. Zhidong He. I have enjoyed all of our discussions and your company during our NAS Friday drinks and other social activities. Robin, thank you for helping me with the Dutch summary of this dissertation. Lastly, I would like to thank our secretaries Laura de Groot and Francis Butlers and our cluster administrator Erik de Vries for always providing their help and support.

Words cannot express how thankful I am to my family. Doing my PhD research during the COVID-19 pandemic would have been impossible without your support and constant phone calls. My gratitude also goes to my friends who have been supporting and encouraging me. Thank you for always being there for me and reminding me to enjoy life during stressful and difficult periods. Last but not least, I am thankful to my boyfriend Nils and his family. Nils, I am extremely grateful for your love, support and the patience you had with me during the busy, difficult and stressful periods. Also, thank you for designing the cover of this dissertation, for your assistance in debugging my code and for making me a better software developer.

Biography

Maria Raftopoulou received the MEng degree in Electrical and Computer Engineering from the National Technical University of Athens, Greece, in 2016 and the MSc degree in Electrical Engineering with *cum laude* from the Delft University of Technology, The Netherlands, in 2018. During her MSc degree, she performed a research internship on the 4G RAN at the Netherlands Organisation for Applied Scientific Research (TNO), where she also completed her MSc thesis, related to the 5G RAN. After graduation, she worked as a Technology Young Talent at KPN, where she worked on projects related to upcoming technologies and applications in the telecommunications sector as well as related to feature designs for deployment in the 4G RAN. After attending her first conference in 2019, she decided to pursue her PhD degree at the Network Architectures and Services (NAS) group at the Delft University of Technology, under the supervision of Dr. Remco Litjens, MSc and Prof.dr.ir. Piet Van Mieghem. As of March 2024, she is working as a Scientist Innovator at the Networks department at TNO.

Her research interests include machine learning and optimization over wireless networks. When not doing research, she explores new places within The Netherlands and abroad or she is crocheting.

List of Publications

6. **M. Raftopoulou**, J. M. B. da Silva Jr., R. Litjens, H. V. Poor and P. Van Mieghem, “Joint vehicle selection and resource allocation for federated learning in vehicular wireless networks", in preparation, 2024.

5. **M. Raftopoulou**, J. M. B. da Silva Jr., R. Litjens, H. V. Poor and P. Van Mieghem, “Agent selection framework for federated learning in resource-constrained wireless networks", under review, 2024.

4. A. Kandoi, **M. Raftopoulou** and R. Litjens, “Assessment of 5G RAN features for integrated services provisioning in smart cities", in *Proceedings of 15th International Workshop on STWiMob*, Thessaloniki, Greece, 2022. DOI:10.1109/WiMob55322.2022.9941612.

3. **M. Raftopoulou**, R. Litjens and P. Van Mieghem, “Frćhet distribution in geometric graphs for drone networks", *Physical Review E*, Vol. 106, No. 2, 2022. DOI:10.1103/PhysRevE.106.024301.

2. **M. Raftopoulou** and R. Litjens, “Optimisation of numerology and packet scheduling in 5G networks: To slice or not to slice?", in *Proceedings of IEEE VTC2021-Spring*, Helsinki, Finland, 2021. DOI:10.1109/VTC2021-Spring51267.2021.9448814.

1. **M. Raftopoulou**, L. Jorguseski and R. Litjens, “Design and assessment of low-latency random access procedures in 5G networks", in *Proceedings of EuCNC*, Valencia, Spain, 2019. DOI:10.1109/EuCNC.2019.8801962.

Software

3. **M. Raftopoulou** and J. M. B. da Silva Jr., FLoverWireless: System-level simulator for FL over wireless networks (Version 1.0), 2023. DOI:10.5281/zenodo.10342352

2. A. Kandoi and **M. Raftopoulou**, System-level simulator for RAN feature evaluation (Version 1.0), 2023. DOI:10.5281/zenodo.10340755

1. **M. Raftopoulou**, SliceSim: System-level simulator for RAN slicing (Version 1.0), 2023. DOI: 10.5281/zenodo.10204995