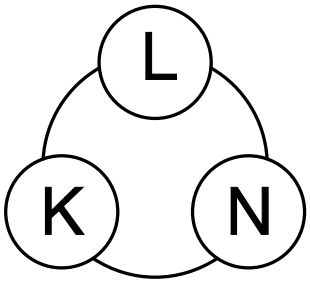


Technical University of Munich
Chair of Communication Networks
Prof. Dr.-Ing. Wolfgang Kellerer

Master Thesis

Joint Network and Edge Cloud
α-fair Resource Allocation to Vehicular Users in a Cell
for URLLC Traffic

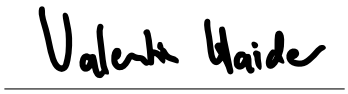
Author:	Haider, Valentin Thomas
Address:	Dr.-Herbert-Quandt-Str. 34
	84130 Dingolfing
	Germany
Matriculation Number:	03735338
Supervisor:	Dr. Mehmeti, Fidan
Begin:	April 4, 2022
End:	December 16, 2022

Master Thesis
at the Chair of Communication Networks (LKN)
of the TUM School of Computation, Information and Technology (CIT)
of the Technical University of Munich (TUM)
Title: Joint Network and Edge Cloud α-fair Resource Allocation to Vehicular Users in a Cell for URLLC Traffic
Author: Valentin Thomas Haider

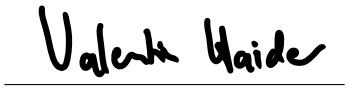
Valentin Thomas Haider
Dr.-Herbert-Quandt-Str. 34
84130 Dingolfing
Germany
valentin.haider@tum.de

With my signature below, I assert that the work in this thesis has been composed by myself independently and no source materials or aids other than those mentioned in the thesis have been used.

Munich, December 16, 2022
Place, Date


Signature

Munich, December 16, 2022
Place, Date


Signature

Acknowledgements

I would like to convey my sincere gratitude to all those who contributed to this master's thesis. Profoundly, I would like to thank Dr. Ph.D. Fidan Mehmeti, Senior Researcher at the Chair of Communication Networks of the TU Munich. In being my academic supervisor he always pointed me in the right direction. His professional competence and consultation made a considerable contribution to this work and guided me throughout the whole thesis. Without his enthusiasm and dedication this work would not have been possible.

Furthermore, I am greatly thankful to my supervisor at the BMW Group, Ana Cantarero. By providing me with her constructive feedback and connecting me with the right people, she always supplied me with the things I needed. I would also like to thank the team leader of DE-711, Fabian Schulte, for giving me the opportunity to write this thesis. In addition, I am deeply grateful to all the people who accompanied me during my master's studies with the BMW Group, namely my mentor Alexander Landes as well as my team leaders Senay Pillny and Sandra Carstens.

For granting me a scholarship during my master's studies, I would like to thank the Hanns-Seidel-Foundation. Their conceptual and financial support from the funds of the Federal Ministry of Education and Research made life more pleasant for me. The seminars I could attend offered me interesting insights and the possibility to help shape today's political and social society.

Finally, I would like to thank my loved ones for their inexhaustible support. My achievements would not have been possible without their encouragement and advice.

Abstract

5G networks have emerged as the only viable solution to render a satisfying level of performance to various types of services relying on mobile communications, where each of them has very challenging traffic requirements. One of those services are ultra reliable low-latency communications (URLLC), which are characterized by the stringent demand to deliver packets within a very short time with a high reliability. A use case where these services are especially sensitive are vehicular networks. Besides being successfully transmitted/received, the data need to be processed as well. To satisfy these strict requirements, both the required data rate and the processing rate need to be determined, given the channel conditions and traffic characteristics of the service. With constraints on both the radio access network (RAN) and edge computing resources as well as with the competition between an ever increasing number of users in cellular networks, a very important question which arises is that of *admission control*. This guarantees users will not suffer from deteriorating performance. Furthermore, after ensuring the availability of enough resources to satisfy the traffic requirements of the vehicular users, adequate *resource allocation schemes* need to be devised in order to maximize the number of users that can be served by the network. However, the time-varying nature of the channel conditions in wireless networks renders this process challenging. In this thesis, first, using analytical modeling, admission control policies for both homogenous and heterogenous sets of users are derived for a scenario consisting of uplink communication and edge processing. The theoretical outcomes are validated using simulations based on a 5G dataset. Results show that the number of admitted users depends on the worst-case channel conditions, the deadline by which the data must be processed, and the available resources. Next, the problem of jointly allocating RAN and computing resources such that all the traffic requirements of individual users are met and the utility is maximized for different types of fairness is considered for the same scenario. To this end, an optimization problem for the general case of α-fairness is formulated and its characteristics are explored. The special cases *no fairness* ($\alpha = 0$), *proportional fairness* ($\alpha = 1$), *delay minimization* ($\alpha = 2$), and *max-min fairness* ($\alpha \to \infty$) are then considered in more detail. For each of these problems, polynomial-time allocation heuristics are proposed. Using data from real traces, it is shown that the performance achieved with these approaches is not more than 7.11 % away from the optimum. Subsequently, the performed analysis and the proposed algorithms are extended to a second scenario, in which a downlink communication link is included as well. Evaluation results show that the performance of the heuristics for this scenario is not more than 11.75 % away from the optimum, while the average performance is significantly better (0.58 % across all scenarios). Finally, the results achieved with this mobile edge computing (MEC) setup are compared to an already existing cloud computing setup from an automotive original equipment manufacturer (OEM).

Kurzfassung

5G-Netze haben sich als die einzige praktikable Lösung erwiesen, um ein zufriedenstellendes Performanceniveau für verschiedene Arten von Diensten zu erreichen, die auf mobiler Datenkommunikation basieren und jeweils sehr anspruchsvolle Anforderungen an den Datenverkehr haben. Eine Art dieser Dienste sind Ultra reliable low-latency communications (URLLC, dt.: extrem zuverlässige Kommunikation mit geringer Latenzzeit), die durch die strikte Anforderung charakterisiert sind, Datenpakete innerhalb einer sehr kurzen Zeit mit hoher Zuverlässigkeit zu übertragen. Ein Anwendungsfall, bei dem diese Dienste besonders empfindlich sind, sind Fahrzeugnetze. Die Daten müssen nicht nur erfolgreich übertragen/empfangen, sondern auch verarbeitet werden. Um diese strengen Anforderungen zu erfüllen, müssen sowohl die erforderliche Datenrate als auch die Verarbeitungsrate unter Berücksichtigung der Kanalbedingungen und der Datenverkehrseigenschaften des Dienstes bestimmt werden. Da sowohl die Radio access network (RAN, dt: Funkzugangsnetz)- als auch die Edge-Computing-Ressourcen begrenzt sind und immer mehr Nutzer in Mobilfunknetzen miteinander konkurrieren, stellt sich die wichtige Frage nach einer geeigneten *Zulassungskontrolle*. Dadurch wird gewährleistet, dass die Nutzer nicht unter einer Verschlechterung der Performance des Netzwerks leiden. Nachdem sichergestellt ist, dass genügend Ressourcen zur Verfügung stehen um den Datenverkehrsanforderungen der mobilen Nutzer gerecht zu werden, müssen geeignete *Ressourcenallokationsschemata* entwickelt werden, um die Anzahl der Nutzer, die vom Netz bedient werden können, zu maximieren. Die zeitlich variierenden Kanalbedingungen in drahtlosen Netzwerken machen diesen Prozess jedoch zu einer Herausforderung. In dieser Arbeit werden zunächst mithilfe analytischer Modellierung Zulassungskontrollstrategien für homogene und heterogene Gruppen von Nutzern für ein Szenario entwickelt, das aus Uplink-Kommunikation und Edge Processing besteht. Die theoretischen Ergebnisse werden anhand von Simulationen auf der Grundlage eines 5G-Datensatzes validiert. Die Resultate zeigen, dass die Anzahl der zugelassenen Nutzer von den schlechtesten Kanalbedingungen, der Latenzfrist, bis zu der die Daten verarbeitet werden müssen, und den verfügbaren Ressourcen abhängt. Im nächsten Schritt wird für dasselbe Szenario das Problem der gemeinsamen Zuweisung von RAN- und Computing-Ressourcen betrachtet, so dass alle Datenverkehrsanforderungen der einzelnen Nutzer erfüllt werden und der Nutzwert für verschiedene Arten von Fairness maximiert wird. Zu diesem Zweck wird ein Optimierungsproblem für den allgemeinen Fall der α-Fairness formuliert und dessen Eigenschaften untersucht. Anschließend werden die Spezialfälle *keine Fairness* ($\alpha = 0$), *proportionale Fairness* ($\alpha = 1$), *Latenzminimierung* ($\alpha = 2$) und *Max-Min-Fairness* ($\alpha \to \infty$) näher betrachtet. Für jedes dieser Probleme werden Zuweisungsheuristiken vorgeschlagen, deren Zeitkomplexität polynomial ist. Anhand von Daten aus realen Messungen wird gezeigt, dass die mit diesen Ansätzen erzielte Performance nicht mehr als $7,11\,\%$ vom Optimum entfernt ist. Anschließend werden die durchgeführte Analyse und die vorgeschlagenen Algorithmen auf ein zweites Szenario ausgedehnt, bei dem auch eine Downlink-Kommunikationsverbindung einbezogen wird. Die Evaluationsergebnisse zeigen, dass die Performance der Heuristiken für dieses Szenario

nicht mehr als $11,75\,\%$ vom Optimum entfernt ist, wobei die durchschnittliche Leistung viel besser ist ($0.58\,\%$ über alle Szenarien). Abschließend werden die mit diesem Mobile Edge Computing (MEC) Setup erzielten Ergebnisse mit einem bereits bestehenden zentralen Cloud Computing Setup eines Automobilherstellers (OEM) verglichen.

Contents

Acronyms

BS	Base Station
CCC	Centralized Cloud Computing
CDF	Cumulative Distribution Function
CQI	Channel Quality Indicator
eMBB	Enhanced Mobile Broadband
EMEA	Europe, the Middle East, and Africa
FDD	Frequency-Division Duplex
FR1	Frequency Range 1
gNodeB	Next Generation Node B
GSM	Global System for Mobile Communication
half-duplex FDD	Half-Duplex Frequency-Division Duplex
HVD	Hybdrid Voice Dialog
ITU	International Telecommunication Union
LTE	Long Term Evolution
MCS	Modulation and Coding Scheme
MEC	Mobile Edge Computing
mMTC	Massive Machine-Type Communications
NUM	Network Utility Maximization
OEM	Original Equipment Manufacturer
PMF	Probability Mass Function
PRB	Physical Resource Block
QoS	Quality of Service
RAN	Radio Access Network
SCS	Subcarrier Spacing
SINR	Signal-to-Interference-Plus-Noise-Ratio
SLA	Service Level Agreement
ST	Secure Time
TDD	Time-Division Duplex
URLLC	Ultra Reliable Low-Latency Communications

Symbols

α	α-fairness value $\in [0, \infty)$
β	slack variable that is equal to $1 - \alpha$
$b_{u,i}$	number of allocated uplink PRBs of user i
$\Delta_{\{u,d\},i}$	uplink/downlink data size of user i
ϵ	outage probability
$f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i)$	utility function of user i in the uplink-only scenario, see (5.2)
$f_i^\alpha(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i)$	utility function of user i in the uplink/downlink scenario, see (6.2)
g	slack variable introduced as objective function for the optimization problem in epigraph form
$\gamma_{\{u,d\},i}$	uplink/downlink RAN data rate of user i, i.e., $\gamma_{\{u,d\},i} = \sum_{j=1}^{K_{\{u,d\}}} I_{\{u,d\},ij}\Phi_{\{u,d\},ij}$
$\boldsymbol{I}_{\{u,d\}}$	uplink/downlink PRB allocation matrix, rows: users, columns: PRBs
$\boldsymbol{I}_{\{u,d\},\{i,j\}}$	uplink/downlink PRB allocation vector of user i or of PRB j
$I_{\{u,d\},ij}$	uplink/downlink PRB allocation indicator of user i for PRB j
$\boldsymbol{J}_{\{u,d\}}$	uplink/downlink int. PRB allocation matrix, rows: users, columns: PRBs
$\boldsymbol{J}_{\{u,d\},\{i,j\}}$	uplink/downlink int. PRB allocation vector of user i or of PRB j
$J_{\{u,d\},ij}$	uplink/downlink int. PRB allocation indicator of user i for PRB j
$\mathcal{K}_{\{u,d\}}$	the set of all uplink or downlink PRBs
$\mathcal{E}$	exponential cone, see (5.18)
$K_{\{u,d\}}$	number of available uplink/downlink PRBs
L	number of available edge computing resources
$\boldsymbol{m}$	edge computing resource allocation vector
m_i	edge computing resource indicator of user i
N	number of users in the system
$\boldsymbol{n}$	int. edge computing resource allocation vector
n_i	int. edge computing resource indicator of user i
p	processing rate of one edge computing unit
$\boldsymbol{\Phi}_{\{u,d\}}$	uplink/downlink data rate matrix, rows: users, columns: PRBs
$\Phi_{\{u,d\},ij}$	uplink/downlink data rate of user i for PRB j
$\mathcal{P}_\zeta^n$	n-dimensional power cone, see (5.17)

$\mathcal{Q}^n$	n-dimensional quadratic cone, see (5.16)
R_i	user i's per-block rate modeled as discrete random variable with $R_i \in \{r_1, r_2, ..., r_{15}\}$
$\rho_{R_i}(r_k)$	probability of user i experiencing the data rate corresponding to CQI k
r_k	data rate corresponding to CQI k
$r_{min,i}$	minimum data rate user i has experienced, i.e. lowest data rate r_k in table 4.1 where $\rho_{R_i}(r_k) > 0$
$\mathcal{Q}_r^n$	n-dimensional rotated quadratic cone, see (6.16)
$\boldsymbol{s}$	slack variable vector with slack variables s_{ki}
s_{1i}	slack variable for the uplink data rate of user i, see (5.19g)
s_{2i}	slack variable for the processing rate of user i, see (5.19h)
s_{3i}	slack variable for the downlink data rate of user i, see (6.17i)
t_i	delay that user i is experiencing in admission control
$t_i(\boldsymbol{I}_{u,i}, m_i)$	delay of user i in the uplink-only scenario, see (5.11)
$t_i(\boldsymbol{I}_{u,i}, \boldsymbol{I}_{d,i}, m_i)$	delay of user i in the uplink/downlink scenario, see (6.11)
T_{max}	maximum allowed delay a packet can experience
$\mathcal{U}$	the set of all users
$\boldsymbol{u}$	slack variable vector with slack variables u_{ki}
u_{ki}	slack variable bounding a term including the corresponding s_{ki}

1 Introduction

In the 1980s, the first generation of mobile communication was developed and deployed. Several incompatible analog technologies existed in parallel, which all provided voice calls with poor reliability and almost no security. About 10 years later, specifications comprising the second generation of mobile communication were designed. While there were multiple technologies at the beginning, Global System for Mobile Communication (GSM) evolved as the main used technology over the years. Since 2G systems relied on digital traffic channels, the transmission of digital data was enabled in addition to offering voice calls. In the early 2000s, one of the biggest steps in the evolution of mobile communication happened. 3G was the first technology for which a standard from the International Telecommunication Union (ITU) was developed. With the deployment of 3G, much higher data rates could be achieved, such that services like emailing, picture exchange, or web browsing could be offered. Additionally, the provided reliability and security of 3G technologies was better than compared to the previous generation. In late 2009, the commercial operation of the next generation of mobile communication, i.e., 4G, started. 4G is constituted by the Long Term Evolution (LTE) technology, whose development started in late 2004. It was the first technology that purely relied on IP packets. Compared to earlier generations, the data rates and latencies provided by 4G systems improved again, which enabled applications like mobile video conferencing, gaming, and IP telephony. Finally, since the second half of the 2010s, the fifth generation of mobile communication is available, but it is still under development. Three main service types are provided by 5G networks: enhanced mobile broadband (eMBB), massive machine-type communications (mMTC), and ultra reliable low-latency communications (URLLC). The services that are of mMTC type require support to serve a large number of devices and low energy consumption. Very high data rates with a high spectral efficiency are needed for eMBB. Lastly, URLLC services require extremely low latencies and also support for high mobility. The outlined evolution of mobile communication generations is summarized in Figure 1.1. [DPS20, Sta21]

1.1 Motivation

URLLC corresponds to applications like autonomous driving, remote surgery, and remote monitoring and control [BDP18]. Their main requirements are to deliver packets with a very high reliability within a short time (on the order of ms), which is quite challenging. Furthermore, besides being transmitted, those data need to be processed as well. This

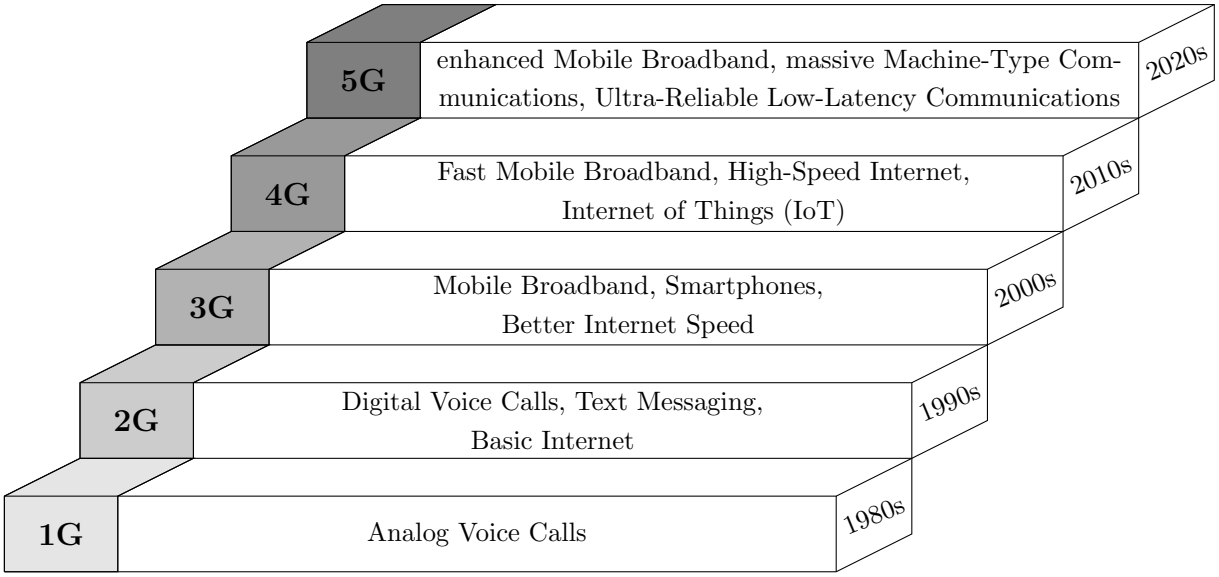


Figure 1.1: Evolution of the mobile communication generations, cf. [Sta21].

emphasizes the difficulty of handling the data even further, as two types of limited resources in the cell must be allocated, and the number of users competing for them is constantly increasing. The two types of resources are the radio access network (RAN) resources that enable the transmission/reception of information and edge computing resources for processing the received data. Since the aforementioned URLLC services are not only sensitive to abiding by those stringent requirements, but, given their nature, any failure to comply may bring a serious risk to human lives, enabling their flawless operation is of paramount importance.

Facilitating this impeccable functionality is particularly strenuous in cellular networks, where the channel characteristics of users exhibit dynamic behavior over time due to mobility and processes like shadowing [Gol05]. Therefore, to provide a given data rate and processing rate that will satisfy the delay requirements of all users, first, a proper admission policy is needed to ensure the availability of enough resources for all users admitted to the network. Secondly, suitable resource allocation schemes on two levels need to be developed: on the RAN side for transmission, and on the analyst side (e.g., edge cloud) for computing. In order to enable the service for as many users as possible, the resources need to be allocated in an *efficient way*. Furthermore, since the two types of resources are interacting with each other, i.e., they both influence the experienced delay, they need to be considered *jointly* during the allocation process. Using realistic assumptions, this thesis displays an approach on designing admission policies suited for different types of vehicular users as well as an approach on allocating RAN and computing resources in a fair and efficient way. The results that are presented are particularly interesting for network operators, as they show how to increase the overall utility of their network. Moreover, the results demonstrate new ways of enabling digital services for car manufacturers.

1.2 Thesis Overview

In order to introduce admission control policies as well as resource allocation schemes for vehicular users requesting URLLC services, this thesis is divided into 8 chapters. The following part of chapter 1 shows the concrete structural layout and provides a summary of the content.

Chapter 2: The second chapter presents the results of a literature research on admission control policies and resource allocation schemes in 5G NR for URLLC. While there exists previous work on both topics, no work pursuing the approaches presented in this thesis or considering resources on both the communication and processing side under the given assumptions was found.

Chapter 3: The third chapter is dedicated to a detailed introduction of the discussed system model and the working principles of 5G NR. Additionally, assumptions made are presented and related to known technical standards.

Chapter 4: In order to guarantee the availability of a sufficient amount of resources to fulfill all traffic requirements of users requesting URLLC services, admission control policies for both homogenous and heterogenous sets of users are designed. By relaxing some assumptions introduced previously, the analytic tractability of the derivations for the admission policies is ensured. Before completing this chapter with a short summary of the main aspects of the policies, the theoretical results are validated using simulations and interesting insights in the performance of the admission control mechanisms are provided.

Chapter 5: In this chapter, a scenario in which vehicular users are sending service requests to a base station (BS), where these requests are processed, is considered. To enable these service inquiries, a joint allocation scheme providing α-fairness is developed. By relaxing an integer-constraint on decision variables, the continuous allocation optimization problem is shown to be convex and solvable in polynomial time. Next, using analyses regarding the assignment of resources in the continuous case, approximation algorithms solving the original integer problem are introduced, which are evaluated in the penultimate section of this chapter. Concluding, the main steps of the allocation scheme development as well as the key performance indicators are summarized.

Chapter 6: Following the same approach as in Chapter 5, the problem of jointly allocating RAN and edge computing resources in a two-way communication scenario is tackled in this chapter. After inserting the downlink communication parts into the optimization problem introduced beforehand, the convexity and polynomial-time solvability of the continuous optimization problem is again proven. Next, the adapted approximation algorithms are outlined and a performance evaluation using simulations is once more conducted. Lastly, the key aspects and results of this chapter are recapitulated in the last section.

Chapter 7: In the penultimate chapter of this work, real datasets from a centralized cloud computing (CCC) server realized by an automotive original equipment manufac-

turer (OEM) are analyzed. These datasets include information regarding the transmission latency of packets from a vehicular user to the server as well as regarding the processing rates of the cloud server. Using averaged results from the analysis, a comparison between the developed mobile edge computing (MEC) system and the automotive OEM's CCC system is conducted.

Chapter 8: Finally, the main results of this thesis are summarized and prospects on possible future works are illustrated.

2 Related Work

In order to be able to classify this thesis thematically, related work on the two main problems addressed in this thesis is discussed in the following. Thereby, first, the background regarding the admission control problem is presented. In the second section, other scientific work related to the problem of optimal RAN and edge computing resource allocation with a fairness guarantee is outlined.

2.1 Admission Control

During the literature research on admission control policies, it became clear that most work regarding this topic relates to admission control on the granularity level of network slices, as can be observed from, e.g., [OF20] and [ON19]. Specifically, an admission and congestion control policy for network slices in 5G, without specifying the type of service, is for example considered in [HDD+18]. In [HNLF19] and [GMRLa20], the joint admission of users with eMBB and URLLC traffic is considered. However, the setup in both of them is different to the setup in this work, since their goal is to maximize the number of eMBB users that can be admitted, while achieving a maximum blocking probability for all URLLC users or serving all URLLC users, respectively. The main difference to the present work stems from the already mentioned fact that in this thesis admission control is performed on a granularity level of physical resource blocks (PRBs), while [HNLF19] and [GMRLa20] consider network slices. Furthermore, as opposed to [HNLF19] and [GMRLa20], in this thesis it is not assumed to have enough resources to serve all URLLC users. Finally, the processing of the transmitted data is not considered [HNLF19] and [GMRLa20], while it is taken into account in the existing work, which makes the present scenario more involved as admission needs to be determined depending on two different network parts.

Admission control for users requiring a consistent data rate is considered in [ML19]. Two scenarios, in which users are experiencing the same or different channel conditions are considered. While the proposed policies show a good performance, the slot duration is 100 ms and hence not realistic. Interesting insights regarding the dependence of the number of admitted users on the experienced channel variability of the users are provided, however, neither a delay constraint nor the processing of data is taken into account, as a downlink communication scenario, probably for eMBB, is considered. When it comes to mMTC traffic, which is characterized by the least stringent requirements, corresponding admission

policies were proposed in [ML21a] for the same scenarios as in [ML19]. Although a latency requirement is taken into consideration in [ML21a], the processing of the transmitted data is again not treated.

The work most similar in spirit to the present thesis is [ML21b], in which the admission control is performed for URLLC traffic. Similar to the work presented in this thesis, in [ML21b] the maximum number of users that can be admitted is determined for a homogenous set of users, whereas for a heterogenous set of users the policy for admitting a newly arriving user is provided given its channel and traffic characteristics. However, only the transmission component of the latency is considered in [ML21b], and the processing of the data is completely omitted, which constitutes the novelty of the present work.

2.2 Joint Network and Edge Cloud Resource Allocation

In [CLLW19], the authors consider a two-level network architecture comprising a lower-level RAN with edge computing resources as well as an upper-level transport network with central cloud computing resources. They investigate a network slicing process for the three types of services in 5G and especially examine the partitioning ratios between the lower- and upper-level resources for the service types. While they constrain their optimization problem with a maximum delay requirement for the services, their objective is to minimize an over-provisioning ratio defined as the ratio of the required delay divided by the actually achieved delay. Moreover, since they consider slices as the unit of allocation, the granularity of the units is much larger than in the present work. A paper that is concerned with uplink communication of URLLC traffic is [CVS20]. However, the authors do not formulate an optimization problem and also do not consider the processing of the data; instead, two protocols for connection-less transmission of URLLC traffic are assessed.

Further, the work in [SIL+16] considers the optimal allocation of transmission attempts and communication channels for URLLC traffic in a cellular system. Two optimization problems for the resource allocation are formulated: in the first scenario, the number of transmission attempt assignments is fixed before starting the transmission, whereas it is adaptive in the second scenario. While [SIL+16] is also concerned with reducing the required resources, the setup and the objective are different from the present work, and providing fairness is not one of the aims. To meet the latency and reliability requirements of URLLC traffic, the authors in [HEGS+18] propose a periodic resource allocation scheme. While minimizing the needed network resources, i.e., choosing the best modulation and coding scheme (MCS) when considering retransmissions and the latency and reliability constraints, the scope of [HEGS+18] is limited due to the assumption of a factory environment, which implies that channel conditions are not changing over time. Furthermore, the objective does again not include providing any sort of fairness.

In [ML22], three objectives similar to the present thesis are considered: maximize the total throughput in the network, provide proportional fairness, and achieve max-min fairness. However, there are some important differences between this work and [ML22]. The primary goal in [ML22] is to provide a given constant data rate to everyone and then reallocate the unused resources to the users according to the respective fairness policies. Besides, while the setup in this thesis is related to URLLC traffic, the target of [ML22] are users with eMBB traffic. Satisfying the requirements of users with URLLC traffic is more challenging. Lastly, the authors of [ML22] only consider a one-dimensional allocation problem, as they assume that the channel conditions are equal across all PRBs.

The authors of [YZR20] aim to provide long-term proportional fairness to eMBB users in the downlink, while simultaneously fulfilling the delay and reliability demands of URLLC users. Although they jointly consider eMBB and URLLC users, their resource allocation scheme is in fact a two-step process. First, downlink RAN resources are allocated with the objective of providing proportional fairness to the eMBB users. Next, the demands of the URLLC users are considered and RAN resources are reallocated to fulfill the latency demands of URLLC users. While the presented approach uses very strict assumptions regarding the latency, it lacks realistic assumptions regarding the channel conditions, i.e., varying channel quality indicator (CQI) values across PRBs. Furthermore, only one type of fairness, i.e., proportional fairness, is considered, and the processing of the data is not included in the system model.

Finally, the authors of [DP19] analyze different questions on URLLC RAN resource allocation. While they define an optimization problem where the sum over users satisfying their service level agreement (SLA) is maximized, they do not provide a solution to the problem but just an analysis of its NP-hardness. Additionally, they cover the problem of deciding whether a given set of users can be scheduled such that their SLAs are fulfilled. They provide a feasible resource allocation in polynomial time. However, the given solution is not optimal and per-PRB rates are either zero or a fixed number, which is a simplified approach compared to the assumptions in this work.

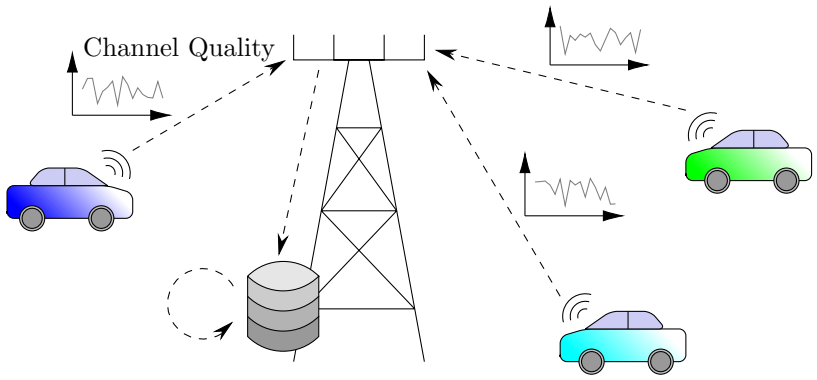
3 System Model

This chapter is dedicated to a general introduction of the system model that is contemplated in this work. Thereby, the different considered scenarios and types of resources are explained in detail. Furthermore, necessary assumptions that are made when working on the optimization problems are presented and related to well-known technical standards.

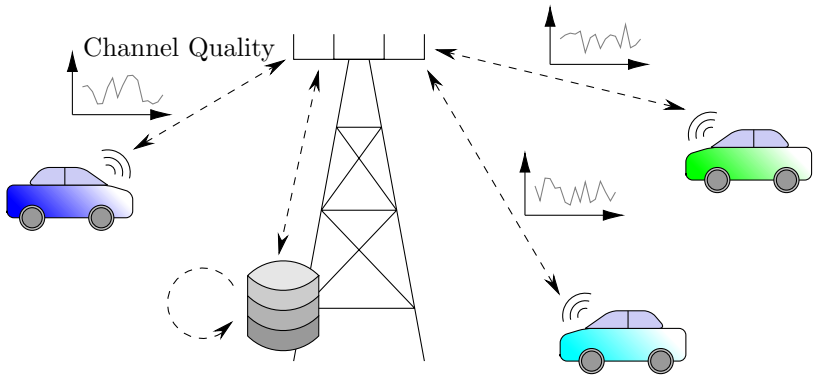
The possibility of network slicing in 5G NR, which introduced a paradigm shift in the operation of cellular networks, enables allocating *dedicated* network resources to users with the same type of service, e.g., users requesting URLLC services that have the same reliability and latency demands. Network slicing can be considered as creating logically independent networks that serve distinct users or various use cases which are grouped by their characteristics, e.g., the level of security that is needed to protect their traffic [EJAGS19]. These logical networks are, however, operated on the same physical networks [DPS20]. Over the course of this work, it is assumed that all considered users are requesting similar services. This implies that all users require the same service quality, meaning that they are assigned to the same dedicated network slice.

Resource allocation in a network slice is performed in two dimensions, *time* and *frequency*. In general, since release 17, 5G NR supports 7 different numerologies in 2 frequency ranges [ETS22a, ETS22b]. A numerology defines the subcarrier spacing (SCS) and the cyclic prefix length that is used for the transmission [DPS20]. In this thesis, it is assumed that all users are situated in the coverage area of a 5G macro base station (gNodeB) and are operating in the frequency range 1 (FR1), i.e., in the frequency range from $0.45-7.125$ GHz. In FR1, only the numerologies 0 to 2, i.e. SCSs of 15, 30, and 60 kHz, are applicable. As the unit of allocation in the frequency domain, 5G NR uses *PRBs*. Independent of the SCS, a PRB spans over 12 subcarriers. In the time domain, resource allocations are carried out on a per-slot basis, where slots are grouped into frames with a length of 10 ms. The slot duration, in turn, depends on the SCS and can be calculated as $1\,\text{ms}/2^{\mu}$, where μ is the used numerology.

Two similar scenarios are covered in this thesis. In the first scenario, there are N vehicular users simultaneously requesting a service by sending a packet to the BS via the uplink, where each inquiry is processed. The set of all users is denoted by $\mathcal{U}$. To enable the communication, there are K_u PRBs available in the uplink RAN. The set of uplink PRBs is denoted by $\mathcal{K}_u$. Additionally, L edge computing resources, which can be virtual machines for instance, are available for the receiving entity to process the information, where the processing rates p of all edge computing resources are the same. For the second scenario,



(a) Uplink only scenario.



(b) Uplink/downlink scenario.

Figure 3.1: Illustration of the system models.

there are again N users requesting a service, however, this time, the edge server that is co-located with the BS is generating a response packet that is returned to each user via the downlink. To this end, there are K_d PRBs available in the downlink RAN. Correspondingly, the set of downlink PRBs is denoted by $\mathcal{K}_d$. The two scenarios are depicted in Figure 3.1a or Figure 3.1b, respectively. Especially note the bidirectional arrays in Figure 3.1b.

5G NR offers three different duplex schemes to enable uplink and downlink communication for the users. These are time-division duplex (TDD), frequency-division duplex (FDD), and half-duplex frequency-division duplex (half-duplex FDD). TDD is characterized as the transmission where uplink and downlink communication happen on the same frequency but are separated in time. As opposed to that, the transmission in the FDD mode occurs at the same time but is segregated in the frequency domain, meaning that different carriers are used for the two links. Finally, half-duplex FDD is a mixture of the former two transmission modes, i.e., the communication is separated in both time and frequency [DPS20].

It is assumed that the channel conditions in the uplink and downlink change over time, i.e., they vary from one frame to another. The reason for this assumption is the time-varying nature of the channels that happens due to, e.g., interference, frequency-selective fading, or

path loss, and the mobility of the vehicular users. However, it is assumed that the channels are flat at a given time (frame), i.e., the channel conditions do not change during the frame. Moreover, users experience different channel conditions, i.e., different CQI values, across various PRBs even within the same frame. The CQI values are integers in the range from 1 to 15 [ETS22c] and depend on the signal-to-interference-plus-noise-ratio (SINR) that a user experiences. The SINR is defined as

$$\text{SINR} = \frac{\text{Signal Power}}{\text{Interference} + \text{Noise Power}} \tag{3.1}$$

and measured using special synchronization and reference signals. Depending on the CQI, the type of modulation (QPSK, 16QAM, 64QAM, or 256QAM) and the code rate are specified [ETS22c]. Given this information, the per-PRB data rate for a user can be calculated. With the data rate given per PRB and per frame for every user, the scheduling can now be performed across the two dimensions time and frequency.

Having in mind the previous assumptions on the varying channel conditions, it follows that in every frame user i's per-block rate can be modeled as a discrete random variable, $R_i \in \{r_1, r_2, ..., r_{15}\}$, such that $r_1 < r_2 < ... < r_{15}$. The corresponding probability mass function (PMF) is denoted by $\rho_{R_i}(r_k)$, and is a function of user i's SINR over time.

Since the focus of this work is set on URLLC traffic, the procedure of sending and processing the information must be executed within a maximum time T_{max} and moreover must be extremely reliable.[1] If t_i denotes the total delay user i is experiencing, the reliability is described by

$$\mathbb{P}\left(t_i \leq T_{max}\right) \geq 1 - \epsilon, \quad \forall i \in \mathcal{U} \tag{3.2}$$

where ϵ denotes the outage probability that has a very small value. E.g., if the requirement is a reliability of 99 %, the value of ϵ is 0.01. Note that there is also the propagation delay contributing to the latency. Nevertheless, there are two reasons it is not considered here. The first is that it cannot be affected, and the second is that the propagation delays are much lower than transmission and computing delays. Therefore, it is assumed that the transmission and computation times comprise the latency. In order to fulfill this strict requirement on reliability and latency, at least one PRB (both in the uplink and downlink) and one edge computing resource must be assigned to every user, as otherwise the delay constraint cannot be fulfilled. Naturally, a PRB and also a computing resource can only be allocated to one user and the resource can either be fully allocated or unassigned.

Finally, it is assumed that the users' packets are generated periodically on a per frame basis. The corresponding uplink and downlink data sizes $\Delta_{\{u,d\},i}$ can vary from one user to another, such that users have the ability to request different services but with the same service quality. As URLLC traffic packets are small [NORDS$^+$20], and not too many of them are transmitted simultaneously, it is assumed that the data generated at once is transmitted with the same rate.

[1] In practice, this latency is in the order of milliseconds, with the reliability requirement usually being above 99 %.

4 Admission Control for Uplink Communication with Edge Processing

Before addressing the problem of optimal resource allocation, it first needs to be determined whether all users requesting a service can actually be handled by the network, i.e., it needs to be decided whether enough resources are available to provide the desired quality of service (QoS) to each user. In this chapter, two admission policies are developed for the scenario of uplink communication with edge processing of the received data. To this end, in the primal section of this chapter, assumptions that are only valid for the scope of admission control are introduced to allow for an analytic tractability of the subsequent derivations. In the following two sections, the first policy, which is applicable to a homogenous set of users, i.e., users experiencing the same channel conditions, as well as the second policy, which is designed for a heterogenous set of users, i.e., users with various channel conditions, is presented. The chapter is finalized with a performance evaluation and a short summary of the admission control policies.

The analyses and results of this chapter were submitted to NOMS 2023 - 36th IEEE/IFIP Network Operations and Management Symposium [MHK22].

4.1 Tractability Assumptions

For analytical tractability, simplifying assumptions are made in this chapter compared to the system model introduced in Chapter 3. Namely, it is assumed that the BS splits the transmission power equally among all PRBs it transmits on, and that the channel characteristics for a user remain static across all PRBs (identical CQI values over all PRBs for a given user). The CQI values are still assumed to change randomly (according to some distribution) from one frame to another and are mutually independent among users. These assumptions reduce the RAN allocation to the number of allocated PRBs such that it is irrelevant which PRBs are assigned to a user. Furthermore, it is assumed that the packet sizes are identical, i.e., $\Delta_{u,i} = \Delta_u$. Hence, if a user receives $b_{u,i}$ uplink PRBs and m_i processing resources, the delay it experiences is calculated as

$$t_i = \frac{\Delta_u}{b_{u,i} R_i} + \frac{\Delta_u}{m_i p}. \tag{4.1}$$

4.2 Admission Policy for Homogenous Sets of Users

In this section, first, an analytical admission control approach for homogenous users is presented. The general idea is to derive the maximum number of users by equally sharing all resources among them. This approach will, however, lead to a policy that is physically impossible. By altering the approach, a valid admission policy is derived, which is not analytically tractable though. Hence, finally, another approach which is based on an algorithm employing binary search and the optimal trade-off between RAN and processing resources is developed. Recall that the homogeneity implies that all users are experiencing the same channel conditions, i.e., their per-block rates follow the same distribution, which is denoted by the random variable R_i as introduced in Chapter 3. This means that the random variable R_i is not dependent on user i in the scenario considered here.

4.2.1 Equal-Share Approach

When pursuing the equal-share approach, the number of PRBs an admitted user receives is $\frac{K_u}{N}$. Similarly, every user will receive $\frac{L}{N}$ edge computing resources. The maximum number of users that can be admitted in the cell in this way is determined as follows:

Substituting the aforementioned facts into (4.1) and (3.2) and rearranging this equation leads to

$$\mathbb{P}\left(\frac{1}{R_i} \leq \frac{K_u T_{max}}{N\Delta_u} - \frac{K_u}{Lp}\right) \geq 1 - \epsilon. \tag{4.2}$$

The left-hand side of (4.2) is the cumulative distribution function (CDF) of the inverse of the per-PRB rate at point $\frac{K_u T_{max}}{N\Delta_u} - \frac{K_u}{Lp}$. Hence,

$$F_{\frac{1}{R_i}}\left(\frac{K_u T_{max}}{N\Delta_u} - \frac{K_u}{Lp}\right) \geq 1 - \epsilon \tag{4.3}$$

must hold. As the CDF is a monotonously increasing function, (4.3) yields

$$\frac{K_u T_{max}}{N\Delta_u} - \frac{K_u}{Lp} \geq F^{-1}_{\frac{1}{R_i}}(1-\epsilon), \tag{4.4}$$

where $F^{-1}_{\frac{1}{R_i}}(1-\epsilon)$ is the inverse of the CDF at point $1-\epsilon$. Lastly, using simple algebraic operations, the upper bound

$$N \leq \frac{K_u T_{max}}{\Delta_u} \cdot \frac{1}{F^{-1}_{\frac{1}{R_i}}(1-\epsilon) + \frac{K_u}{Lp}}, \tag{4.5}$$

or equivalently,

$$N_{max} = \frac{K_u T_{max}}{\Delta_u} \cdot \frac{1}{F^{-1}_{\frac{1}{R_i}}(1-\epsilon) + \frac{K_u}{Lp}} \tag{4.6}$$

is obtained.

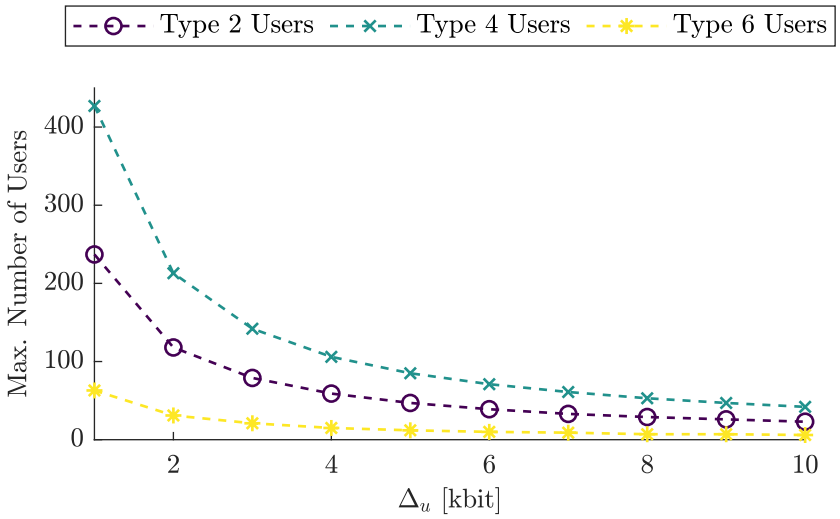


Figure 4.1: Maximum number of users that can be admitted in the cell depending on the data size Δ_u for different types of sets of homogenous users.

Next, by evaluating the above admission policy in an example scenario, it is described why the above result is physically impossible, i.e., it is not correct. For a subcarrier spacing of 30 kHz, the maximum number of available PRBs is $K_u = 273$ [ETS22a]. The number of processing units on the edge cloud is assumed to be $L = 500$, with each processing unit providing a processing rate of $p = 1$ Mbps. The maximum allowed latency is $T_{max} = 5$ ms. Different packet sizes Δ_u are considered simultaneously. Figure 4.1 illustrates the maximum number of users that can be admitted for this type of URLLC traffic vs. the data size for three different types of users. The users are characterized by their experienced channel characteristics given in Table 4.1, which describes the probabilities of experiencing a specific data rate for six different users.

What can be observed from Figure 4.1 is the fact that when all the users of type 4 transmit regularly (periodically) data of size $\Delta_u = 1$ kbit, a total of 430 users can be admitted in the cell. However, this stands in contradiction to the maximum number of available PRBs, which is (only) 273. As the granularity level in resource allocation in 5G NR is the PRB per slot, the maximum number of simultaneously transmitting users in this scenario could only be 273, not 430 as implied by (4.6). The reason for this contradiction stems from the fact that with the above approach a user can receive a non-integer number of PRBs or processing units, i.e., $\frac{K_u}{N}$ and $\frac{L}{N}$, respectively. This would lead to a user receiving an amount of PRBs lower than 1, e.g., 0.7, especially in case there are users with good channel conditions and a low amount of submitted data. This, for apparent reasons, is infeasible.

The correct way of writing the amount of uplink PRBs a user receives, if there are in total N users, would be $\left\lfloor \frac{K_u}{N} \right\rfloor$, whereas the number of processing units in the edge cloud would

Table 4.1: CQI values, per-PRB rates and the corresponding probabilities for six users from the Republic of Ireland trace [RLSQ20, ML22]

CQI	1	2	3	4	5	6	7	8
R (kbps)	48	73.6	121.8	192.2	282	378	474.2	712
$\rho_{R_1}(r_k)$	0	0.1	0.72	0.04	0.05	0.09	0	0
$\rho_{R_2}(r_k)$	0	0	0.2	0.7	0.1	0	0	0
$\rho_{R_3}(r_k)$	0	0	0	0	0.01	0.12	0.51	0.32
$\rho_{R_4}(r_k)$	0	0	0	0	0	0.01	0.98	0.01
$\rho_{R_5}(r_k)$	0.24	0.04	0.07	0.04	0.04	0.06	0.16	0.15
$\rho_{R_6}(r_k)$	0.18	0.11	0.1	0.06	0.05	0.1	0.17	0.11
CQI	9	10	11	12	13	14	15	
R (kbps)	772.2	874.8	1063.8	1249.6	1448.4	1640.6	1778.4	
$\rho_{R_1}(r_k)$	0	0	0	0	0	0	0	
$\rho_{R_2}(r_k)$	0	0	0	0	0	0	0	
$\rho_{R_3}(r_k)$	0.01	0.01	0.02	0	0	0	0	
$\rho_{R_4}(r_k)$	0	0	0	0	0	0	0	
$\rho_{R_5}(r_k)$	0.01	0.01	0.06	0.06	0	0.03	0.03	
$\rho_{R_6}(r_k)$	0.02	0.04	0	0.03	0	0.02	0.01	

be $\left\lfloor \frac{L}{N} \right\rfloor$. Combining (3.2) and (4.1) would then result in

$$\mathbb{P}\left(\frac{\Delta_u}{\left\lfloor \frac{K_u}{N} \right\rfloor R_i} + \frac{\Delta_u}{\left\lfloor \frac{L}{N} \right\rfloor p} \leq T_{max} \right) \geq 1 - \epsilon. \tag{4.7}$$

Note that $\left\lfloor \frac{K_u}{N} \right\rfloor \neq K_u \left\lfloor \frac{1}{N} \right\rfloor$. Hence, solving inequality (4.7) is not analytically tractable.

Given the previous reasoning, in the subsequent subsection, a different approach in determining the maximum number of URLLC users that can be admitted in the cell, while taking into account the two types of resources (uplink RAN and processing units), is followed.

4.2.2 Trade-off between Network and Processing Resources

In this section, the optimal trade-off between the number of PRBs and processing units that need to be allocated to a user, such that the number of admitted homogenous users is maximized, is determined. In order to derive the admission policy for a reliability of $1 - \epsilon$, first, a result for the strictest possible reliability, i.e. 100 %, is deduced.

The number of PRBs user i will receive is denoted as $b_{u,i}$ and m_i stands for the number of allocated computing resources. Since the set of users is assumed to be homogenous in this section, $b_{u,i}$ and m_i will be the same for every user i. It holds that $\sum_i b_{u,i} \leq K_u$ and $\sum_i m_i \leq L$. In order to derive a continuous trade-off feasibility border and to admit as

many users as possible, let $b_{u,i}$ and m_i be continuous variables and set the delay every user is experiencing to T_{max}. As the reliability requirement is 100 %, the following equation can be stated:

$$\frac{\Delta_u}{b_{u,i}R_i} + \frac{\Delta_u}{m_i p} = T_{max}. \tag{4.8}$$

Further, since (4.8) must always be satisfied, it needs to be considered for the worst-case scenario in terms of the channel conditions, i.e., when the user experiences the lowest per-PRB rate, because in that case the user needs the largest amount of resources to meet the latency requirement.

Let $r_{min,i} = \min_k \{r_k | \rho_{R_i}(r_k) > 0, k \in \{1, \ldots, 15\}\}$ denote the lowest possible per-PRB rate for user i, where r_k corresponds to the data rates given in Table 4.1 for the specific CQI values k. Again, $r_{min,i}$ is the same for every user i. Using the introduced worst-case scenario data rate of a user, (4.8) transforms into

$$\frac{\Delta_u}{b_{u,i}r_{min,i}} + \frac{\Delta_u}{m_i p} = T_{max}. \tag{4.9}$$

From (4.9), the amount of needed computation resources as a function of the number of assigned PRBs can be expressed as

$$m_i = \frac{1}{p} * \frac{1}{\frac{T_{max}}{\Delta_u} - \frac{1}{b_{u,i}r_{min,i}}} \tag{4.10}$$

by dividing by Δ_u, subtracting the transmission delay, multiplying by p, and then inverting both sides of the equation. Since $m_i > 0$, from (4.10), it must hold that $\frac{T_{max}}{\Delta_u} > \frac{1}{b_{u,i}r_{min,i}}$, resulting in

$$b_{u,i} > \frac{\Delta_u}{r_{min,i}T_{max}}. \tag{4.11}$$

Following the same reasoning for m_i, from (4.9), it must hold that $\frac{T_{max}}{\Delta_u} > \frac{1}{m_i p}$, leading to

$$m_i > \frac{\Delta_u}{pT_{max}}. \tag{4.12}$$

Until now, the infima of the amount of RAN $\left(\frac{\Delta_u}{r_{min,i}T_{max}}\right)$ and edge computing $\left(\frac{\Delta_u}{pT_{max}}\right)$ resources that are needed were determined. Apparently, to satisfy the latency constraint, there are multiple combinations of $(b_{u,i}, m_i)$ possible. Providing more RAN resources (reducing the transmission delay) will compensate for the allocation of fewer computing resources (higher processing delay). Thus, the question that arises is, *what is the optimal combination of* $(b_{u,i}, m_i)$ *that will enable admitting the highest number of URLLC users?* To answer this question, first, the dependency of m_i on $b_{u,i}$ must be understood. To that end, the first derivative of $m_i(b_{u,i})$ from (4.10) is calculated as

$$m_i{}'(b_{u,i}) = \frac{-1}{pr_{min,i}b_{u,i}^2} * \frac{1}{\left(\frac{T_{max}}{\Delta_u} - \frac{1}{b_{u,i}r_{min,i}}\right)^2} < 0, \tag{4.13}$$

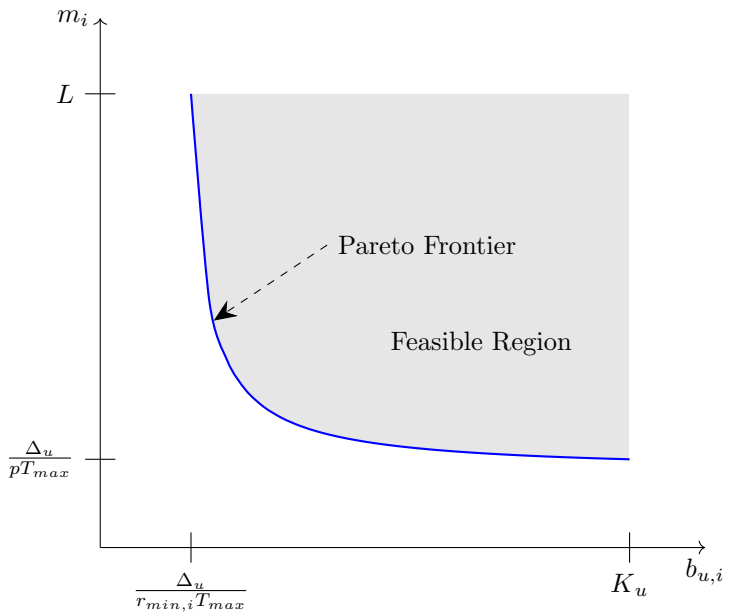


Figure 4.2: The general shape of the Pareto frontier and the feasible region for the amount of needed PRBs and processing units in the edge cloud so that the latency requirement is met with a reliability of 100 %. Note the dependence of the Pareto frontier on the worst-case channel conditions ($r_{min,i}$).

implying that m_i is a monotonous decreasing function in $b_{u,i}$. For the second derivative,

$$\begin{aligned} m_i''(b_{u,i}) &= \frac{1}{\left(\frac{T_{max}}{\Delta_u} - \frac{1}{b_{u,i}r_{min,i}}\right)^2} * \frac{2}{pr_{min,i}b_{u,i}^3} + \frac{-1}{pr_{min,i}b_{u,i}^2} * \frac{-2}{\left(\frac{T_{max}}{\Delta_u} - \frac{1}{b_{u,i}r_{min,i}}\right)^3 r_{min,i}b_{u,i}^2} \\ &= \frac{2}{pr_{min,i}\left(\frac{T_{max}b_{u,i}}{\Delta_u} - \frac{1}{r_{min,i}}\right)^3}\left(\frac{T_{max}}{\Delta_u} - \frac{1}{b_{u,i}r_{min,i}} + \frac{1}{b_{u,i}r_{min,i}}\right) \\ &= \frac{2T_{max}}{p\Delta_u r_{min,i}} * \frac{1}{\left(\frac{T_{max}b_{u,i}}{\Delta_u} - \frac{1}{r_{min,i}}\right)^3} \end{aligned} \tag{4.14}$$

is obtained, which is always greater than zero due to (4.11). Thus, (4.14) implies the fact that m_i is a convex function in $b_{u,i}$.

Taking into account (4.10)-(4.12), the general shape of the dependency between m_i and $b_{u,i}$ for a given $r_{min,i}$ is derived. The feasible region of the values for the ordered pair $(b_{u,i}, m_i)$ is shown in Figure 4.2. Having this in mind, the optimal combination can be found along the curve shown in Figure 4.2. This is the well-known *Pareto frontier* [BV04].

Note: In Figure 4.2, the general "continuous" Pareto frontier is shown to illustrate the dependency of m_i on $b_{u,i}$. In practice, the curve would be a discrete function where the

values of m_i would have to be rounded up such that the delay constraint is fulfilled. However, the shape of the frontier would not change.

The convexity of $m_i(b_{u,i})$ provides an interesting insight. Namely, reducing the value of $b_{u,i}$ implies a higher incline in m_i, and vice versa (the *increasing return* property of convex functions, the opposite of the *diminishing return* property encountered in concave functions). Because of this observation, the need arises to choose the point on the Pareto frontier as the solution to the allocation problem, where both the RAN and edge computing resources are sufficient to satisfy the traffic demands of the largest possible number of users.

The final question that needs to be answered now is: *What is the optimal choice on the Pareto frontier?* Denote by $(b_{u,0}, m_0)$ any ordered set of points on the Pareto frontier. For any such point, the maximum number of users that can be admitted if considering only the amount of needed RAN resources $b_{u,0}$ would be $\left\lfloor \frac{K_u}{b_{u,0}} \right\rfloor$. In case the number of users is determined based solely on the number of processing resources, its maximum number would be $\left\lfloor \frac{L}{m_0} \right\rfloor$. Therefore, for a given point $(b_{u,0}, m_0)$ on the Pareto frontier, if both resources are considered, the maximum number of users that can be admitted is

$$N_{max}(b_{u,0}, m_0) = \left\lfloor \min\left(\frac{K_u}{b_{u,0}}, \frac{L}{m_0} \right) \right\rfloor. \tag{4.15}$$

When looking over the entire possible set of ordered pairs $(b_{u,0}, m_0)$, the following result for the maximum number of admitted users in the cell can be stated:

Result 1. *Given a BS with K_u uplink PRBs and L edge computing resources (with a processing rate of p per computing resource), users with URLLC traffic that should never experience a latency higher than T_{max}, i.e., with a joint delivery and processing reliability of 100 %, whose worst-case per-PRB rate is $r_{min,i}$, and who transmit packets with a total data size Δ_u, the maximum number of users that can be admitted in the cell is given as*

$$N_{max} = \max_{b_{u,0}, m_0} \left\{ \left\lfloor \min\left(\frac{K_u}{b_{u,0}}, \frac{L}{m_0} \right) \right\rfloor \right\}, \tag{4.16}$$

where the ordered set $(b_{u,0}, m_0)$ satisfies the inequality

$$\frac{\Delta_u}{b_{u,0} r_{min,i}} + \frac{\Delta_u}{m_0 p} \leq T_{max}. \tag{4.17}$$

The interesting thing to observe from Result 1 is that with this approach for all the users with the same lowest possible CQI, the amount of resources needed is the same, i.e., this approach is valid not only for users with identical per-PRB rate distributions, *but for all users with the same lowest CQI*. Said differently, the approach is oblivious to the entire channel condition statistics. Apparently, the higher the $r_{min,i}$ that all users are experiencing, the higher the number of admitted users.

4.2.3 Admission Control with General Reliability

For a reliability lower than 100 % and homogenous users, the constraint (3.2) that needs to be fulfilled for every user i reads as

$$\mathbb{P}\left(\frac{\Delta_u}{b_{u,i}r_{min,i}} + \frac{\Delta_u}{m_i p} \leq T_{max}\right) \geq 1-\epsilon, \quad \forall i \in \mathcal{U}, \tag{4.18}$$

where $b_{u,i}$ and m_i are integers. Note that $b_{u,i}$, $r_{min,i}$, and m_i are still the same for all users i. Obviously, relaxing the reliability requirement should lead to an increased number of admitted users. However, determining that number is not feasible via a closed-form expression. Instead, a rather different approach based on an algorithm employing binary search is used.

W.l.o.g., it is assumed that all per-PRB rates are possible, i.e., $\rho_{R_i}(r_k) > 0, \forall k \in \{1, ..., 15\}$. Then, in a given setup, for the data rate r_k from Table 4.1 that is the worst possible per-PRB rate, i.e., $r_{min,i} = r_k$, the maximum number of admitted users is found using (4.16) when $\epsilon = 0$. This number is denoted as $N(r_{min,i})$. Note that this number is a lower bound, as relaxing the reliability to $\epsilon > 0$ enables more users to be admitted in the cell.

Next, the "minimum" possible per-PRB rate is increased to the next possible value and denoted as $r^+_{min,i}$, e.g., if $r_{min,i} = r_6$, then $r^+_{min,i} = r_7$. For the latter value, using (4.16), the corresponding maximum number of users that can be admitted for $\epsilon = 0$ is obtained. This number is the new reference value, corresponding to the integer allocations $b^+_{u,i}$ and m^+_i. As this new reference value was planned for better channel conditions, but with strict reliability of 100 %, it is checked whether this new number of users, which is denoted as $N(r^+_{min,i})$, can be admitted in the cell such that their latency is satisfied with a reliability of $1-\epsilon$. It must hold that

$$\mathbb{P}\left(\frac{1}{b^+_{u,i}r_{min,i}} + \frac{1}{m^+_i p} \leq \frac{T_{max}}{\Delta_u}\right) \geq 1-\epsilon. \tag{4.19}$$

If the previous condition is satisfied, then $r^+_{min,i}$ is increased to the next higher per-PRB rate, and using (4.16) the new $b^+_{u,i}$ and m^+_i, as well as the new $N(r^+_{min,i})$, are found. Afterwards, it is once more checked if the updated condition (4.19) is satisfied. If it holds, the procedure continues until the corresponding (4.19) is not fulfilled for the first time.

When the latter is the case, then it is known that this number of users cannot be admitted. Nevertheless, it is also known that it is possible to admit the number of users corresponding to the previous $r^+_{min,i}$ for ϵ. Hence, an upper and a lower bound on the maximum number of users that can be admitted were determined. Therefore, the binary search algorithm [GG07] can be employed to find the largest possible number $N_{max}(r_{min,i}, \epsilon)$ between $N(r_{min,i})$ and $N(r^+_{min,i})$. For every iteration of the binary search, using (4.19), it is checked whether the "new" number of users can be admitted to the network. If yes, the upper interval is taken as the new range, while the lower interval is correspondingly taken over as the new range

Algorithm 1 Admission Policy Providing General Reliability for Homogenous Users

Input: $r_{min,i}$, K_u, L, p, T_{max}, Δ_u, ϵ

Output: $N_{max}(r_{min,i}, \epsilon)$, $b_{u,i} \in \mathbb{N}$, $m_i \in \mathbb{N}$

function GenRelAdmission($r_{min,i}$, K_u, L, p, T_{max}, Δ_u, ϵ)
  Calculate $N(r_{min,i}) = \max_{b_{u,i},m_i} \left\{ \left\lfloor \min\left(\frac{K_u}{b_{u,i}}, \frac{L}{m_i}\right) \right\rfloor \right\}$ s.t. $\frac{1}{b_{u,i} r_{min,i}} + \frac{1}{m_i p} \leq \frac{T_{max}}{\Delta_u}$.
  Note $b_{u,i}$ and m_i.
  Set $r^+_{min,i}$ to r_k where $k = j + 1$ if $r_{min,i} = r_j$.
  Calculate $N(r^+_{min,i}) = \max_{b^+_{u,i},m^+_i} \left\{ \left\lfloor \min\left(\frac{K_u}{b^+_{u,i}}, \frac{L}{m^+_i}\right) \right\rfloor \right\}$ s.t. $\frac{1}{b^+_{u,i} r^+_{min,i}} + \frac{1}{m^+_i p} \leq \frac{T_{max}}{\Delta_u}$.
  Note $b^+_{u,i}$ and m^+_i.
  while $\mathbb{P}\left(\frac{1}{b^+_{u,i} r_{min,i}} + \frac{1}{m^+_i p} \leq \frac{T_{max}}{\Delta_u}\right) \geq 1 - \epsilon$ and $r^+_{min,i} < r_{15}$ **do**
    Set $N(r_{min,i}) = N(r^+_{min,i})$, $b_{u,i} = b^+_{u,i}$, and $m_i = m^+_i$.
    Increase $r^+_{min,i}$ to the next higher r_k, i.e., set $k = k + 1$.
    Calculate $N(r^+_{min,i}) = \max_{b^+_{u,i},m^+_i} \left\{ \left\lfloor \min\left(\frac{K_u}{b^+_{u,i}}, \frac{L}{m^+_i}\right) \right\rfloor \right\}$ s.t. $\frac{1}{b^+_{u,i} r^+_{min,i}} + \frac{1}{m^+_i p} \leq \frac{T_{max}}{\Delta_u}$.
    Note $b^+_{u,i}$ and m^+_i.
  end while
  while $N(r^+_{min,i}) - N(r_{min,i}) > 1$ **do**
    Set $b^t_{u,i} = \left\lfloor \frac{K_u}{N(r_{min,i}) + \left\lfloor \frac{N(r^+_{min,i}) - N(r_{min,i})}{2} \right\rfloor} \right\rfloor$, $m^t_i = \left\lfloor \frac{L}{N(r_{min,i}) + \left\lfloor \frac{N(r^+_{min,i}) - N(r_{min,i})}{2} \right\rfloor} \right\rfloor$.
    if $\mathbb{P}\left(\frac{1}{b^t_{u,i} r_{min,i}} + \frac{1}{m^t_i p} \leq \frac{T_{max}}{\Delta_u}\right) \geq 1 - \epsilon$ **then**
      Set $N(r_{min,i}) = N(r_{min,i}) + \left\lfloor \frac{N(r^+_{min,i}) - N(r_{min,i})}{2} \right\rfloor$, $b_{u,i} = b^t_{u,i}$, and $m_i = m^t_i$.
    else
      Set $N(r^+_{min,i}) = N(r_{min,i}) + \left\lfloor \frac{N(r^+_{min,i}) - N(r_{min,i})}{2} \right\rfloor$.
    end if
  end while
  return $N_{max}(r_{min,i}, \epsilon) = N(r_{min,i})$, $b_{u,i}$, m_i
end function

if the condition was not fulfilled. This procedure is repeated until the largest number of users which satisfies (4.19) is found.

The described method is summarized in Algorithm 1. Note that since the binary search is used when performing the search, at most $15 + \left\lfloor \log_2(N(r^+_{min,i}) - N(r_{min,i}) + 1) \right\rfloor$ evaluations are performed when applying the algorithm.

4.3 Admission Policy for Heterogenous Sets of Users

When it comes to a set of users with heterogenous conditions, a simple admission policy for a newly arriving user that is valid for any ϵ is provided. The first step is to check whether the newly arriving user and the current $N-1$ users receiving service satisfy the inequality (heterogenous users have different $r_{min,i}$)

$$\sum_{i=1}^{N} \frac{1}{r_{min,i}} \leq K_u \left(\frac{T_{max}}{\Delta_u} - \frac{1}{\lfloor L/N \rfloor \, p} \right). \tag{4.20}$$

If that is the case, then the *new user can be admitted*. Condition (4.20) is obtained by combining the latency equation (4.1) and $\sum_{i=1}^{N} b_{u,i} \leq K_u$ while assuming an equal share of computing resources. Solving the latency equation for $b_{u,i}$ results in

$$b_{u,i} = \frac{1}{R_i \left(\frac{t_i}{\Delta_u} - \frac{1}{m_i p} \right)}. \tag{4.21}$$

Plugging in T_{max} for t_i, the equal share of computing resources, i.e., $m_i = \lfloor L/N \rfloor$, and assuming every user is experiencing its worst possible data rate gives

$$b_{u,i} = \frac{1}{r_{min,i} \left(\frac{T_{max}}{\Delta_u} - \frac{1}{\lfloor L/N \rfloor p} \right)}. \tag{4.22}$$

Lastly, plugging in (4.22) into $\sum_{i=1}^{N} b_{u,i} \leq K_u$ and multiplying the inequality by the term in brackets in (4.22) leads to (4.20).

Condition (4.20) pertains to the case of $\epsilon = 0$. Essentially, if there are enough resources for the newly arriving user to be admitted for the most restrictive case, i.e., $\epsilon = 0$, the user can be admitted for any other lower reliability, i.e., higher ϵ. If user N has a high $r_{min,N}$, it would lead to a lower left-hand side of (4.20), and thus to a higher chance for the user to be admitted.

If (4.20) does not hold, it needs to be checked with what probability the worst-case scenario occurs, i.e., what is the probability that all the users will have their lowest corresponding per-PRB rates $r_{min,i}$ simultaneously. That probability is calculated as $\prod_{i=1}^{N} \rho_{R_i}(r_{min,i})$, and if it is lower than the outage, i.e., if $\prod_{i=1}^{N} \rho_{R_i}(r_{min,i}) \leq \epsilon$, it means that the planning can be done not for the worst-case per-PRB rate, but for higher rates, which in turn implies that fewer resources are needed for a user. This means that there are enough resources for user N to be admitted.

Summarizing, the following admission policy for heterogenous users can be stated:

Result 2. *Given a set of $N-1$ users with URLLC traffic in the cell, whose worst-case per-PRB rates are $r_{min,i}$, $i = 1, ..., N-1$, and a reliability requirement of $1-\epsilon$, a sufficient condition for a new user with worst-case per-PRB rate $r_{min,N}$ to be admitted is if one of the following holds:*

$$\sum_{i=1}^{N} \frac{1}{r_{min,i}} \leq K_u \left(\frac{T_{max}}{\Delta_u} - \frac{1}{\lfloor L/N \rfloor \, p} \right), \qquad or \tag{4.23}$$

$$\prod_{i=1}^{N} \rho_{R_i}(r_{min,i}) \leq \epsilon. \tag{4.24}$$

4.4 Performance Evaluation

After introducing the simulation setup for the subsequent evaluations, first the theoretical results for 100 % reliability regarding homogenous sets of users are validated. Then, examinations on scenarios with $\epsilon > 0$ and homogenous sets of users as well as on heterogenous sets of users are conducted. Finally, the performance of the admission control for homogenous sets of users is compared to other policies.

4.4.1 Simulation Setup

A 5G trace with data measured in the Republic of Ireland was used as input to the simulations. These traces are described in detail in [RLSQ20], and a statistical analysis is given in [ML21c]. The parameter of interest from the trace is the CQI with 15 levels, which serves to determine the experienced rate of a user in a frame. The measurements were conducted for one user, but at different days, for different applications, and when the user was static or moving around. To mimic the dynamic nature of the users from the simulation, only measurements where the user was moving were picked. Six different measurements were selected to mimic six different users for the subsequent simulations. Based on the frequency of occurrence of a per-PRB rate for every user, the corresponding per-PRB rate probabilities were obtained. They are given in Table 4.1.

Since the subcarrier spacing is assumed to be 30 kHz, the slot duration is 0.5 ms. Given that a block consists of 12 subcarriers, the PRB width is then 360 kHz. The total number of PRBs is $K_u = 273$ [ETS22a], whereas the total number of computing resources is $L = 500$, where the processing rate per resource is $p = 1$ Mbps. The simulations were conducted in MATLAB R2021b.

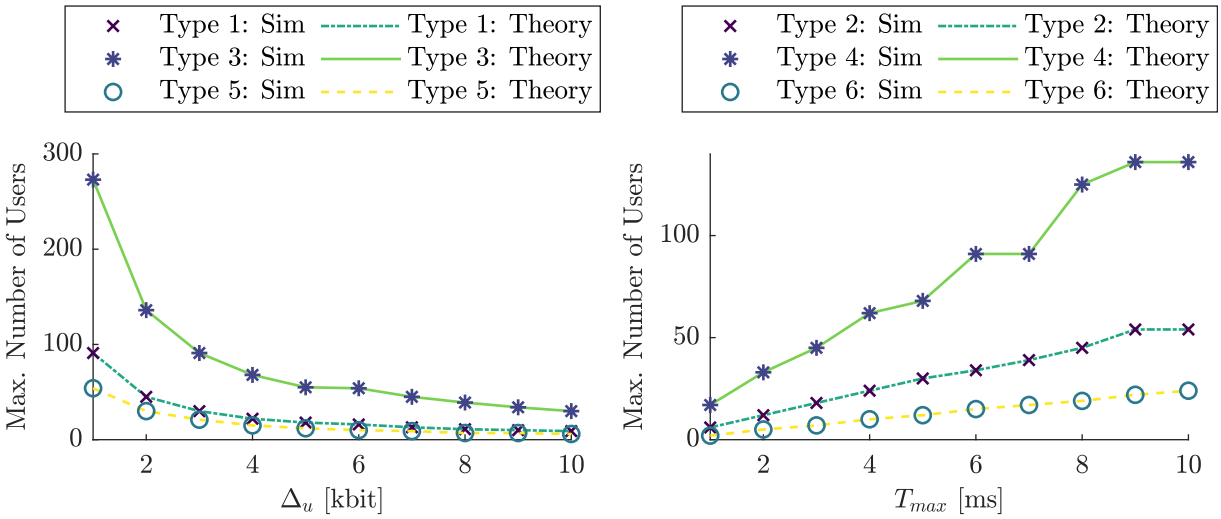
4.4.2 Validation of the Theoretical Result for Homogenous User Sets

First, the theoretical result for the maximum number of users from a homogenous set that can be admitted in the cell is validated (4.16). The reliability requirement is 100 %. Three types of users from Table 4.1 are considered: type 1, 3, and 5. The latency requirement is set to $T_{max} = 5\,\text{ms}$. To obtain simulation results, the number of users is increased one by one until there is a packet that is not sent and processed within the deadline. In case this happens, the last number of users for which all packets were handled within the maximum latency is taken as the maximum number that can be served by the network. Figure 4.3a shows the results vs. the size of the data that is transmitted at once. The first thing which can be noticed is the perfect match between the simulation and the theoretical results, which corroborates the validity of the analytical approach. The second observation is the decline in the admissions as the data size increases. This is to be expected as it would take more resources to deliver and process more data during the same time. The third outcome of the simulations is the higher number of type 3 users that can be admitted. The reason lies in the best worst-case channel conditions of user type 3. Namely, for user type 3, the worst per-PRB rate is r_5, as opposed to r_2 and r_1 for user types 1 and 5, respectively.

Next, the presented theoretical result is validated as a function of the maximum allowed latency (T_{max}). The reliability is again 100 %, i.e., $\epsilon = 0$. The simulation is executed in the same way as in the previous scenario. To introduce diversity, now, results for user types 2, 4, and 6 from Table 4.1 are shown. The data size is fixed to be 5 kbit. Figure 4.3b depicts the results. Again, there is a perfect match between theory and the simulations. At least 100 % more type 4 users can be admitted because the worst-case channel conditions for user type 4 (their lowest CQI is 6) are better than for the other two user types, which experience the worst CQI values of 3 and 1. While it is expected that relaxing the latency would always allow for a higher number of admitted users, this is not always the case. This happens for cases in which the latency requirement is already quite loose. The reason for this behavior is the unavailability of enough resources to fulfill the delay constraint for all users. Although the latency requirement is relaxed, the delay constraint could only be fulfilled if fractions of resources could be assigned to the users. Since this is not possible in reality, the loosened latency requirement sometimes has no influence on the number of users that can be admitted.

4.4.3 Studies on Outage Probabilities and Heterogenous User Sets

The results constituted so far pertain to the case of 100 % reliability. Next, the impact of reduced reliability on the number of admitted users is investigated for the case of a homogenous set of users. To that end, the user types 1, 3, and 5 are considered. The size of the data is $\Delta_u = 5\,\text{kbit}$, whereas the allowed latency is $T_{max} = 5\,\text{ms}$. Figure 4.4 depicts the results for different outages ϵ. What can be observed first is the higher number of type 3 users that can be admitted for the same reasons as in the scenario corresponding to



(a) $T_{max} = 5$ ms. (b) $\Delta_u = 5$ kbit.

Figure 4.3: The maximum number of users that can be admitted for 100 % reliability and varying data size/delay constraint.

Figure 4.3a. The second observation, which is rather surprising, is that the number of users that can be admitted does not increase drastically with the outage ϵ. This is completely different from the case when only the RAN limitations are considered when deciding on how many users can be admitted [ML21b]. The rationale behind this stems from the large number of users receiving service. Namely, for large N, $\lfloor K_u/N \rfloor = \lfloor K_u/(N+1) \rfloor$, and only where a shift down by 1 occurs, there is a jump in N_{max}.

Having considered the case of a homogenous set of users until now, subsequently, the performance of the admission policy for a heterogenous user set is evaluated. As there is not a high dependency on the number of admitted users on ϵ, the case of 100 % reliability is considered. The setup for deciding whether or not a URLLC user is admitted after some other users are already present in the cell is described as follows: Users of types 1-5 are already present in the cell, and it is decided whether a user of type 6 can be admitted. In all the following scenarios there are three type 1, three type 2 and three type 3 users, as well as two users of type 4 and type 5 already present. In total, there are 13 users present before the arrival of the user of type 6. Five different scenarios are considered in terms of Δ_u and T_{max}:

- Scenario A: $\Delta_u = 1$ kbit, $T_{max} = 5$ ms
- Scenario B: $\Delta_u = 2$ kbit, $T_{max} = 4$ ms
- Scenario C: $\Delta_u = 3$ kbit, $T_{max} = 3$ ms
- Scenario D: $\Delta_u = 4$ kbit, $T_{max} = 3$ ms
- Scenario E: $\Delta_u = 4$ kbit, $T_{max} = 2$ ms

Figure 4.5 shows the results. On the y-axis, the ratio of the LHS and RHS of (4.20) is

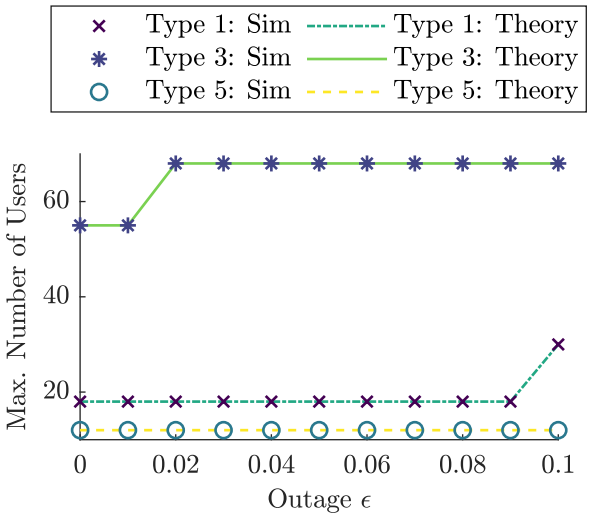


Figure 4.4: The maximum number of users that can be admitted for different reliabilities for $\Delta_u = 5\,\text{kbit}$ and $T_{max} = 5\,\text{ms}$.

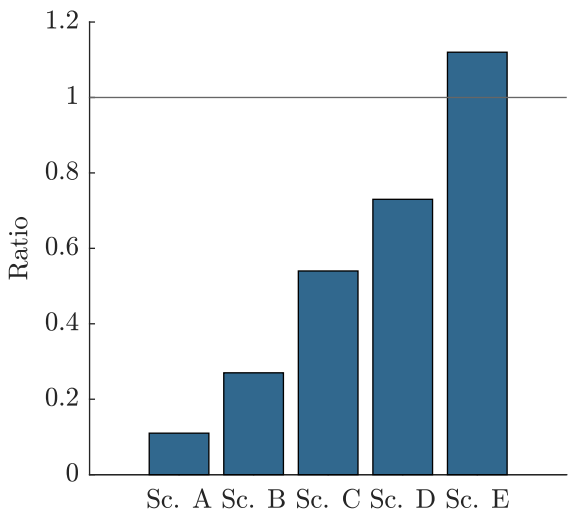
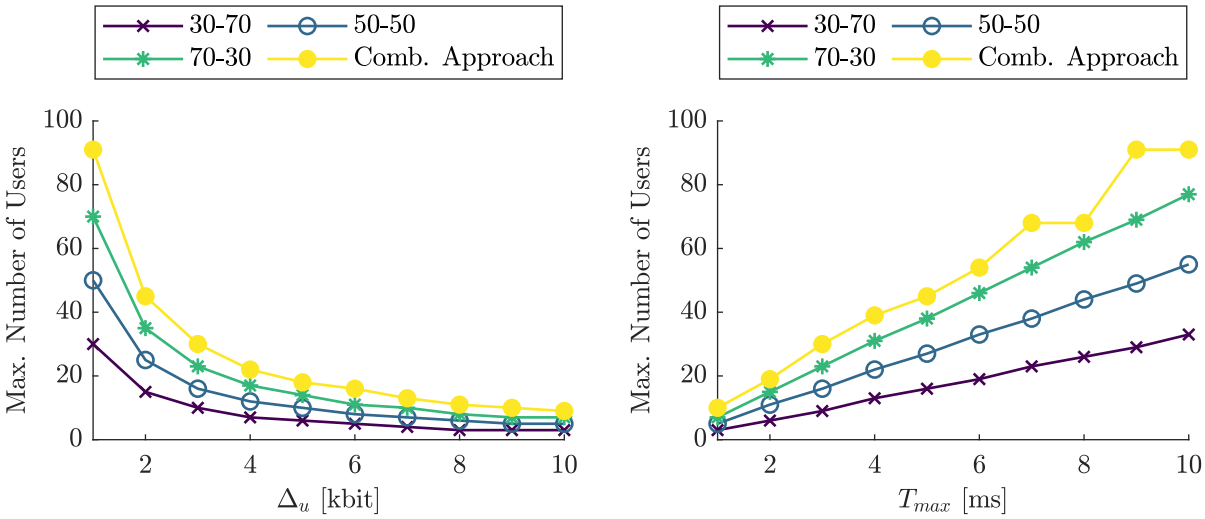


Figure 4.5: The decision whether to admit a newly arriving user of type 6 for different combinations of T_{max} and Δ_u, when $\epsilon = 0$ (heterogenous user set).

depicted. As long as this ratio is smaller than 1, the user of type 6 can be admitted in the cell. Note that for $\epsilon = 0$ condition (4.24) is never satisfied, so for a user to be admitted (4.23) must hold. As can be observed from Figure 4.5, in Sc. A-D the user of type 6 is always admitted. The reason is that in these cases the maximum latency is not lower than 3 ms, or the packet size is not large enough, or both requirements are not too restrictive. However, in Sc. E, both the data size is large and the latency is low. Hence, the resources are not sufficient to admit the user of type 6.

4.4.4 Performance Comparisons

In the last subsection, the developed approach of jointly allocating RAN and computing resources is compared to the approach in which the number of admitted users is decided separately for RAN and edge cloud resources. The user of interest is of type 1, and $T_{max} = 5\,\text{ms}$. Three "separate" approaches are considered. In the first, up to 50 % of the latency can be experienced during the transmission, while the other 50 % can be encountered during the processing. In the second approach, up to 30 % of the time can be dedicated to transmission and the remaining time to the processing. Vice versa, for the third type, up to 70 % of the time can be spent on transmission and 30 % on processing. For the separate allocation approach, the maximum number of users that can be admitted with regard to the RAN or computing resources, respectively, are determined. Then, the minimum of those two numbers is taken as the number of users that can be served. For all scenarios, the reliability is 100 %. Figure 4.6a shows the results as a function of the data size transmitted



(a) Users are of type 1, $T_{max} = 5$ ms.

(b) Users are of of type 2, $\Delta_u = 3$ kbit.

Figure 4.6: The number of users that can be admitted with the new joint approach and when splitting T_{max} strictly between transmission and processing for varying data size/delay constraint.

at once. As can be observed, the number of admitted users is the highest with the new joint approach, outperforming the others by at least 30 %. The reason is that one resource can compensate for the other, which is not possible with the separate approaches. The second thing to observe is that from the separate approaches $70 - 30$ performs the best. The rationale behind this is that there are more computing than RAN resources ($L = 500$ vs. $K_u = 273$). Therefore, the less restricted requirement on the transmission time (higher respective T_{max}) enables admitting more users.

Finally, the three aforementioned "separate" approaches are compared with the newly introduced joint approach for different values of T_{max}, when the amount of data transmitted at once by each user is $\Delta_u = 3$ kbit. For this comparison, users of type 2 are considered. Figure 4.6b illustrates the results. Similar to the previous scenario, the joint approach outperforms the approaches where resources are assigned separately by at least 30 %. Once more, among the separate approaches, the splitting $70 - 30$ performs the best, which can be explained with the same reasons as previously.

4.5 Summary

In this chapter, admission policies for URLLC users with computation demands were developed. Given the setup of the problem, there was the need to jointly consider the uplink

RAN and the edge computing resources, which both influence the overall delay. Admission policies for both homogenous and heterogenous user sets were designed, taking into account traffic parameters and channel conditions. For a homogenous set of users, the maximum number of users that can be admitted in the cell was determined, whereas for a heterogenous set of users the explicit inequality the newly arriving user needs to satisfy was provided, given the set of users (with different channel conditions) that are already being served. Lastly, simulations were run to validate the analytical approaches and to provide interesting insights in the performance of the policies.

5 Scenario 1: Uplink Communication with Edge Processing

In the fifth chapter, allocating RAN and edge computing resources in the first scenario, i.e., a moving user is sending data via a RAN to a BS where this data is processed, is considered. Based on the system model established in Chapter 3, the optimization problem is formulated. Thereby, mathematical expressions and symbols that are needed in the subsequent derivations are introduced. Analyzing the properties of the optimization problem leads to interesting insights regarding its solvability. Finally, a performance evaluation on proposed allocation heuristics shows the availability of good approximation algorithms for the original optimization problem.

The majority of the analyses and results of this chapter will be presented at the 2023 IEEE 20th Consumer Communications & Networking Conference (CCNC) [HMCK22b].

5.1 Optimization Problem Formulation

The overall goal of the RAN and edge computing resource allocation is to maximize the utility over all users after satisfying their traffic requirements, taking into account the finiteness of the available resources. The focus is set on the general case of guaranteeing α-fairness, in the same spirit as the network utility maximization (NUM) approach [Sri04]. Thus, the following optimization formulation can be stated:

$$\max_{\boldsymbol{I}_u, \boldsymbol{m}} \; f^{\alpha}(\boldsymbol{I}_u, \boldsymbol{m}) = \sum_{i=1}^{N} f_i^{\alpha}(\boldsymbol{I}_{u,i},\, m_i) \tag{5.1a}$$

$$\text{s.t.} \quad \frac{\Delta_{u,i}}{\sum\limits_{j=1}^{K_u} I_{u,ij} \Phi_{u,ij}} + \frac{\Delta_{u,i}}{m_i p} \leq T_{max}, \quad \forall i \in \mathcal{U}, \tag{5.1b}$$

$$\sum_{i=1}^{N} m_i \leq L, \tag{5.1c}$$

$$\sum_{i=1}^{N} I_{u,ij} \leq 1, \quad \forall j \in \mathcal{K}_u, \tag{5.1d}$$

$$\sum_{j=1}^{K_u} I_{u,ij} \geq 1, \quad \forall i \in \mathcal{U}, \tag{5.1e}$$

$$I_{u,ij} \in \{0,1\}, \quad \forall i \in \mathcal{U},\, j \in \mathcal{K}_u, \tag{5.1f}$$

$$m_i \in \mathbb{N} \setminus \{0\}, \quad \forall i \in \mathcal{U}, \tag{5.1g}$$

where

$$f_i^{\alpha}(\boldsymbol{I}_{u,i},\, m_i) = \begin{cases} \frac{1}{1-\alpha}\left(\left(\sum_{j=1}^{K_u} I_{u,ij}\Phi_{u,ij}\right)^{1-\alpha} + (m_i p)^{1-\alpha}\right), & \alpha \neq 1 \\ \log\left(\sum_{j=1}^{K_u} I_{u,ij}\Phi_{u,ij}\right) + \log\left(m_i p\right), & \alpha = 1 \end{cases}. \tag{5.2}$$

In the problem formulation, the decision variable $\boldsymbol{I}_u = \{I_{u,ij}\}$ denotes the $N \times K_u$ uplink PRB allocation matrix in a given frame. Namely, if $I_{u,ij} = 1$, then PRB j is assigned to user i in that frame. The $N \times K_u$ matrix $\boldsymbol{\Phi}_u = \{\Phi_{u,ij}\}$ contains the data rates user i would experience when being allocated PRB j. It is derived from the CQI values that are reported for the users. The decision variable $\boldsymbol{m} = \{m_i\}$ is an $N \times 1$ vector consisting of the number of allocated edge computing resources per user i. The amount of information sent by each user at a time is the data size $\Delta_{u,i}$. Lastly, the parameter p denotes the processing rate that one edge computing resource can provide.

The objective (5.1a) maximizes the utility for general $\alpha \in [0, \infty)$. Note that $\alpha = 0$ corresponds to the case of *no fairness* (throughput maximization), $\alpha = 1$ denotes the *proportional fairness* case, $\alpha = 2$ implies *delay minimization*, and $\alpha \to \infty$ describes the *max-min fairness*. Apparently, as there are two types of resources to be allocated, they both affect the value of utility gained. In (5.2), the first term (both for $\alpha \neq 1$ and $\alpha = 1$) corresponds to the utility from assigning RAN resources to user i, while the second term denotes the utility after allocating a number of computing resources.

Constraint (5.1b) describes the maximum tolerable latency for every user. The finite amount of computing resources is captured by (5.1c). Constraint (5.1d) merely states that every block can be assigned to at most one user, whereas (5.1e) stipulates that every user has to receive at least one PRB. Finally, (5.1f) and (5.1g) describe the integer nature of the decision variables, where (5.1g) also implies that at least one computing resource must be assigned to every user.

5.2 Analysis

The structure of the optimization problem described previously belongs to the class of Integer Nonlinear Programs, which are generally known to be NP-hard [LL11]. Therefore, some heuristics are needed to obtain a solution to the aforementioned optimization problem.

The approach that is followed in this work consists of two steps. First, the requirement on the decision variables to be integer is relaxed, i.e., they are now continuous variables. Then, it is shown that under those circumstances the transformed optimization problem is convex and solvable in polynomial time. Finally, in Section 5.3, the second step of the method is described, where special approximation algorithms are developed to obtain integer solutions.

The first step is to show the convex nature of the problem (5.1), when $I_{u,ij} \in [0,1]$ and $m_i \in [1,\infty)$. As the constraints (5.1c)-(5.1g) are linear, they are obviously convex. To show that the objective function is concave, it must be shown that the function $f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i)$ is concave, as the sum of concave functions is a concave function itself.

Lemma 3. *The function $f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i)$ is concave.*

Proof. The gradient of $f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i)$ for $\alpha \neq 1$ is

$$\nabla f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i) = \begin{bmatrix} \Phi_{u,i1}\gamma_{u,i}^{-\alpha} & \dots & \Phi_{u,iK_u}\gamma_{u,i}^{-\alpha} & p(m_i p)^{-\alpha} \end{bmatrix}^T, \tag{5.3}$$

where $\gamma_{u,i} = \sum_{j=1}^{K_u} I_{u,ij}\Phi_{u,ij}$. The gradient of $f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i)$ for $\alpha = 1$ is

$$\nabla f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i) = \begin{bmatrix} \Phi_{u,i1}\gamma_{u,i}^{-1} & \dots & \Phi_{u,iK_u}\gamma_{u,i}^{-1} & m_i^{-1} \end{bmatrix}^T. \tag{5.4}$$

Then, the Hessian matrix of $f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i)$ for $\alpha \neq 1$ is

$$\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i) = -\alpha\gamma_{u,i}^{-\alpha-1} \begin{bmatrix} \Phi_{u,i1}^2 & \dots & \Phi_{u,i1}\Phi_{u,iK_u} & 0 \\ \vdots & \ddots & \vdots & \vdots \\ \Phi_{u,iK_u}\Phi_{u,i1} & \dots & \Phi_{u,iK_u}^2 & 0 \\ 0 & \dots & 0 & \frac{p^2(m_i p)^{-\alpha-1}}{\gamma_{u,i}^{-\alpha-1}} \end{bmatrix}, \tag{5.5}$$

and the Hessian of $f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i)$ for $\alpha = 1$ is

$$\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i) = -\gamma_{u,i}^{-2} \begin{bmatrix} \Phi_{u,i1}^2 & \dots & \Phi_{u,i1}\Phi_{u,iK_u} & 0 \\ \vdots & \ddots & \vdots & \vdots \\ \Phi_{u,iK_u}\Phi_{u,i1} & \dots & \Phi_{u,iK_u}^2 & 0 \\ 0 & \dots & 0 & \gamma_{u,i}^2 m_i^{-2} \end{bmatrix}. \tag{5.6}$$

The characteristic polynomial of $\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i)$ for $\alpha \neq 1$ is

$$\begin{aligned} \det(\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i) - \lambda\mathbb{I}) = (-1)^{K_u-1}\lambda^{K_u-1}\left(\alpha p^2(m_i p)^{-\alpha-1} + \lambda\right) * \\ \left(\alpha\gamma_{u,i}^{-\alpha-1}\Phi_{u,i1}^2 + \dots + \alpha\gamma_{u,i}^{-\alpha-1}\Phi_{u,iK_u}^2 + \lambda\right), \end{aligned} \tag{5.7}$$

while for $\alpha = 1$ it is

$$\det(\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i) - \lambda\mathbb{I}) = (-1)^{K_u-1} * \lambda^{K_u-1}\left(m_i^{-2} + \lambda\right) * \left(\gamma_{u,i}^{-2}\Phi_{u,i1}^2 + \cdots + \gamma_{u,i}^{-2}\Phi_{u,iK_u}^2 + \lambda\right), \quad (5.8)$$

where $\mathbb{I}$ denotes the identity matrix in the corresponding dimension and λ are the eigenvalues of the Hessian $\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i)$. For $\alpha \neq 1$, they can easily be found to be

$$\lambda_1, ..., \lambda_{K_u-1} = 0, \quad (5.9a)$$
$$\lambda_{K_u} = -\alpha\gamma_{u,i}^{-\alpha-1}\left(\Phi_{u,i1}^2 + \cdots + \Phi_{u,iK_u}^2\right), \quad (5.9b)$$
$$\lambda_{K_u+1} = -\alpha p^2 (m_i p)^{-\alpha-1}. \quad (5.9c)$$

The eigenvalues of the Hessian $\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i)$ for $\alpha = 1$ are

$$\lambda_1, ..., \lambda_{K_u-1} = 0, \quad (5.10a)$$
$$\lambda_{K_u} = -\gamma_{u,i}^{-2}(\Phi_{u,i1}^2 + \cdots + \Phi_{u,iK_u}^2), \quad (5.10b)$$
$$\lambda_{K_u+1} = -m_i^{-2}. \quad (5.10c)$$

Since all eigenvalues of the Hessian $\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i)$ (for any α) are smaller than or equal to 0, the Hessian is negative semidefinite, and thus the function $f_i^\alpha(\boldsymbol{I}_{u,i},\, m_i)$ is concave $\forall\alpha$. □

Next, the nature of (5.1b) is explored. It can be stated:

Lemma 4. *Constraint* (5.1b) *is convex.*

Proof. Denote the left-hand side of (5.1b) as

$$t_i(\boldsymbol{I}_{u,i},\, m_i) = \frac{\Delta_{u,i}}{\sum_{j=1}^{K_u} I_{u,ij}\Phi_{u,ij}} + \frac{\Delta_{u,i}}{m_i p} = \frac{\Delta_{u,i}}{\gamma_{u,i}} + \frac{\Delta_{u,i}}{m_i p}. \quad (5.11)$$

The gradient of $t_i(\boldsymbol{I}_{u,i},\, m_i)$ is

$$\nabla t_i(\boldsymbol{I}_{u,i},\, m_i) = \begin{bmatrix} \frac{-\Delta_{u,i}\Phi_{u,i1}}{\gamma_{u,i}^2} & \cdots & \frac{-\Delta_{u,i}\Phi_{u,iK_u}}{\gamma_{u,i}^2} & \frac{-\Delta_{u,i}}{m_i^2 p} \end{bmatrix}^T. \quad (5.12)$$

Then, the Hessian of $t_i(\boldsymbol{I}_{u,i},\, m_i)$ is given as

$$\nabla^2 t_i(\boldsymbol{I}_{u,i},\, m_i) = \frac{2\Delta_{u,i}}{\gamma_{u,i}^3} * \begin{bmatrix} \Phi_{u,i1}^2 & \dots & \Phi_{u,i1}\Phi_{u,iK_u} & 0 \\ \vdots & \ddots & \vdots & \vdots \\ \Phi_{u,iK_u}\Phi_{u,i1} & \dots & \Phi_{u,iK_u}^2 & 0 \\ 0 & \dots & 0 & \frac{\gamma_{u,i}^3}{m_i^3 p}, \end{bmatrix}. \quad (5.13)$$

Computing the determinant of $\nabla^2 t_i(\boldsymbol{I}_{u,i},\, m_i) - \lambda\mathbb{I}$,

$$\det(\nabla^2 t_i(\boldsymbol{I}_{u,i},\, m_i) - \lambda\mathbb{I}) = (-1)^{K_u-1}\lambda^{K_u-1}\left(2\Delta_{u,i}m_i^{-3}p^{-1} - \lambda\right) * \left(2\Delta_{u,i}\gamma_{u,i}^{-3}\Phi_{u,i1}^2 + \cdots + 2\Delta_{u,i}\gamma_{u,i}^{-3}\Phi_{u,iK_u}^2 - \lambda\right) \quad (5.14)$$

is obtained. The eigenvalues of the Hessian $\nabla^2 t_i(\boldsymbol{I}_{u,i},\, m_i)$ are hence

$$\lambda_1, ..., \lambda_{K_u-1} = 0, \quad (5.15a)$$

$$\lambda_{K_u} = 2\Delta_{u,i}\gamma_{u,i}^{-3}\left(\Phi_{u,i1}^2 + \cdots + \Phi_{u,iK_u}^2\right), \quad (5.15b)$$

$$\lambda_{K_u+1} = 2\Delta_{u,i}m_i^{-3}p^{-1}. \quad (5.15c)$$

Since all the eigenvalues of the Hessian $\nabla^2 t_i(\boldsymbol{I}_{u,i},\, m_i)$ are greater than or equal to zero, the Hessian is positive semidefinite and thus the function $t_i(\boldsymbol{I}_{u,i},\, m_i)$ is convex. □

With the proof of the concavity of the objective function and the convexity of the delay function, the following theorem can be stated:

Theorem 5. *The relaxed-variable version of the optimization problem* (5.1) *is convex.*

Proof. Given Lemmas 3 and 4, and the fact that (5.1c)-(5.1g) are linear proves that (5.1) is a convex problem. □

For the purpose of proving the polynomial-time solvability of the relaxed optimization, the problem is reformulated into a convex optimization problem with generalized inequality constraints in the following. For the subsequent derivations, define the n-dimensional quadratic cone as

$$\mathcal{Q}^n = \left\{\boldsymbol{x} \in \mathbb{R}^n \,|\, x_1 \geq \sqrt{x_2^2 + \cdots + x_n^2}\right\}, \quad (5.16)$$

the n-dimensional power cone parameterized by a real number $\zeta \in [0, 1]$ as

$$\mathcal{P}_\zeta^n = \left\{\boldsymbol{x} \in \mathbb{R}^n \,|\, x_1^\zeta x_2^{1-\zeta} \geq \sqrt{x_3^2 + \cdots + x_n^2},\, x_1,\, x_2 \geq 0\right\}, \quad (5.17)$$

and the exponential cone as

$$\mathcal{E} = \left\{\boldsymbol{x} \in \mathbb{R}^3 \,|\, x_1 \geq x_2 e^{x_3/x_2},\, x_1,\, x_2 > 0\right\}. \quad (5.18)$$

First, the relaxed optimization problem is written in epigraph form and the slack variables $s_{ki},\ k \in \{1,2\} = \mathcal{L}_u,\ i \in \mathcal{U}$ are introduced, so that the problem transforms into

$$\min_{g,\boldsymbol{I}_u,\boldsymbol{m},\boldsymbol{s}} \quad g \tag{5.19a}$$

$$\text{s.t.} \quad -\sum_{i=1}^{N} h_i^{\alpha}(s_{1i},\, s_{2i},\, g) \leq 0, \tag{5.19b}$$

$$\frac{\Delta_{u,i}}{s_{1i}} + \frac{\Delta_{u,i}}{s_{2i}} - T_{max} \leq 0, \quad \forall i \in \mathcal{U}, \tag{5.19c}$$

$$\text{(5.1c), (5.1d), (5.1e)}, \tag{5.19d}$$

$$0 \leq I_{u,ij} \leq 1, \quad \forall i \in \mathcal{U},\, j \in \mathcal{K}_u, \tag{5.19e}$$

$$1 - m_i \leq 0, \quad \forall i \in \mathcal{U}, \tag{5.19f}$$

$$s_{1i} = \sum_{j=1}^{K_u} I_{u,ij}\Phi_{u,ij}, \quad \forall i \in \mathcal{U}, \tag{5.19g}$$

$$s_{2i} = m_i p, \quad \forall i \in \mathcal{U}, \tag{5.19h}$$

where

$$h_i^{\alpha}(s_{1i},\, s_{2i},\, g) = \begin{cases} \frac{1}{1-\alpha}\left(s_{1i}^{1-\alpha} + s_{2i}^{1-\alpha}\right) + g, & \alpha \neq 1 \\ \log\left(s_{1i}\right) + \log\left(s_{2i}\right) + g, & \alpha = 1 \end{cases}. \tag{5.20}$$

Next, conic reformulations for the constraints (5.19b) and (5.19c) are introduced.

Lemma 6. *The constraint* (5.19c) *can be written as*

$$\left(s_{1i} + s_{2i} - \frac{\Delta_{u,i}}{T_{max}};\, s_{1i},\, s_{2i},\, \frac{\Delta_{u,i}}{T_{max}}\right) \in \mathcal{Q}^4. \tag{5.21}$$

Proof. By definition, (5.21) transforms into

$$\sqrt{s_{1i}^2 + s_{2i}^2 + \frac{\Delta_{u,i}^2}{{T_{max}}^2}} \leq s_{1i} + s_{2i} - \frac{\Delta_{u,i}}{T_{max}}. \tag{5.22}$$

Squaring both sides and subtracting the expression under the square root on both sides leads to

$$0 \leq 2s_{1i}s_{2i} - 2\frac{\Delta_{u,i}}{T_{max}}(s_{1i} + s_{2i}), \tag{5.23}$$

which is easily transformed into

$$\frac{\Delta_{u,i}}{s_{1i}} + \frac{\Delta_{u,i}}{s_{2i}} - T_{max} \leq 0 \tag{5.24}$$

by dividing by $-2s_{1i}s_{2i}$ and multiplying by T_{max}. □

For the cases $\alpha \in (0,1)$ and $\alpha \in (1,\infty)$, the constraint (5.19b) is converted to the constraints (5.26) by setting $\beta = 1-\alpha$ and introducing the slack variable u_{ki}: Bringing the sum over s_{ki}^{β} to the right side of the inequality results in

$$-g \leq \frac{1}{\beta} \sum_{k=1}^{2} \sum_{i=1}^{N} s_{ki}^{\beta}, \tag{5.25}$$

which is transformed into

$$(5.25) = \begin{cases} -g\beta \leq \sum_{k=1}^{2} \sum_{i=1}^{N} u_{ki}, & (5.26a) \\ u_{ki} \leq s_{ki}^{\beta}, \quad \forall k \in \mathcal{L}_u,\, i \in \mathcal{U};\, \alpha \in (0,1) & (5.26b) \\ g\,|\beta| \geq \sum_{k=1}^{2} \sum_{i=1}^{N} u_{ki}, & (5.26c) \\ u_{ki} \geq s_{ki}^{\beta}, \quad \forall k \in \mathcal{L}_u,\, i \in \mathcal{U};\, \alpha \in (1,\infty) & (5.26d) \end{cases}.$$

Now, the case $\alpha \in (0,1)$, which implies that $\beta \in (0,1)$, is considered.

Lemma 7. *The constraint* (5.26b) *can be written as*

$$(s_{ki},\, 1;\, u_{ki}) \in \mathcal{P}_{\beta}^{3}. \tag{5.27}$$

Proof. By definition, (5.27) is equivalent to

$$s_{ki}^{\beta} 1^{1-\beta} \geq \sqrt{u_{ki}^2},\; s_{ki} \geq 0, \tag{5.28}$$

which simplifies to

$$s_{ki}^{\beta} \geq u_{ki},\; s_{ki} \geq 0. \tag{5.29}$$

The constraint $s_{ki} \geq 0$ that is introduced with this reformulation is fulfilled due to constraints (5.19e) and (5.19f). □

Next, consider $\alpha \in (1,\infty)$, which implies that $\beta \in (-\infty, 0)$.

Lemma 8. *The constraint* (5.26d) *can be written as*

$$(u_{ki},\, s_{ki};\, 1) \in \mathcal{P}_{1/(1-\beta)}^{3}. \tag{5.30}$$

Proof. By definition, (5.30) transforms into

$$u_{ki}^{1/(1-\beta)} s_{ki}^{-\beta/(1-\beta)} \geq \sqrt{1^2},\ u_{ki} \geq 0,\ s_{ki} \geq 0, \tag{5.31}$$

which simplifies to

$$u_{ki} \geq s_{ki}^{\beta},\ u_{ki} \geq 0,\ s_{ki} \geq 0, \tag{5.32}$$

when taking everything to the power of $(1-\beta)$ and multiplying both sides by s_{ki}^{β}. The additional constraints $u_{ki} \geq 0$ and $s_{ki} \geq 0$ that are introduced with this reformulation are met due to the positiveness of s_{ki} implied by (5.19e) and (5.19f). □

Lastly, consider the case of $\alpha = 1$. Then, the constraint (5.19b) must be rewritten to

$$-g \leq \sum_{k=1}^{2} \sum_{i=1}^{N} \log s_{ki} \tag{5.33}$$

by bringing the sum over the logarithms to the right side of the inequality. Using again the slack variable u_{ki}, (5.33) can be formulated as

$$-g \leq \sum_{k=1}^{2} \sum_{i=1}^{N} u_{ki}, \tag{5.34a}$$

$$u_{ki} \leq \log s_{ki}, \quad \forall k \in \mathcal{L}_u,\ i \in \mathcal{U}. \tag{5.34b}$$

Lemma 9. *Constraint* (5.34b) *can be rewritten as*

$$(s_{ki},\ 1,\ u_{ki}) \in \mathcal{E}. \tag{5.35}$$

Proof. By definition, (5.35) is equivalent to

$$s_{ki} \geq 1 * e^{u_{ki}/1},\ s_{ki} > 0, \tag{5.36}$$

which can be written as

$$\log s_{ki} \geq u_{ki},\ s_{ki} > 0, \tag{5.37}$$

when taking the logarithm of both sides. The additional constraint $s_{ki} > 0$ that is introduced with this reformulation is fulfilled due to the constraints (5.1e) and (5.19f). □

Theorem 10. *The relaxed-variable version of the optimization problem* (5.1) *can be written as a convex optimization problem with generalized inequality constraints.*

Proof. Given Lemmas 6, 7, 8, and 9 and the fact that (5.19b) is linear for $\alpha = 0$ concludes the proof. □

The optimization problem (5.1) reads in a relaxed form, written as a convex optimization problem with generalized inequality constraints, for any $\alpha \in [0, \infty)$:

$$\min_{g, \boldsymbol{I}_u, \boldsymbol{m}, \boldsymbol{s}, \boldsymbol{u}} \quad g \tag{5.38a}$$

$$\text{s.t.} \quad -\sum_{i=1}^{N} e_i^{\alpha}(s_{1i},\, s_{2i},\, u_{1i},\, u_{2i},\, g) \leq 0, \tag{5.38b}$$

$$(5.1c),\ (5.1d),\ (5.1e),\ (5.19e),\ (5.19f),\ (5.19g),\ (5.19h),\ (5.21), \tag{5.38c}$$

where

$$(5.38b) = \begin{cases} (5.19b), & & \alpha = 0 \\ (5.26a), (5.27), & \forall k \in \mathcal{L}_u,\, i \in \mathcal{U}, & 0 < \alpha < 1 \\ (5.34a), (5.35), & \forall k \in \mathcal{L}_u,\, i \in \mathcal{U}, & \alpha = 1 \\ (5.26c), (5.30), & \forall k \in \mathcal{L}_u,\, i \in \mathcal{U}, & \alpha > 1 \end{cases}. \tag{5.39}$$

For the final verification of the polynomial-time solvability of the optimization problem stated in (5.38), define the following generalized logarithms and note their degrees. More details on the generalized logarithm can be found in Section 11.6 in [BV04]. The generalized logarithm for the n-dimensional quadratic cone $\mathcal{Q}^n$ can be designed as [BV04]

$$\Gamma_{\mathcal{Q}}(\boldsymbol{x}) = \log\left(x_1^2 - \sum_{i=2}^{n} x_i^2\right). \tag{5.40}$$

The degree of a generalized logarithm is calculated as $\theta_\Gamma = \nabla\Gamma(\boldsymbol{x})^T\boldsymbol{x}$, cf. [BV04]. The degree of the function $\Gamma_{\mathcal{Q}}(\boldsymbol{x})$ is therefore

$$\theta_{\mathcal{Q}} = \nabla\Gamma_{\mathcal{Q}}(\boldsymbol{x})^T\boldsymbol{x} = \left[\frac{2x_1}{\left(x_1^2 - \sum_{i=2}^{n} x_i^2\right)} \quad \cdots \quad \frac{-2x_n}{\left(x_1^2 - \sum_{i=2}^{n} x_i^2\right)}\right]\boldsymbol{x} = 2. \tag{5.41}$$

Additionally, define the generalized logarithm for the n-dimensional power cone $\mathcal{P}_\zeta^n$ as

$$\Gamma_{\mathcal{P}}(\boldsymbol{x}) = \log\left(x_1^{2\zeta}x_2^{(2-2\zeta)} - \sum_{i=3}^{n} x_i^2\right) + (1-\zeta)\log(x_1) + \zeta\log(x_2), \tag{5.42}$$

as introduced in [Cha09]. The degree of the function $\Gamma_{\mathcal{P}}(\boldsymbol{x})$ is calculated as

$$\begin{aligned}\theta_{\mathcal{P}} = \nabla\Gamma_{\mathcal{P}}(\boldsymbol{x})^T\boldsymbol{x} = \left[\frac{2\zeta x_1^{(2\zeta-1)}x_2^{(2-2\zeta)}}{a} + \frac{1-\zeta}{x_1} \quad \frac{(2-2\zeta)x_1^{2\zeta}x_2^{(1-2\zeta)}}{a} + \frac{\zeta}{x_2} \quad \frac{-2x_3}{a} \quad \ldots \quad \frac{-2x_n}{a}\right]\boldsymbol{x} = \\ = \frac{2(\zeta+1-\zeta)x_1^{2\zeta}x_2^{(2-2\zeta)} - 2\sum_{i=3}^{n} x_i^2}{a} + 1 - \zeta + \zeta = 3,\end{aligned} \tag{5.43}$$

where $a = x_1^{2\zeta} x_2^{(2-2\zeta)} - \sum_{i=3}^{n} x_i^2$. Finally, define the generalized logarithm for the exponential cone $\mathcal{E}$ as [Cha09]

$$\Gamma_{\mathcal{E}}(\boldsymbol{x}) = \log\left(x_2 \log\left(\frac{x_1}{x_2}\right) - x_3\right) + \log x_1 + \log x_2. \tag{5.44}$$

The degree of the generalized logarithm $\Gamma_{\mathcal{E}}(\boldsymbol{x})$ is also determined as

$$\begin{aligned}\theta_{\mathcal{E}} = \nabla\Gamma_{\mathcal{E}}(\boldsymbol{x})^T \boldsymbol{x} = \left[\frac{x_2/x_1}{x_2\log(x_1/x_2)-x_3} + \frac{1}{x_1} \quad \frac{\log(x_1/x_2)-1}{x_2\log(x_1/x_2)-x_3} + \frac{1}{x_2} \quad \frac{-1}{x_2\log(x_1/x_2)-x_3}\right]\boldsymbol{x} = \\ = \frac{x_2 + x_2\log(x_1/x_2) - x_2 - x_3}{x_2\log(x_1/x_2) - x_3} + 2 = 3.\end{aligned} \tag{5.45}$$

Lastly, a slack variable that is attached to the system of equality constraints can be inserted for every linear inequality constraint of the optimization. The corresponding generalized logarithm for these slack variables has degree 1, as the slack variable needs to be in $\mathbb{R}_+$. Using these definitions of the generalized logarithms, a logarithmic barrier function $\Lambda_u(\boldsymbol{w}_u)$ can be defined as

$$\Lambda_u(\boldsymbol{w}_u) = -\sum_{c=1}^{Z} \Gamma_c(\boldsymbol{w}_u),\ \mathbf{dom}\,\Lambda = \{\boldsymbol{w}_u \mid f_c(\boldsymbol{w}_u) \prec_{K_c} 0,\ c = 1, ..., Z\}, \tag{5.46}$$

where $Z = (3 + 2K_u)N + 2 + K_u$ for $\alpha = 0$ and $Z = (5 + 2K_u)N + 2 + K_u$ for $\alpha \neq 0$. $\boldsymbol{w}_u$ is composed of the vectorized matrix $\boldsymbol{I}_u$ as well as the vectors $\boldsymbol{m}$, $\boldsymbol{s} = \{s_{ki}\}$, and $\boldsymbol{u} = \{u_{ki}\}$. $\Gamma_c(\boldsymbol{w}_u)$ are the generalized logarithms defined above for each generalized inequality constraint $f_c(\boldsymbol{w}_u)$ in the convex optimization problem with generalized inequalities defined in (5.38). This implies that the barrier method can be applied in order to solve this optimization problem.

The subsequent complexity analysis is based on the property of self-concordance. For a definition of self-concordance, see Section 9.6 in [BV04].

Lemma 11. *The logarithmic barrier function $\Lambda_u(\boldsymbol{w}_u)$ is self-concordant.*

Proof. The logarithmic barrier for the positive orthant defined by all linear inequalities and their slack variables is a self-concordant function because $-\log x$ is self-concordant and the sum of self-concordant functions is again self-concordant [BV04]. The logarithmic barriers established using the generalized logarithms defined in (5.40), (5.42), and (5.44) are self-concordant as well, see Section 11.6 in [BV04] and Sections 2.4 and 3.1 in [Cha09]. □

Lemma 12. *The number of total Newton steps excluding the initial centering step for solving* (5.38) *using the Barrier method can be bounded by [BV04]*

$$T_{Barrier} = \left\lceil \frac{\log(\bar{\theta}/(t^{(0)}\xi))}{\log\mu} \right\rceil * \left(\frac{\bar{\theta}(\mu - 1 - \log\mu)}{\chi} + \log_2\log_2(1/\xi)\right). \tag{5.47}$$

Proof. Given Lemma 11 and the fact that (5.38a) is linear, the function $tg + \Lambda_u(\boldsymbol{w}_u)$, which is the objective of the Barrier method, is self-concordant. Additionally, given that this function is closed and the sublevel sets of the optimization problem (5.38) are bounded leads to (5.47), cf. [BV04]. □

The parameter $\mu > 1$ is an algorithm parameter of the barrier method, $t^{(0)} > 0$ is the initial value of the algorithm parameter t of the barrier method, and $\xi > 0$ is the specified tolerance of the barrier method, see Algorithm 11.1 in [BV04]. The parameter χ is a constant that depends on the backtracking parameters κ and τ, Alg. 9.2 in [BV04], which is used for line search in Newton's method. It is given as

$$\frac{1}{\chi} = \frac{20 - 8\kappa}{\kappa\tau(1-2\kappa)^2}. \tag{5.48}$$

The last parameter $\bar{\theta}$ is the sum of the degrees of the generalized logarithms Γ_c, which for the considered problem is computed as

$$\bar{\theta} = \begin{cases} (4 + 2K_u)N + 2 + K_u, & \alpha = 0 \\ (10 + 2K_u)N + 2 + K_u, & \alpha \neq 0 \end{cases}. \tag{5.49}$$

Finally, the following theorem can be stated:

Theorem 13. *The complexity of solving the optimization problem* (5.38) *in terms of Newton steps is*

$$T_{Barrier} = \mathcal{O}\left(\log\left(K_u N/\xi\right)\left(K_u N + \log_2\log_2\left(1/\xi\right)\right)\right). \tag{5.50}$$

Proof. Plugging (5.49) into (5.47) and simplifying this expression yields (5.50). □

5.3 Conversion Algorithms

In the previous section, it was shown that an optimal solution to the relaxed optimization problem can be found in polynomial time. However, this is a continuous solution, which violates the natural restriction that only integer fractions of RAN and edge computing resources can be allocated. Therefore, for the particular values of $\alpha = 0$, $\alpha = 1$, $\alpha = 2$, and $\alpha \to \infty$, specific algorithms for the conversion of the continuous solution to an integer resource allocation are developed. First, the conversion algorithm for the edge computing resource allocation is introduced, which is used for all fairness cases. Afterwards, the algorithms for the specific cases of α are explained. Thereby, $\boldsymbol{J}_u$ denotes the $N \times K_u$ RAN allocation matrix with entries $J_{u,ij} \in \{0, 1\}$ and $\boldsymbol{n}$ is the $N \times 1$ edge computing resource allocation vector with entries $n_i \in \mathbb{N} \setminus \{0\}$. The variables $\boldsymbol{I}_u$ and $\boldsymbol{m}$ are their continuous equivalents. For the conversion algorithms, it is assumed that the admission

control for a heterogenous set of users presented in Section 4.3 was performed before accepting users to the network. In case there are users sending packets with a different overall data size, i.e., $\Delta_{u,i} \neq \Delta_{u,k}$, the unified data size Δ_u in the admission policy can be set to $\Delta_u = \max_i \Delta_{u,i}$ to ensure the availability of enough resources. Although the admission control for a heterogenous set of users only decides whether a new user can be accepted to the network, this is not an applicability restriction, since every network starts with zero users at some point.

5.3.1 Edge Computing Resources

The conversion of the continuous edge computing resource allocation to an integer allocation is done by simple mathematical rounding. As this procedure can lead to the assignment of more than L edge computing resources, a limit check is conducted after the rounding. If more than L edge computing resources are allocated, then the user with a continuous allocation value closest above $\star.5$, where $\star$ denotes an arbitrary integer, is assigned one edge computing resource less than it would have received by strict mathematical rounding. This is done until L edge computing resources are allocated. Similarly, if less than L resources are allocated, the users closest below $\star.5$ will receive one more resource until L resources are assigned. The described procedure is summarized in Algorithm 2. Its complexity is $\mathcal{O}(N)$, i.e., it is linear.

5.3.2 No Fairness (Throughput Maximization)

If all constraints were neglected, the case $\alpha = 0$ would lead to an allocation where each PRB j is allocated to the user who is experiencing the best channel conditions, i.e., the user with the highest CQI value for that PRB. The allocation of the edge computing resources could be done randomly, as each edge computing resource offers the same processing rate and thus contributes in the same way to the objective no matter to which user the resource is assigned. However, each user's data must be sent and processed within a given time. Therefore, the edge computing resources are allocated such that users with worse channel conditions and larger packets get more edge computing resources in order to minimize the number of required PRB allocations for that user, as allocations of PRBs to users with lower CQI values have a negative impact on the maximization of the objective. Once all users are assigned enough edge computing and RAN resources to fulfill their delay constraints, the remaining PRBs are allocated to the users experiencing the best channel conditions.

The approximation algorithm that was designed using these intuitions can be explained as follows: First, all users are allocated enough edge computing and RAN resources such that they can fulfill their delay constraints. This is done using the continuous allocations $\boldsymbol{I}_u$ and $\boldsymbol{m}$. Afterwards, the remaining PRBs are assigned to the users experiencing the best channel conditions. Note that for solving the continuous optimization problem only

Algorithm 2 Edge Computing Resource Allocation

Input: N, L, $\boldsymbol{m}$
Output: $\boldsymbol{n}$

function ECRAlloc(N, L, $\boldsymbol{m}$)
  for all m_i **do**
    $n_i = \lfloor m_i + 0.5 \rfloor$
  end for
  Create empty lists w and z.
  while $\sum_{i=1}^{N} n_i > L$ **do**
    $l = 1$, $k = 0$
    for $i = 1$ to N **do**
      if $i \notin w$ **then**
        $r_i = m_i \mod \lfloor m_i \rfloor - 0.5$
        if $0 < r_i < l$ **then**
          $l = r_i$, $k = i$
        end if
      end if
    end for
    $n_k = \lfloor m_k \rfloor$, attach k to list w.
  end while
  while $\sum_{i=1}^{N} n_i < N$ **do**
    $a = -1$, $b = 0$
    for $i = 1$ to N **do**
      if $i \notin z$ **then**
        $r_i = m_i \mod \lfloor m_i \rfloor - 0.5$
        if $a < r_i < 0$ **then**
          $a = r_i$, $b = i$
        end if
      end if
    end for
    $n_b = \lceil m_b \rceil$, attach b to list z.
  end while
  return $\boldsymbol{n}$
end function

$L - N$ edge computing resources are used. The reason for this procedure is that the integer resource allocation per user is then at least as high as the continuous allocation, ensuring the feasibility of the integer solution attained from the heuristic. Algorithm 3 summarizes the previous explanations. Its complexity is $\mathcal{O}(N + K_u)$.

Algorithm 3 Resource Allocation for $\alpha = 0$ in the Uplink-Only Scenario

Input: N, K_u, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{\Phi}_u$
Output: $\boldsymbol{n}$, $\boldsymbol{J}_u$

function AllocA0(N, K_u, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{\Phi}_u$)
  $\boldsymbol{n} = \text{ECRAlloc}(N, L - N, \boldsymbol{m}) + \mathbf{1}$, $\boldsymbol{J}_u = 0$.
  for $i = 1$ to N **do**
    Calculate $w_i = \sum_{j=1}^{K_u} I_{u,ij}\Phi_{u,ij}$.
  end for
  Create list z with users i ordered s.t. $\Delta_{u,i}/w_i$ is decreasing.
  while list z is non-empty **do**
    for user i in list z **do**
      Find $\arg\max_j I_{u,ij}\Phi_{u,ij}$.
      if $\exists$ more than one j maximizing $I_{u,ij}\Phi_{u,ij}$ **then**
        Choose randomly between those j.
      end if
      Allocate PRB j to user i, update $\boldsymbol{J}_{u,j}$ and set $\boldsymbol{I}_{u,j} = \mathbf{0}$.
      Calculate delay δ_i using n_i and $\boldsymbol{J}_{u,i}$.
      if $\delta_i \leq T_{max}$ **then**
        Remove user i from list z.
      end if
    end for
  end while
  for all non-allocated PRBs k **do**
    Find $\arg\max_i \Phi_{u,ik}$, then allocate PRB k to user i and update $\boldsymbol{J}_{u,k}$.
  end for
  return $\boldsymbol{n}$, $\boldsymbol{J}_u$
end function

5.3.3 Proportional Fairness

In the proportional fairness case, every user gets the same amount of resources, independent of the channel conditions it is experiencing. Mathematically, this can be explained as follows: for $\alpha = 1$, the pure objective is to maximize the sum of the natural logarithms of the RAN data and edge processing rates. The natural logarithm is characterized by the fact that its output value always increases by the same amount when the argument of the logarithm doubles, independent of the absolute value of the input argument. This implies that the objective value increases by the same amount irrespective of whether a user with bad channel conditions can double its allocated resources or a user who experiences good channel conditions can double its resources. It also means that every user should be assigned the same amount of resources, as the objective value would decrease if a user with good channel conditions has four times as many assigned PRBs than a user with bad

Algorithm 4 Resource Allocation for $\alpha = 1$ in the Uplink-Only Scenario

Input: N, K_u, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{\Phi}_u$
Output: $\boldsymbol{n}$, $\boldsymbol{J}_u$

1: **function** AllocA1(N, K_u, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{\Phi}_u$)
2: Follow lines 2 to 19 from Algorithm 3.
3: **for** $i = 1$ to N **do**
4: Calculate $w_i = \sum_{j=1}^{K_u} J_{u,ij}$.
5: **end for**
6: Create list z with users i ordered s.t. w_i is increasing.
7: **for all** non-allocated PRBs k **do**
8: Take $z(1)$, find $\arg\min_k \left(\max_i (\Phi_{u,ik}) - \Phi_{u,z(1)k} \right)$.
9: Allocate PRB k to user $z(1)$ and update $\boldsymbol{J}_{u,k}$.
10: Set $w_{z(1)} = \sum_{j=1}^{K_u} J_{u,z(1)j}$.
11: Reorder list z with users i s.t. w_i is increasing.
12: **end for**
13: **return** $\boldsymbol{n}$, $\boldsymbol{J}_u$
14: **end function**

channel conditions, as the growth of the natural logarithm is larger the smaller the input arguments are.

These insights lead to the design of Algorithm 4. Thereby, again first the required resources for fulfilling the delay constraint are allocated, and then the proportional fairness aim, i.e., giving every user the same number of resources, is followed when assigning the remaining PRBs. The complexity of Algorithm 4 can be given as $\mathcal{O}(N + K_u)$.

5.3.4 Delay Minimization

For the scenario $\alpha = 2$, the prefactor in the objective (5.2) turns into -1, transforming the maximization problem into a minimization problem. Moreover, the exponent of the RAN data rate and the edge processing rate of each user turns into -1 as well, leading to the minimization of the reciprocals of these rates. Comparing this objective function with the left-hand side of the delay constraint (5.1b), it is observable that the two functions are the same, with the only difference being the missing data size $\Delta_{u,i}$ in the objective function. If the data sizes are equal for all users, e.g., they all request the same type of service, then the contemplated optimization problem is perfectly equal to delay minimization, as the data sizes are just a constant not influencing the optimization. In contrast, diverse data sizes across users lead to a suboptimal delay minimization, as the delay is influenced by the various data sizes. However, since the data sizes are all of the same order, the impact on the optimization is marginal.

Algorithm 5 Resource Allocation for $\alpha = 2$ in the Uplink-Only Scenario

Input: N, K_u, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{\Phi}_u$
Output: $\boldsymbol{n}$, $\boldsymbol{J}_u$

1: **function** AllocA2(N, K_u, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{\Phi}_u$)
2: Follow lines 2 to 19 from Algorithm 3.
3: **for** $i = 1$ to N **do**
4: Calculate $w_i = \sum_{j=1}^{K_u} J_{u,ij}\Phi_{u,ij}$.
5: **end for**
6: Create list z with users i ordered s.t. w_i is increasing.
7: **for all** non-allocated PRBs k **do**
8: Take $z(1)$, find $\arg\min_k \left(\max_i (\Phi_{u,ik}) - \Phi_{u,z(1)k}\right)$.
9: Allocate PRB k to user $z(1)$ and update $\boldsymbol{J}_{u,k}$.
10: Set $w_{z(1)} = \sum_{j=1}^{K_u} J_{u,z(1)j}\Phi_{u,z(1)j}$.
11: Reorder list z with users i s.t. w_i is increasing.
12: **end for**
13: **return** $\boldsymbol{n}$, $\boldsymbol{J}_u$
14: **end function**

In order to minimize the overall system delay, the knowledge of all RAN assignment combinations is needed. Since this knowledge is not existent when performing the approximation, the focus of the developed heuristic is to minimize the maximum experienced delay of any user. The pursued scheme equals the strategy from Algorithms 3 and 4. First, the resources for meeting the latency requirements are allocated using the continuous allocation values. Then, the remaining resources are distributed according to the fairness metric. The final algorithm is detailed specified in Algorithm 5 with its complexity being $\mathcal{O}(N + K_u)$.

5.3.5 Max-Min Fairness

When $\alpha \to \infty$, the pure objective is characterized as the minimization of the sum of the reciprocals of the data and processing rates raised to a large positive number. This minimization is achieved once the users' data and processing rates are equal to each other. Hence, the computing resources are split equally among the users and the PRBs are allocated such that the difference between the users' data rates is minimized while maximizing the minimum data rate any user experiences. This allocation scheme is also pursued if a delay constraint is introduced. However, the computing resources and PRBs are then allocated such that the delay constraint is fulfilled for all users, which implies that the differences between the users' data rates might increase and the minimum data rate achieved by any user might decrease. The same type of redistribution is also done for the edge resource assignment, if needed. In general, this implies that the attained objective value is in most cases smaller than the objective value attained when neglecting the delay constraint.

Algorithm 6 Resource Allocation for $\alpha \to \infty$ in the Uplink-Only Scenario

Input: N, K_u, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{\Phi}_u$
Output: $\boldsymbol{n}$, $\boldsymbol{J}_u$

1: **function** AllocAInf(N, K_u, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{\Phi}_u$)
2: Follow lines 2 to 19 from Algorithm 3.
3: **for** $i = 1$ to N **do**
4: Calculate $w_i = \left(\sum_{j=1}^{K_u} J_{u,ij}\Phi_{u,ij}\right)^{|1-\alpha|}$.
5: **end for**
6: Create list z with users i ordered s.t. w_i is increasing.
7: **for all** non-allocated PRBs k **do**
8: Take $z(1)$, find $\arg\min_k \left(\max_i (\Phi_{u,ik}) - \Phi_{u,z(1)k}\right)$.
9: Allocate PRB k to user $z(1)$ and update $\boldsymbol{J}_{u,k}$.
10: Set $w_{z(1)} = \left(\sum_{j=1}^{K_u} J_{u,z(1)j}\Phi_{u,z(1)j}\right)^{|1-\alpha|}$.
11: Reorder list z with users i s.t. w_i is increasing.
12: **end for**
13: **return** $\boldsymbol{n}$, $\boldsymbol{J}_u$
14: **end function**

The last heuristic adjusted for the case $\alpha \to \infty$ again follows the same concept as the approximation algorithms for the other special fairness cases. It is summarized in Algorithm 6. Its computational complexity is determined as $\mathcal{O}(N + K_u)$.

5.4 Performance Evaluation

In the penultimate section of this chapter, the performances of the proposed allocation heuristics are evaluated. To this end, first, the simulation setup is described. Next, a benchmark algorithm is introduced. Finally, the remaining sections address the performance of the approximation algorithms for the cases $\alpha = 0$, $\alpha = 1$, $\alpha = 2$, and $\alpha \to \infty$.

5.4.1 Simulation Setup

As input parameters, once more the 5G trace with data measured in the Republic of Ireland [RLSQ20] was used. The parameter of interest from the traces is again the CQI, from which one per-block rate of a user in a frame can be determined. The corresponding data rates per CQI are given in Table 4.1. Since there is only one CQI value given per time step in each measurement, the per-PRB CQI values were derived from the measured CQI value by generating a population of CQI values in $\{\text{CQI}-1, \text{CQI}, \text{CQI}+1\}$, with its mean being the measured CQI and the frequency of each value being 1/3. In case the CQI value

was 1 or 15, no population was generated since CQI values of 0 and 16 are impossible and hence the population's mean would not have been equal to the measured CQI. Different measurements were again taken to mimic different users.

The subcarrier spacing is 30 kHz, making the PRB width 360 kHz and a frame consisting of 20 slots [ETS22b]. The total number of PRBs is $K_u = 120$. At least 1 slot and at most 6, 10, or 20 slots are allocated to each user, which follows from the latency requirement and the duration of one slot (0.5 ms). The number of edge computing resources is $L = 120$ and the processing rate per resource is 500 kbps. The data size of the packets is 5 kbit for every user. For all types of fairness, simulation data were gathered for $N = \{5, 8, 10\}$ and for $T_{max} = \{3, 5, 10\}$ ms. The simulations were conducted in MATLAB R2021b. To solve the optimization problems CVX [GB14, GB08] together with the Mosek optimizer was used [MOS22].

5.4.2 Benchmark (Round-Robin)

The benchmark allocation against which the special approximation algorithms are compared is the Round-Robin principle. This means that all users are allocated one computing resource and one PRB in each iteration. Once a user fulfills its delay constraint, it will not be assigned any more resources until every user complies with its latency target. Thereafter, the remaining computing and RAN resources are allocated one by one to all users, until no resources are available anymore. The described method is summarized in Algorithm 7.

5.4.3 Results for No Fairness (Throughput Maximization)

Various measurement points for different CQI inputs are depicted in Figure 5.1. The value RG that is given in the title of the subfigures of Figure 5.1 and also in all subsequent evaluation plots denotes the relative gap that was set when solving the original integer problem using Mosek. The relative gap is mathematically calculated as [MOS22]

$$RG = \frac{|\bar{z} - \underline{z}|}{\max\left(10^{-10}, |\bar{z}|\right)}, \tag{5.51}$$

where $\underline{z}$ denotes the solution to the relaxed optimization problem and $\bar{z}$ denotes the best feasible integer solution. The relative gap guarantees that the best feasible integer solution is at most $RG * 10^2$ % away from the true optimum. The first three rows in Figure 5.1 show all setup combinations including the objective values from the benchmark. For a better comparability, the last three rows show the same results but without the values obtained from the Round-Robin principle. It is observable that the approximation algorithm outperforms the benchmark by far and is very close to the integer and continuous optimum. Of course, the continuous optimum is always better than or equal to the integer

Algorithm 7 Round-Robin Allocation in the Uplink-Only Scenario

Input: N, K_u, L
Output: $\boldsymbol{n}$, $\boldsymbol{J}_u$

```
1: function RRAlloc(N, K_u, L)
2:     Create list z with users i.
3:     while list z is non-empty do
4:         for user i in list z do
5:             Allocate one edge computing resource and one PRB to user i.
6:             Update n_i and J_{u,i}.
7:             Calculate delay δ_i using n_i and J_{u,i}.
8:             if δ_i ≤ T_max then
9:                 Remove user i from list z.
10:            end if
11:        end for
12:    end while
13:    while ∃ non-allocated edge computing or RAN resource(s) do
14:        for i = 1 to N do
15:            if ∃ non-allocated edge computing resource then
16:                Allocate one edge computing resource to user i.
17:            end if
18:            if ∃ non-allocated RAN resource then
19:                Allocate one PRB to user i.
20:            end if
21:            Update n_i and J_{u,i}.
22:        end for
23:    end while
24:    return n, J_u
25: end function
```

optimum, as in this case fractions of resources are assigned, which is not possible in reality. In Figure 5.2 the deviation of the heuristic from the integer and continuous optimum is shown in percent. Due to the NP-hardness of the integer problem, it was sometimes not possible to obtain results attaining an adequate accuracy, which is the reason why some data points from the approximation algorithm "outperform" the integer optimum, i.e., the percentual deviation is negative. When comparing the solution obtained from the heuristic to the continuous optimum among 100 data points, the maximum deviation that can be observed is 1 %, whereas the average deviation is 0.24 %. This indicates the very good performance of the approximation algorithm. The evaluation is supported by Figure 5.3, where the average objective value from the heuristic is very close or equal to the average continuous optimum of 100 measurement points. Another observation from Figure 5.3 is that the average objective value slightly increases when loosening the delay constraint. The reason for this behavior is that more PRBs can be allocated to users experiencing the

best channel conditions, as in general a user does not need that many resources to fulfill its delay constraint when the maximum allowed latency T_{max} is increased. Finally, it can again be seen that the benchmark is surpassed.

5.4.4 Results for Proportional Fairness

The same plots as for the throughput maximization are shown for the case $\alpha = 1$, i.e., proportional fairness, in Figures 5.4 to 5.6. The benchmark algorithm is again outperformed by the special approximation algorithm, but this time the benchmark objective values are much closer to the integer optimum than for the case $\alpha = 0$. This is especially observable when comparing the average values in Figure 5.6. The reason for this behavior is that in the proportional fairness case all users are allocated the same amount of resources. This is very similar to the Round-Robin principle, in which all the users are assigned one resource after each other until no resources are available anymore. Because users get assigned the resources where the difference between the data rate they experience and the maximum experienced data rate of any user for a PRB is the smallest, the approximation algorithm for proportional fairness still outperforms the benchmark heuristic. Furthermore, in Figure 5.6 it can be seen that the objective value does not depend on the delay constraint, i.e., T_{max}, but it increases with the number of users in the network. The reason for this behavior is the large gradient of the natural logarithm for small input arguments, such that every added user has a large impact on the overall objective. In Figure 5.5 it is observable that the deviation of the results from the approximation algorithm from the integer optimum is again very small. The maximum deviation that can be detected among 100 data points is 0.14 %, while the average deviation is 0.03 %. These examinations prove the excellent performance of the presented approximation algorithm.

5.4.5 Results for Delay Minimization

The results for the scenario $\alpha = 2$ are depicted in Figures 5.7 to 5.9. It is discernible that the special approximation algorithm surpasses the benchmark heuristic, as it happened for $\alpha = 0$ and $\alpha = 1$. In most of the cases, the objective value obtained with the approximation algorithm is very close to the integer optimum, with the average deviation among 100 points being 0.91 %. However, for the measurements 15 and 16, two outliers are detectable. The reason for these outliers is the presence of a user who is experiencing bad channel conditions compared to the other users. Due to the slightly changed objective of the approximation algorithm, there are certainly resource allocations possible where the objective value can be maximized compared to the minimization of the maximum encountered delay, in case there are users whose data rates are a lot worse than all other users' data rates. The cost for this objective maximization is an increased delay experienced by the user with bad channel conditions. Despite the aforementioned drawback, it can be concluded that the performance of the approximation algorithm for delay minimization is still very good,

as the maximum observed deviation from the integer optimum is 7.11 %. Besides, in Figure 5.9, an inverse dependence of the objective value on the number of users can be detected compared to $\alpha = 1$. The less users are present in the system, the higher is the objective value. This is an apparent behavior as the overall delay is of course smaller the less users there are.

5.4.6 Results for Max-Min Fairness

For the max-min fairness, a high α-value, i.e., $\alpha = 13$, was used to mimic the behavior of $\alpha \to \infty$. Due to numerical reasons during the optimization, a higher α-value could not be used. A similar output as for the other fairness cases can be contemplated for the max-min fairness in Figures 5.10 to 5.12. The benchmark heuristic is again outperformed by the approximation algorithm, especially in the presence of a user with bad channel conditions (measurements 15 and 16). In Figure 5.12 it can be observed that the objective value gets worse the higher the number of users is, because then the resources have to be split among more users and the reciprocals of the RAN data and the edge processing rates get larger. The benchmark averages are largely influenced by some outliers, where few users are experiencing very bad channel conditions, which is the reason for the random appearance of the average bars in Figure 5.12. Due to the NP-hardness of the integer optimization problem, the quality of the integer optimal values is not good enough for a comparison. However, the objective values obtained with the heuristic are almost attaining the continuous optimum (not more than 0.0617 % away), which indicates that the continuous optimum is almost an integer optimum. The average deviation from the continuos optimum among 100 data points is only 0.0006 %, which certifies the exceptional performance of the heuristic.

5.5 Summary

In this chapter, the problem of jointly allocating RAN and processing resources to vehicular users so that their latency requirement is met, while simultaneously providing certain types of fairness, was considered. The contemplated scenario was an uplink-only scenario, meaning that the users sent data to the base station, where the data was processed, but no response was generated. It was shown that the integer-relaxed allocation optimization problem is solvable in polynomial time, and approximation algorithms with polynomial-time complexity were provided for the cases no fairness, proportional fairness, delay minimization, and max-min fairness. For each fairness scenario, the performance of the approximation algorithm is very close to the optimum. The key results regarding the deviation of the objective values from the optimal values are summarized in Table 5.1.

Table 5.1: Maximum and average deviation of the approximation algorithm objective values from the continuous/integer optimum among 100 data points in the uplink-only scenario

α	0	1	2	13
max. dev. from int. opt. in %	—	0.14	7.11	—
avg. dev. from int. opt. in %	—	0.03	0.91	—
max. dev. from cont. opt. in %	1.00	0.14	7.11	0.0617
avg. dev. from cont. opt. in %	0.24	0.03	0.93	0.0006

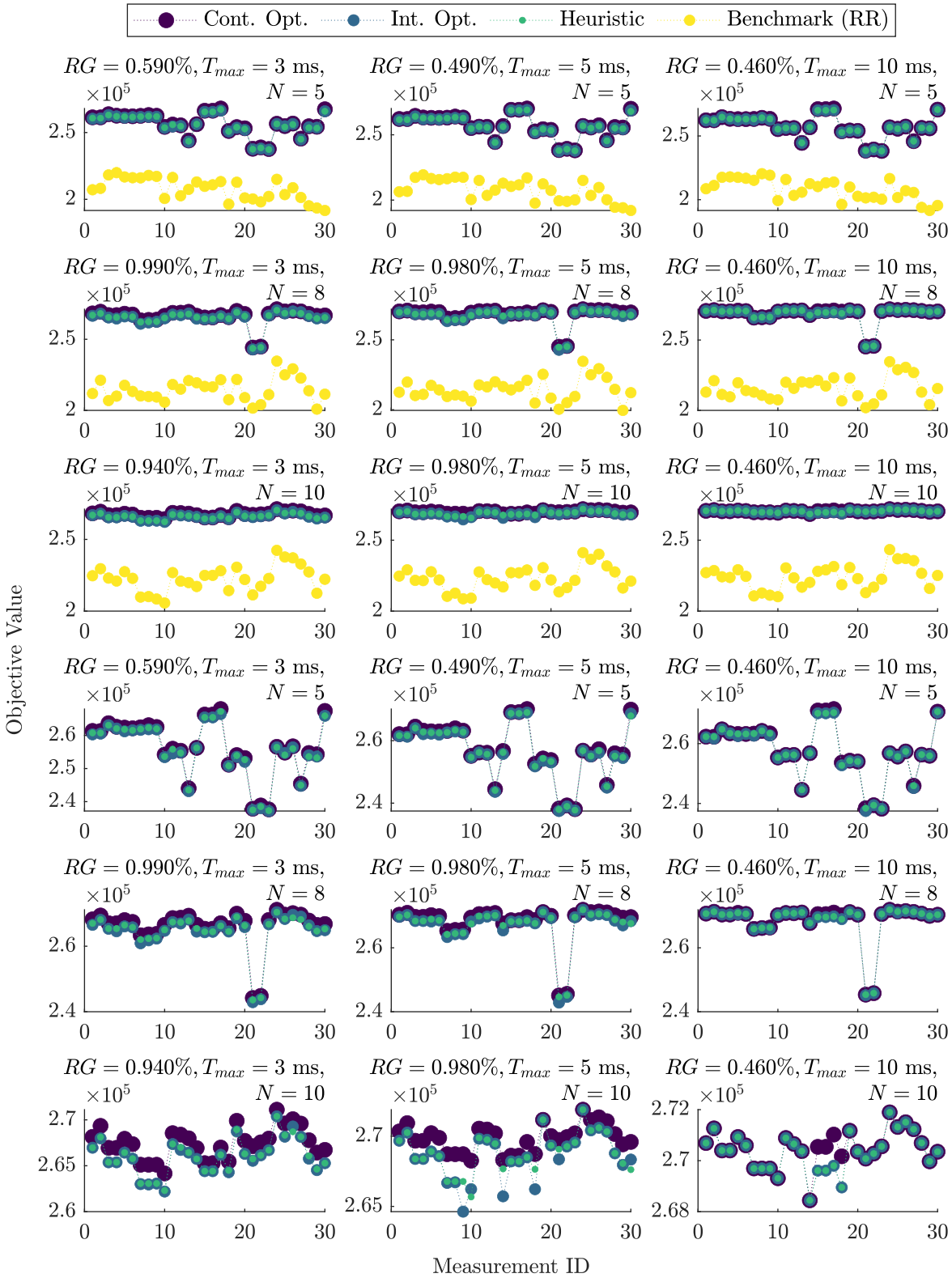


Figure 5.1: Obj. values for various CQI inputs for $\alpha = 0$ in the uplink-only scenario.

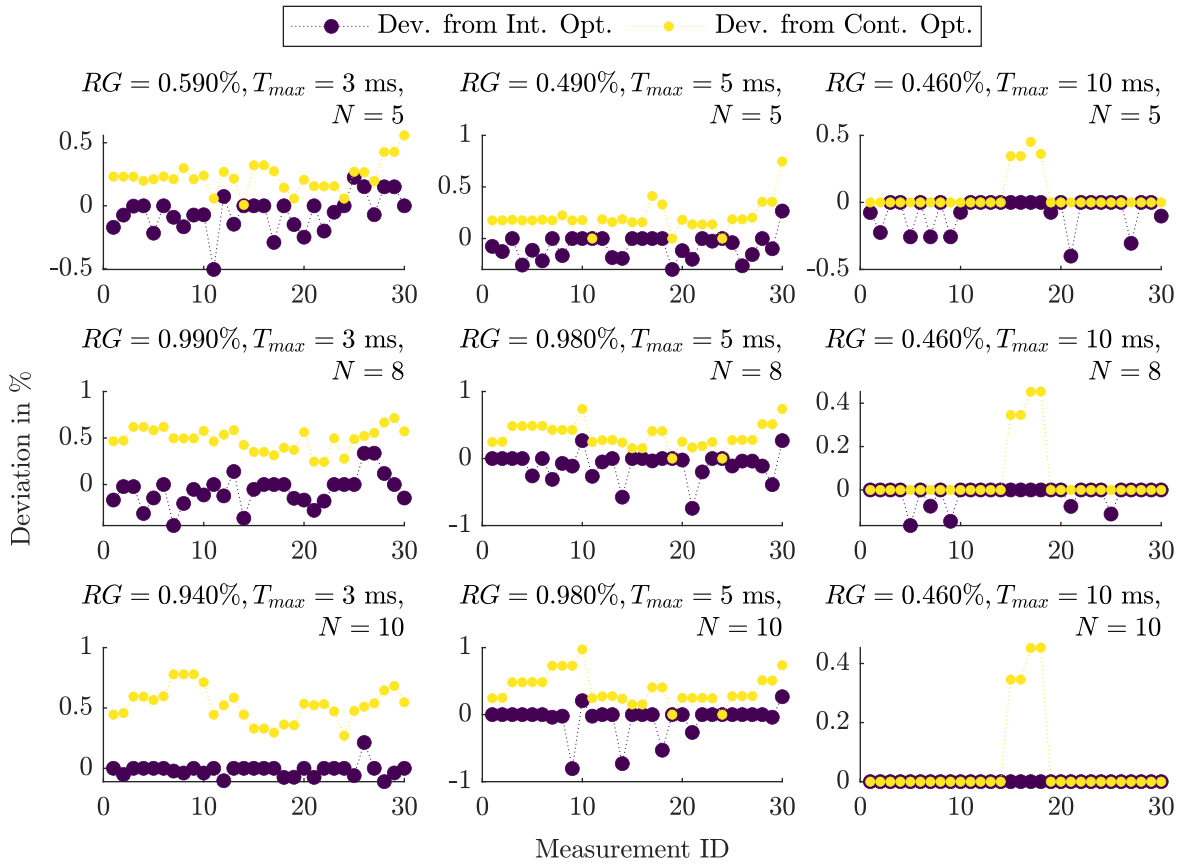


Figure 5.2: Deviation of the heuristic solution from the continuous/integer optimum for $\alpha = 0$ in the uplink-only scenario.

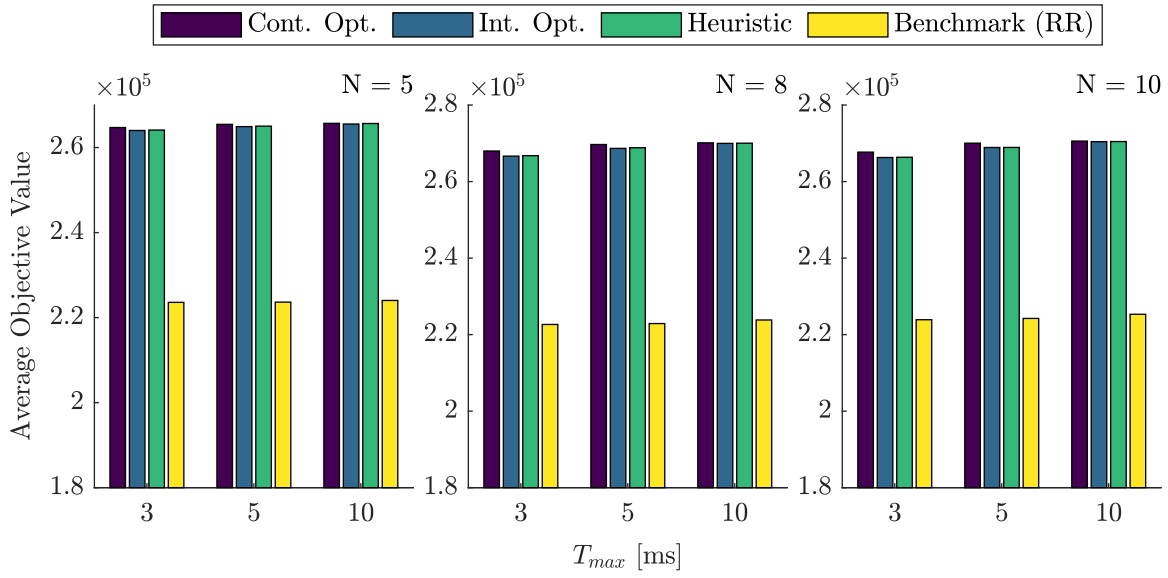


Figure 5.3: Average objective values for $\alpha = 0$ in the uplink-only scenario.

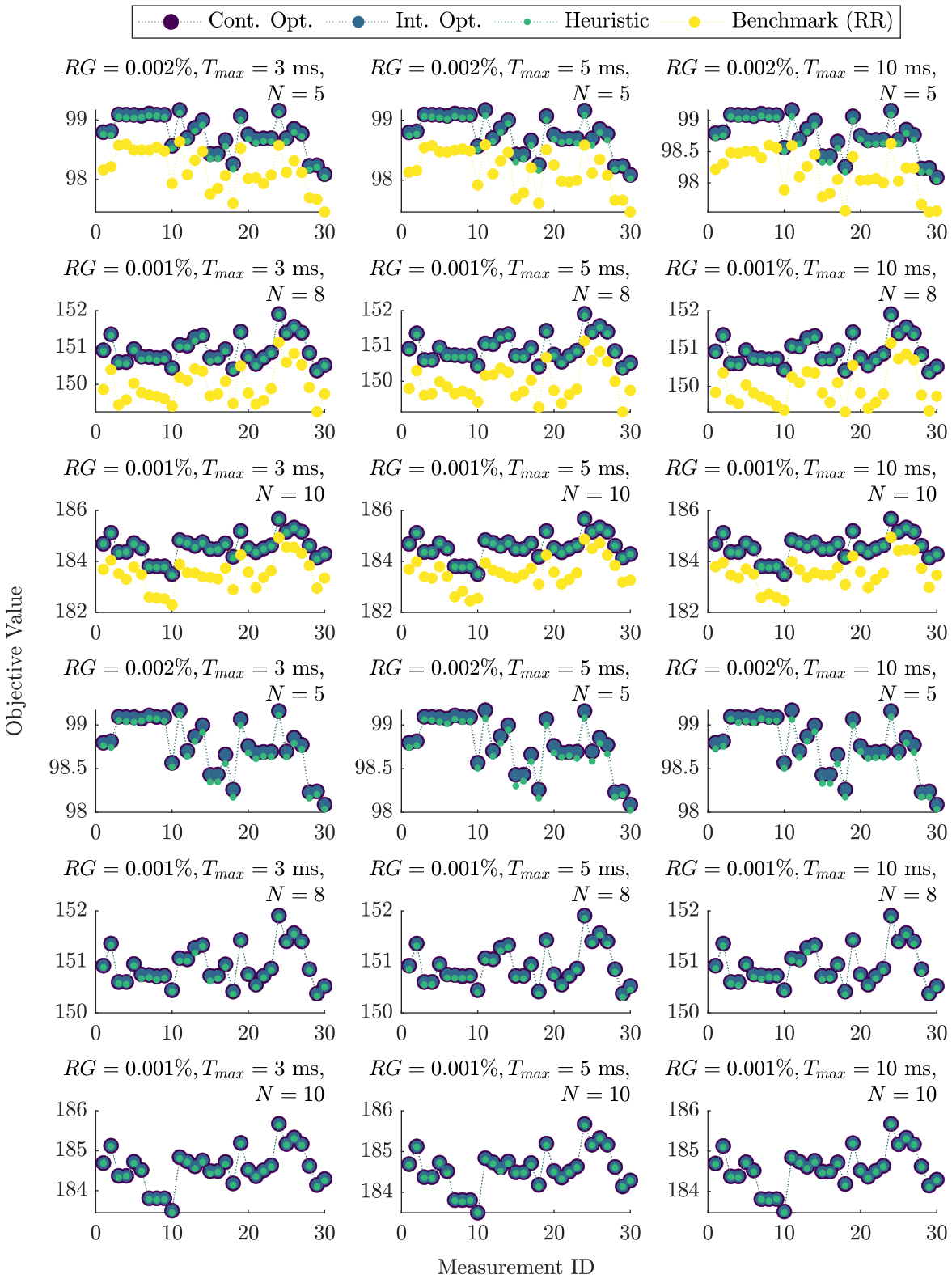


Figure 5.4: Obj. values for various CQI inputs for $\alpha = 1$ in the uplink-only scenario.

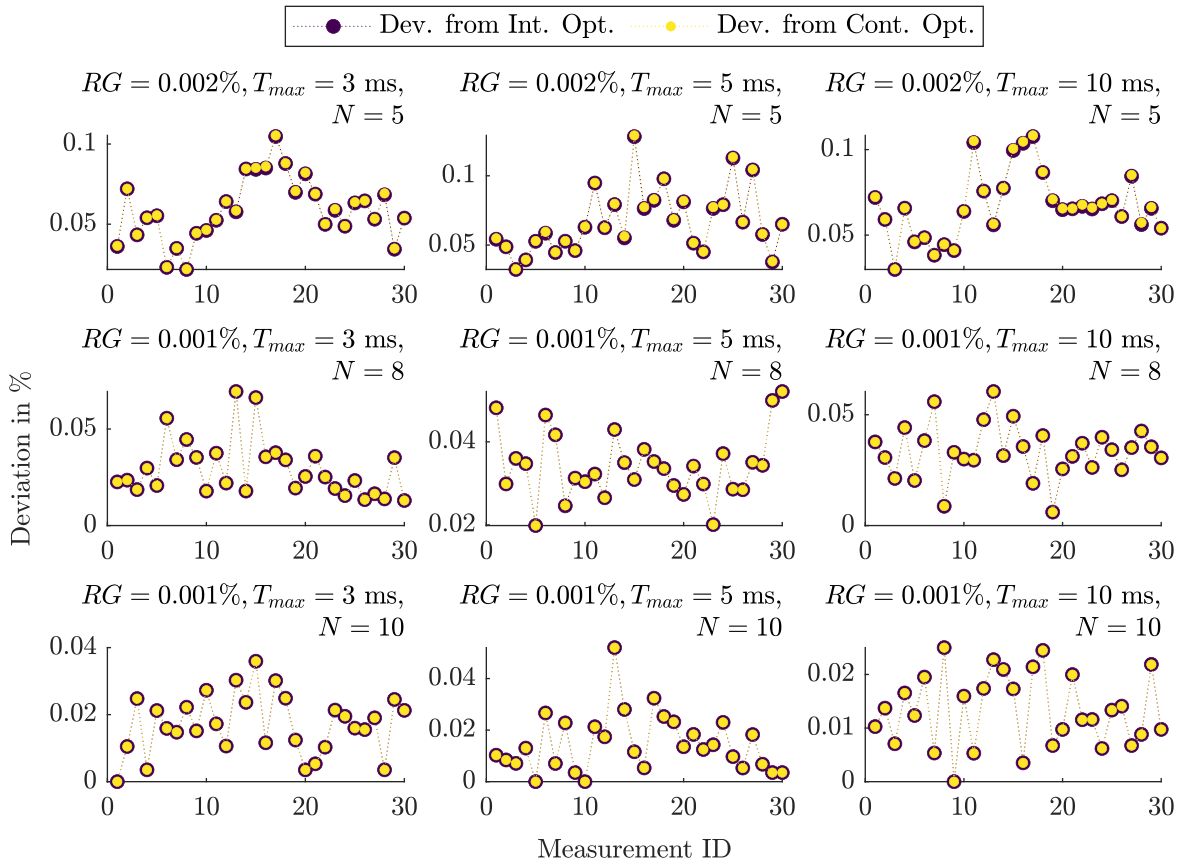


Figure 5.5: Deviation of the heuristic solution from the continuous/integer optimum for $\alpha = 1$ in the uplink-only scenario.

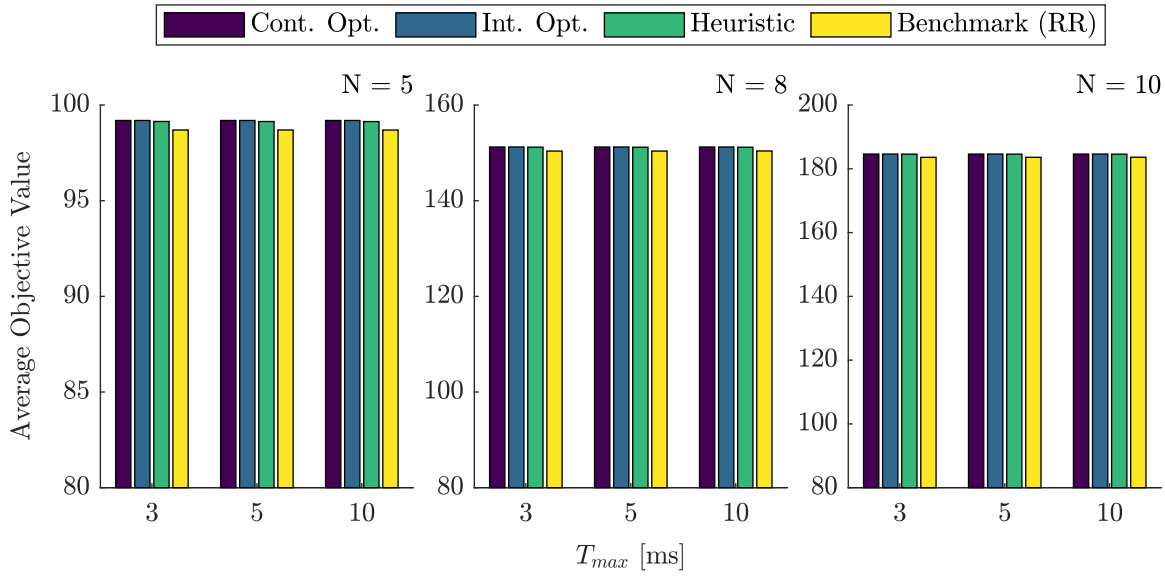


Figure 5.6: Average objective values for $\alpha = 1$ in the uplink-only scenario.

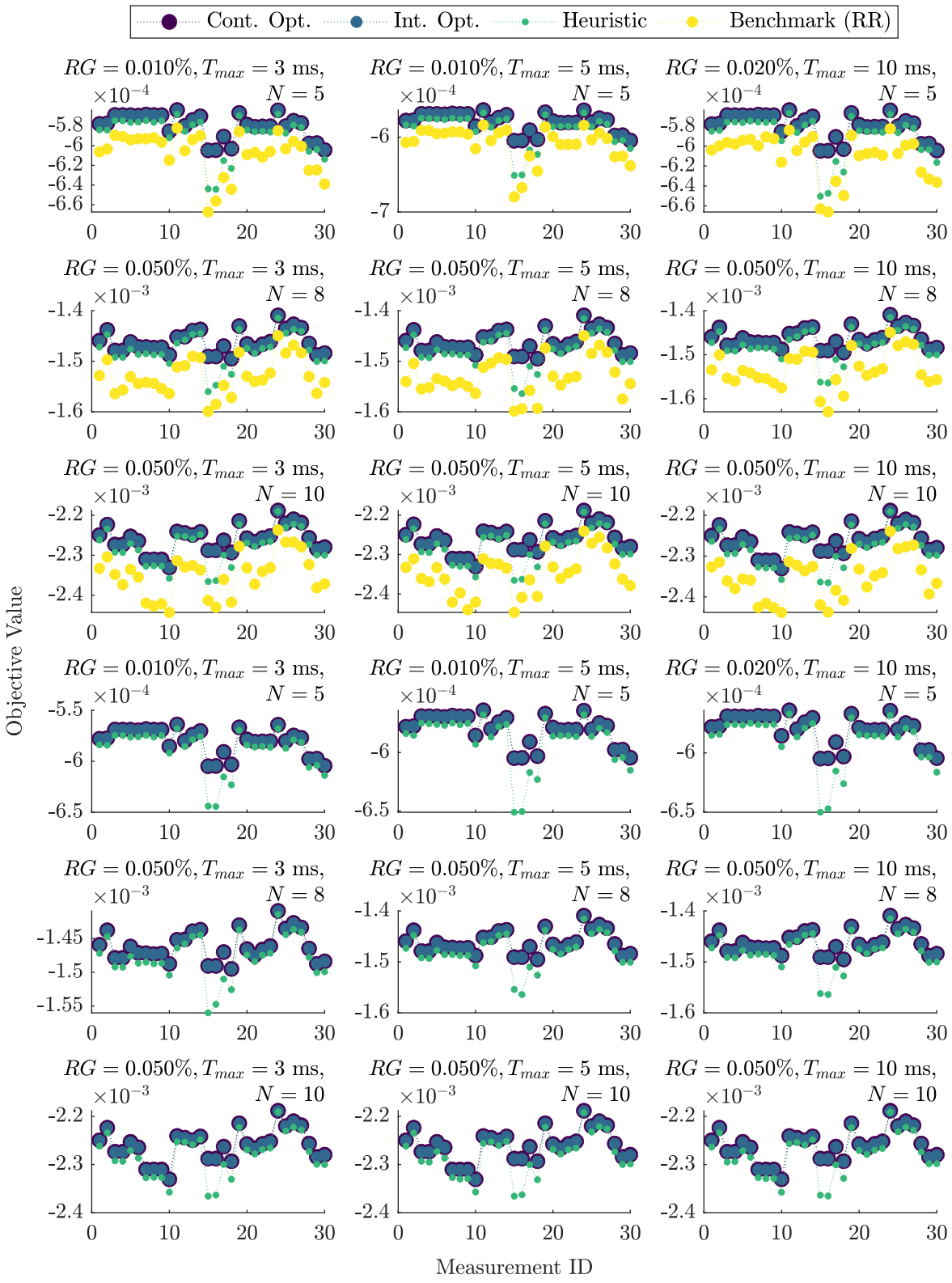


Figure 5.7: Obj. values for various CQI inputs for $\alpha = 2$ in the uplink-only scenario.

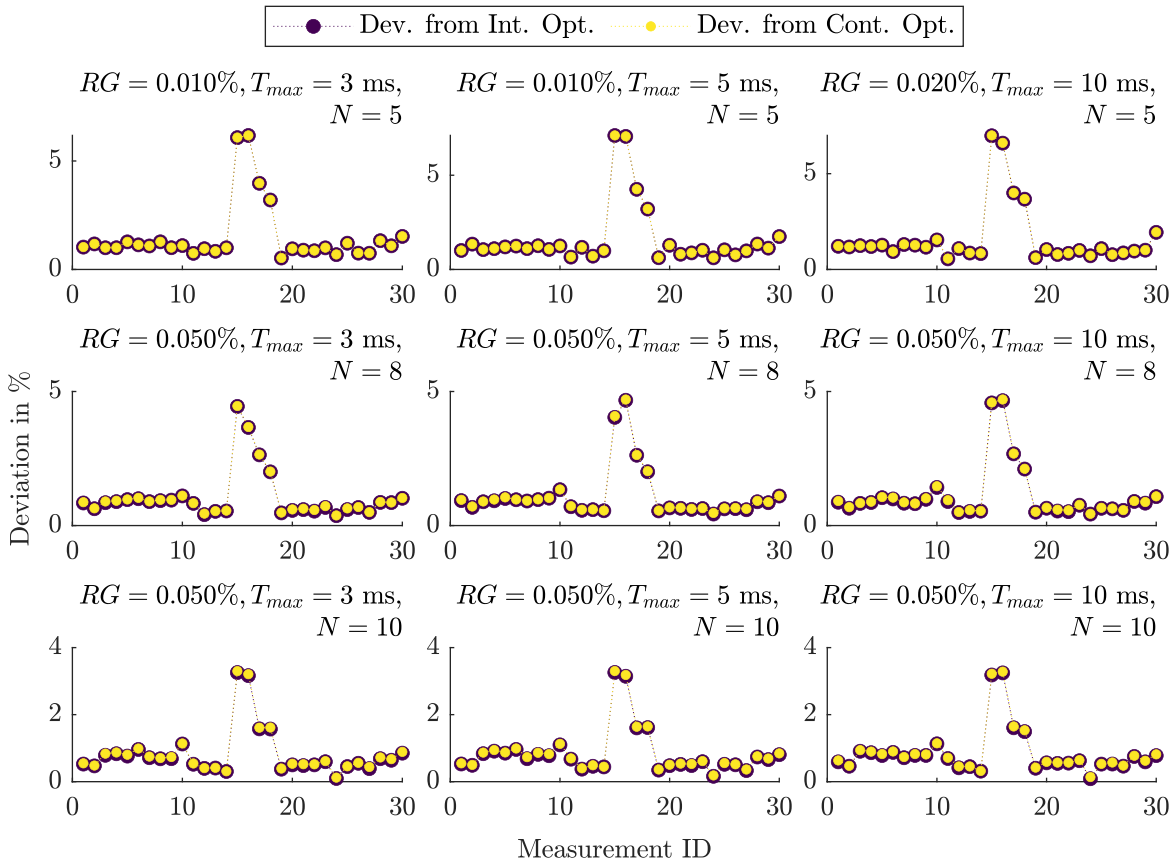


Figure 5.8: Deviation of the heuristic solution from the continuous/integer optimum for $\alpha = 2$ in the uplink-only scenario.

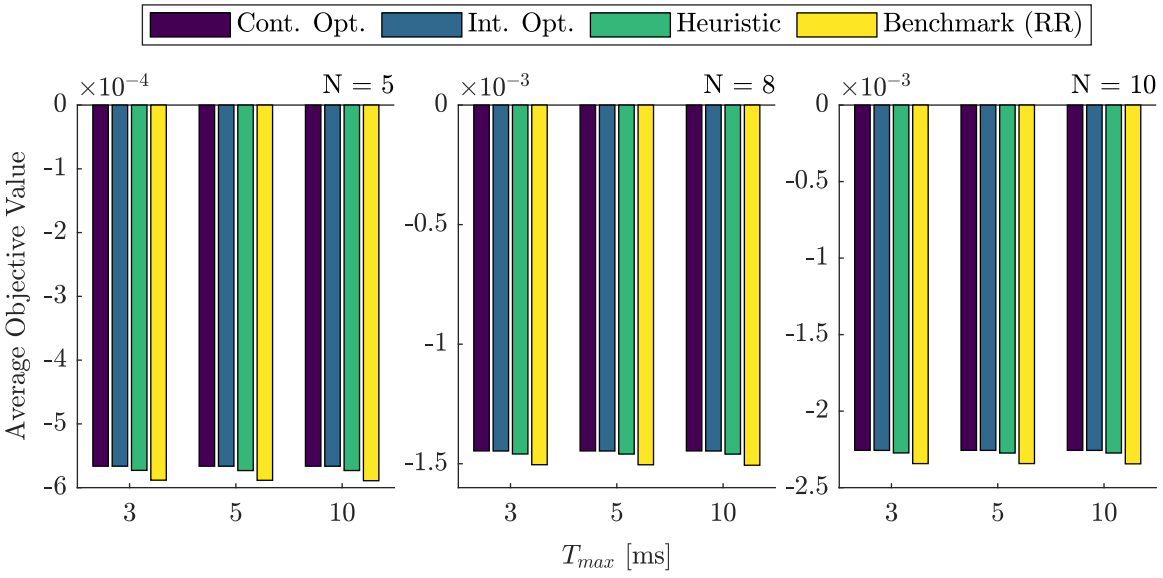


Figure 5.9: Average objective values for $\alpha = 2$ in the uplink-only scenario.

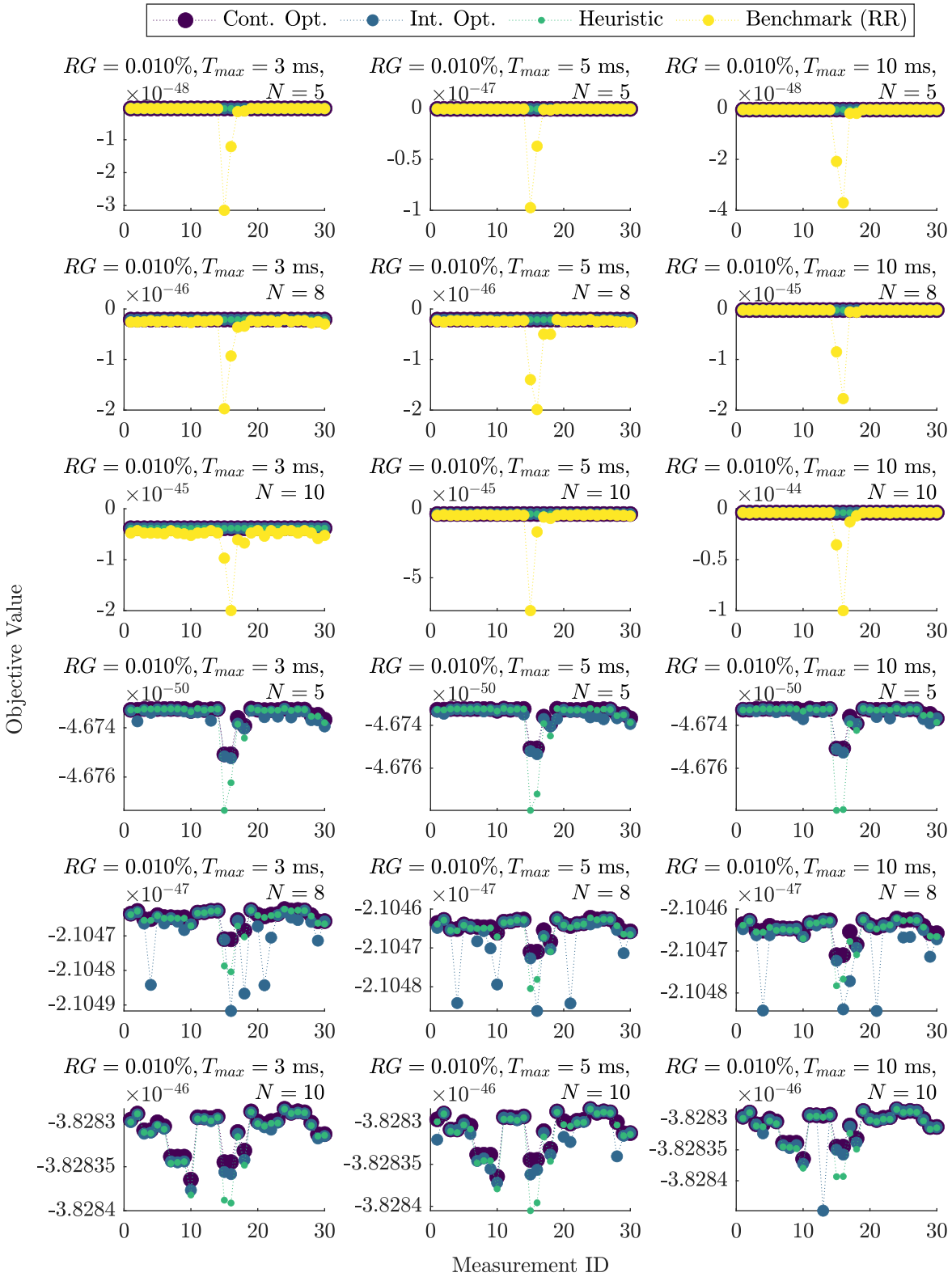


Figure 5.10: Obj. values for various CQI inputs for $\alpha = 13$ in the uplink-only scenario.

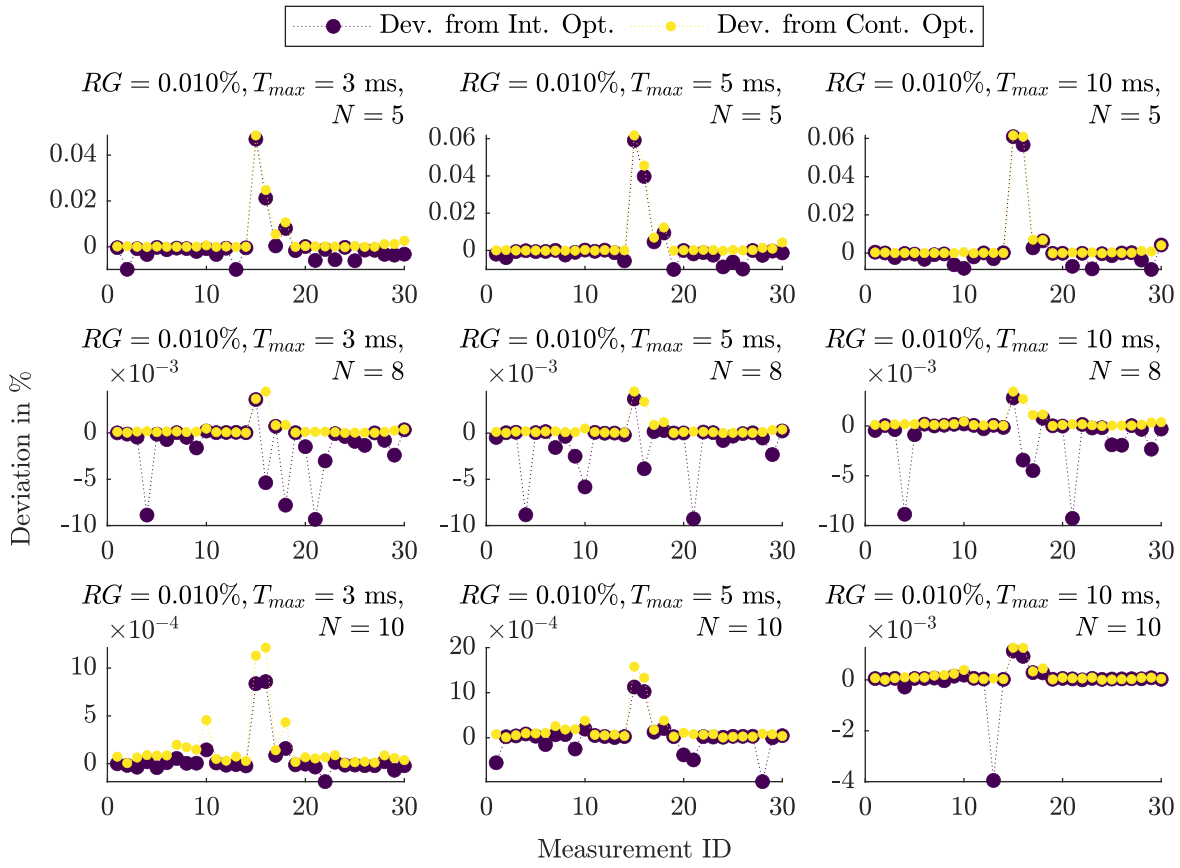


Figure 5.11: Deviation of the heuristic solution from the continuous/integer optimum for $\alpha = 13$ in the uplink-only scenario.

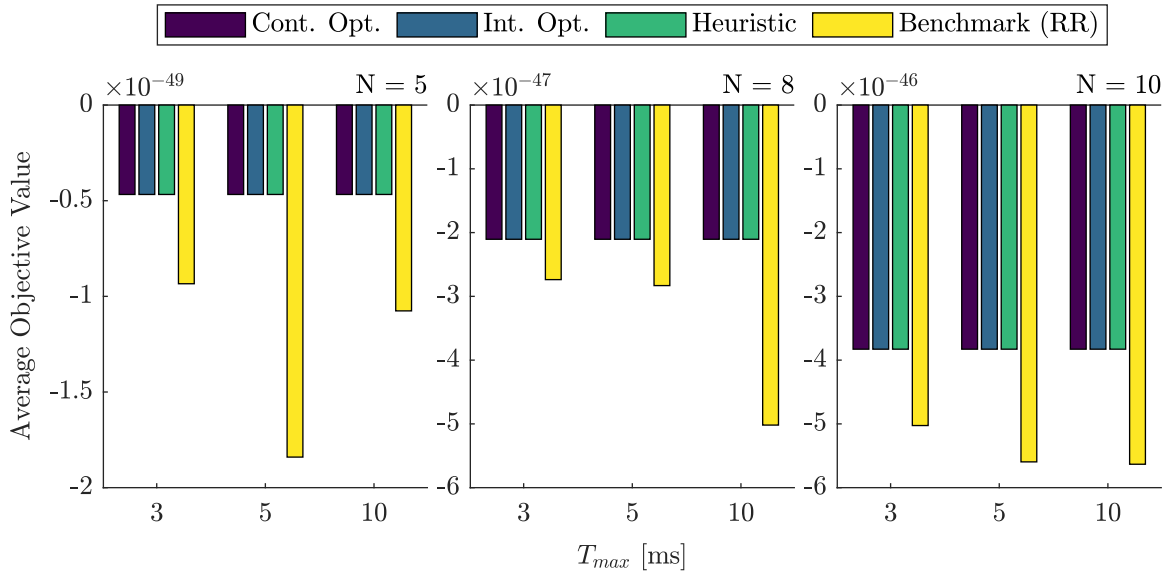


Figure 5.12: Average objective values for $\alpha = 13$ in the uplink-only scenario.

6 Scenario 2: Uplink and Downlink Communication with Edge Processing

Following the same structure as in Chapter 5, in this chapter, allocating RAN and edge computing resources in the second scenario, i.e., a moving user is sending data via a RAN to a BS where this data is processed and a response is generated, is examined. First, the mathematical optimization problem introduced in Section 5.1 is adapted to the new system model. Subsequently, the optimization formulation is investigated regarding its solvability and the approximation algorithms from Section 5.3 are adjusted to a two-way communication as well. Concluding, simulation data verify the good performance of the heuristics for the extended scenario.

The analyses and results of this chapter were submitted to the IEEE Journal on Selected Areas in Communications (JSAC), Issue: 3GPP Technologies: 5G-Advanced and Beyond [HMCK22a].

6.1 Optimization Problem Formulation

The objective of the optimization problem is again to maximize the overall utility, which is expanded with the third term representing the downlink RAN part. While most of the constraints remain unchanged, the downlink communication naturally influences the delay of the packets. Once more, with the focus set on providing general α-fairness, the adapted optimization problem is formulated as

$$\max_{\boldsymbol{I}_u, \boldsymbol{I}_d, \boldsymbol{m}} \sum_{i=1}^{N} f_i^{\alpha}(\boldsymbol{I}_{u,i}, \boldsymbol{I}_{d,i}, m_i) \tag{6.1a}$$

$$\text{s.t. } \frac{\Delta_{u,i}}{\sum_{j=1}^{K_u} I_{u,ij}\Phi_{u,ij}} + \frac{\Delta_{u,i}}{m_i p} + \frac{\Delta_{d,i}}{\sum_{j=1}^{K_d} I_{d,ij}\Phi_{d,ij}} \leq T_{max}, \quad \forall i \in \mathcal{U}, \tag{6.1b}$$

$$\sum_{i=1}^{N} m_i \leq L, \tag{6.1c}$$

$$\sum_{i=1}^{N} I_{\{u,d\},ij} \leq 1, \quad \forall j \in \mathcal{K}_{\{u,d\}}, \tag{6.1d}$$

$$\sum_{j=1}^{K_{\{u,d\}}} I_{\{u,d\},ij} \geq 1, \quad \forall i \in \mathcal{U}, \tag{6.1e}$$

$$I_{\{u,d\},ij} \in \{0,1\}, \quad \forall i \in \mathcal{U},\ j \in \mathcal{K}_{\{u,d\}}, \tag{6.1f}$$

$$m_i \in \mathbb{N} \setminus \{0\}, \quad \forall i \in \mathcal{U}, \tag{6.1g}$$

where

$$f_i^{\alpha}(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i) = \\ = \begin{cases} \frac{1}{1-\alpha}\left(\left(\sum_{j=1}^{K_u} I_{u,ij}\Phi_{u,ij}\right)^{1-\alpha} + (m_i p)^{1-\alpha} + \left(\sum_{j=1}^{K_d} I_{d,ij}\Phi_{d,ij}\right)^{1-\alpha}\right), & \alpha \neq 1 \\ \log\left(\sum_{j=1}^{K_u} I_{u,ij}\Phi_{u,ij}\right) + \log\left(m_i p\right) + \log\left(\sum_{j=1}^{K_d} I_{d,ij}\Phi_{d,ij}\right), & \alpha = 1 \end{cases}. \tag{6.2}$$

Relating to the previously introduced expressions, the decision variable $\boldsymbol{I}_d = \{I_{d,ij}\}$ denotes the $N \times K_d$ downlink PRB allocation matrix in a given frame. In the same manner as before, this means that if $I_{d,ij} = 1$, then PRB j is allocated to user i in that frame. The downlink data rates which are deduced from the CQI values that are given for each user are contained in the $N \times K_d$ matrix $\boldsymbol{\Phi}_{d,ij}$, denoted in the same sense as the uplink data rate matrix $\boldsymbol{\Phi}_{u,ij}$. Clearly, the variable $\Delta_{d,i}$ denotes the size of the data of user i's downlink packets. For an explanation of the remaining mathematical expressions, the reader is referred to Section 5.1.

The objective (6.1a) maximizes the overall utility for general $\alpha \in [0, \infty)$. The first and third term in (6.2) (both for $\alpha \neq 1$ and $\alpha = 1$) correspond to the utility from assigning uplink or downlink RAN resources to user i, whereas the second term denotes the utility after allocating a fraction of the edge computing resources.

The maximum tolerable latency which is extended by the downlink delay for every user is described by constraint (6.1b). Constraint (6.1c) stays unchanged and captures the finite amount of computing resources that are available. On the one hand, every uplink and downlink block can be assigned to at most one user, which is indicated by constraint (6.1d). On the other hand, constraint (6.1e) dictates that every user has to receive at least one PRB both in the uplink and downlink. Lastly, the integer nature of the decision variables is described by (6.1f) and (6.1g), where the latter constraint includes the minimum number of one edge computing resource that needs to be assigned to every user.

6.2 Analysis

Since the adjusted optimization problem still is an Integer Nonlinear Program, the problem is still NP-hard [LL11]. Hence, approximation algorithms are once more needed to obtain a near-optimal solution to (6.1).

The pursued procedure consists again of the two main steps of proving the polynomial-time solvability of the integer-relaxed version of the optimization problem (6.1) and the development of special approximation algorithms. Thereby, the already introduced heuristics from Section 5.3 are adjusted to the scenario considered in this chapter.

Continuing with the first step, the convexity of (6.1), when $I_{\{u,d\},ij} \in [0,1]$ and $m_i \in [1,\infty)$, is shown. Since the constraints (6.1c)-(6.1g) are still linear inequalities, they are apparently convex. In order to prove the concavity of the objective function, the concavity of $f_i^\alpha(\boldsymbol{I}_{u,i}, \boldsymbol{I}_{d,i}, m_i)$ needs to be shown. It can be stated:

Lemma 14. *The function $f_i^\alpha(\boldsymbol{I}_{u,i}, \boldsymbol{I}_{d,i}, m_i)$ is concave.*

Proof. The gradient of $f_i^\alpha(\boldsymbol{I}_{u,i}, \boldsymbol{I}_{d,i}, m_i)$ for $\alpha \neq 1$ is

$$\nabla f_i^\alpha(\boldsymbol{I}_{u,i}, \boldsymbol{I}_{d,i}, m_i) = \big[\Phi_{u,i1}\gamma_{u,i}^{-\alpha} \quad \dots \quad \Phi_{u,iK_u}\gamma_{u,i}^{-\alpha} \quad p(m_i p)^{-\alpha} \quad \Phi_{d,i1}\gamma_{d,i}^{-\alpha} \quad \dots \quad \Phi_{d,iK_d}\gamma_{d,i}^{-\alpha}\big]^T, \quad (6.3)$$

whereas the gradient of $f_i^\alpha(\boldsymbol{I}_{u,i}, \boldsymbol{I}_{d,i}, m_i)$ for $\alpha = 1$ is

$$\nabla f_i^\alpha(\boldsymbol{I}_{u,i}, \boldsymbol{I}_{d,i}, m_i) = \big[\Phi_{u,i1}\gamma_{u,i}^{-1} \quad \dots \quad \Phi_{u,iK_u}\gamma_{u,i}^{-1} \quad m_i^{-1} \quad \Phi_{d,i1}\gamma_{d,i}^{-1} \quad \dots \quad \Phi_{d,iK_d}\gamma_{d,i}^{-1}\big]^T, \quad (6.4)$$

where $\gamma_{d,i} = \sum_{j=1}^{K_d} I_{d,ij}\Phi_{d,ij}$ in the same manner as $\gamma_{u,i}$. Next, the Hessian matrix of

$f_i^\alpha(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i)$ for $\alpha \neq 1$ is calculated as

$$\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i) =$$
$$= -\alpha \begin{bmatrix} \Phi_{u,i1}^2/\gamma_{u,i}^{\alpha+1} & \dots & \Phi_{u,i1}\Phi_{u,iK_u}/\gamma_{u,i}^{\alpha+1} \\ \vdots & \ddots & \vdots \\ \Phi_{u,iK_u}\Phi_{u,i1}/\gamma_{u,i}^{\alpha+1} & \dots & \Phi_{u,iK_u}^2/\gamma_{u,i}^{\alpha+1} \\ 0 & \dots & 0 \\ 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{bmatrix}$$
$$\begin{bmatrix} 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \\ p^2/(m_i p)^{\alpha+1} & 0 & \dots & 0 \\ 0 & \Phi_{d,i1}^2/\gamma_{d,i}^{\alpha+1} & \dots & \Phi_{d,i1}\Phi_{d,iK_d}/\gamma_{d,i}^{\alpha+1} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \Phi_{d,iK_d}\Phi_{d,i1}/\gamma_{d,i}^{\alpha+1} & \dots & \Phi_{d,iK_d}^2/\gamma_{d,i}^{\alpha+1} \end{bmatrix}, \quad (6.5)$$

and the Hessian of $f_i^\alpha(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i)$ for $\alpha = 1$ is

$$\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i) =$$
$$= \begin{bmatrix} \Phi_{u,i1}^2/\gamma_{u,i}^2 & \dots & \Phi_{u,i1}\Phi_{u,iK_u}/\gamma_{u,i}^2 & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \Phi_{u,iK_u}\Phi_{u,i1}/\gamma_{u,i}^2 & \dots & \Phi_{u,iK_u}^2/\gamma_{u,i}^2 & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & 1/m_i^2 & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & \Phi_{d,i1}^2/\gamma_{d,i}^2 & \dots & \Phi_{d,i1}\Phi_{d,iK_d}/\gamma_{d,i}^2 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & \Phi_{d,iK_d}\Phi_{d,i1}/\gamma_{d,i}^2 & \dots & \Phi_{d,iK_d}^2/\gamma_{d,i}^2 \end{bmatrix}. \quad (6.6)$$

Then, the characteristic polynomial of $\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i)$ for $\alpha \neq 1$ is computed as

$$\begin{aligned} \det(\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i) - \lambda\mathbb{I}) = (-1)^{K_u+K_d-1}\lambda^{K_u+K_d-2}\left(\alpha p^2 (m_i p)^{-\alpha-1} + \lambda\right) * \\ \left(\lambda^2 + \alpha^2 \gamma_{u,i}^{-\alpha-1}\gamma_{d,i}^{-\alpha-1} * \left(\Phi_{u,i1}^2\Phi_{d,i1}^2 + \dots + \Phi_{u,iK_u}^2\Phi_{d,iK_d}^2\right) + \right. \\ \left. \lambda\alpha\left(\Phi_{u,i1}^2\gamma_{u,i}^{-\alpha-1} + \dots + \Phi_{u,iK_u}^2\gamma_{u,i}^{-\alpha-1} + \Phi_{d,i1}^2\gamma_{d,i}^{-\alpha-1} + \dots + \Phi_{d,iK_d}^2\gamma_{d,i}^{-\alpha-1}\right)\right), \end{aligned} \quad (6.7)$$

while the characteristic polynomial of $\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i)$ for $\alpha = 1$ is given as

$$\begin{aligned} \det(\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i) - \lambda\mathbb{I}) = (-1)^{K_u+K_d-1}\lambda^{K_u+K_d-2}\left(m_i^{-2} + \lambda\right) * \\ \left(\lambda^2 + \gamma_{u,i}^{-2}\gamma_{d,i}^{-2} * \left(\Phi_{u,i1}^2\Phi_{d,i1}^2 + \dots + \Phi_{u,iK_u}^2\Phi_{d,iK_d}^2\right) + \right. \\ \left. \lambda\left(\Phi_{u,i1}^2\gamma_{u,i}^{-2} + \dots + \Phi_{u,iK_u}^2\gamma_{u,i}^{-2} + \Phi_{d,i1}^2\gamma_{d,i}^{-2} + \dots + \Phi_{d,iK_d}^2\gamma_{d,i}^{-2}\right)\right), \end{aligned} \quad (6.8)$$

where $\mathbb{I}$ denotes the identity matrix in the corresponding dimension and λ are the eigenvalues of the Hessian $\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i}, \boldsymbol{I}_{d,i}, m_i)$. Finally, for $\alpha \neq 1$, the eigenvalues can be determined as

$$\lambda_1, ..., \lambda_{K_u+K_d-2} = 0, \tag{6.9a}$$
$$\lambda_{K_u+K_d-1} = -\alpha\gamma_{u,i}^{-\alpha-1}\left(\Phi_{u,i1}^2 + \cdots + \Phi_{u,iK_u}^2\right), \tag{6.9b}$$
$$\lambda_{K_u+K_d} = -\alpha\gamma_{d,i}^{-\alpha-1}\left(\Phi_{d,i1}^2 + \cdots + \Phi_{d,iK_d}^2\right), \tag{6.9c}$$
$$\lambda_{K_u+K_d+1} = -\alpha p^2 (m_i p)^{-\alpha-1}, \tag{6.9d}$$

and the eigenvalues of the Hessian $\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i}, \boldsymbol{I}_{d,i}, m_i)$ for $\alpha = 1$ are

$$\lambda_1, ..., \lambda_{K_u+K_d-2} = 0, \tag{6.10a}$$
$$\lambda_{K_u+K_d-1} = -\gamma_{u,i}^{-2}\left(\Phi_{u,i1}^2 + \cdots + \Phi_{u,iK_u}^2\right), \tag{6.10b}$$
$$\lambda_{K_u+K_d} = -\gamma_{d,i}^{-2}\left(\Phi_{d,i1}^2 + \cdots + \Phi_{d,iK_d}^2\right), \tag{6.10c}$$
$$\lambda_{K_u+K_d+1} = -m_i^{-2}. \tag{6.10d}$$

Hence, the Hessian $\nabla^2 f_i^\alpha(\boldsymbol{I}_{u,i}, \boldsymbol{I}_{d,i}, m_i)$ is negative semidefinite $\forall\alpha$, as all eigenvalues of the Hessian are less than or equal to 0 and thus the function $f_i^\alpha(\boldsymbol{I}_{u,i}, \boldsymbol{I}_{d,i}, m_i)$ is concave $\forall\alpha$. □

Next, the characteristics of (6.1b) are explored.

Lemma 15. *Constraint* (6.1b) *is convex.*

Proof. Denote the left-hand side of (6.1b) as

$$t_i(\boldsymbol{I}_{u,i}, \boldsymbol{I}_{d,i}, m_i) = \frac{\Delta_{u,i}}{\sum_{j=1}^{K_u} I_{u,ij}\Phi_{u,ij}} + \frac{\Delta_{u,i}}{m_i p} + \frac{\Delta_{d,i}}{\sum_{j=1}^{K_d} I_{d,ij}\Phi_{d,ij}} = \frac{\Delta_{u,i}}{\gamma_{u,i}} + \frac{\Delta_{u,i}}{m_i p} + \frac{\Delta_{d,i}}{\gamma_{d,i}}. \tag{6.11}$$

Calculating the gradient of $t_i(\boldsymbol{I}_{u,i}, \boldsymbol{I}_{d,i}, m_i)$ leads to

$$\nabla t_i(\boldsymbol{I}_{u,i}, \boldsymbol{I}_{d,i}, m_i) = \Bigg[\frac{-\Delta_{u,i}\Phi_{u,i1}}{\gamma_{u,i}^2} \quad \cdots \quad \frac{-\Delta_{u,i}\Phi_{u,iK_u}}{\gamma_{u,i}^2} \quad \frac{-\Delta_{u,i}}{m_i^2 p} \quad \frac{-\Delta_{d,i}\Phi_{d,i1}}{\gamma_{d,i}^2} \quad \cdots \quad \frac{-\Delta_{d,i}\Phi_{d,iK_d}}{\gamma_{d,i}^2}\Bigg]^T. \tag{6.12}$$

The Hessian of $t_i(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i)$ is given as

$$\nabla^2 t_i(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i) = \\ = \begin{bmatrix} 2\Delta_{u,i}\Phi_{u,i1}^2/\gamma_{u,i}^3 & \dots & 2\Delta_{u,i}\Phi_{u,i1}\Phi_{u,iK_u}/\gamma_{u,i}^3 \\ \vdots & \ddots & \vdots \\ 2\Delta_{u,i}\Phi_{u,iK_u}\Phi_{u,i1}/\gamma_{u,i}^3 & \dots & 2\Delta_{u,i}\Phi_{u,iK_u}^2/\gamma_{u,i}^3 \\ 0 & \dots & 0 \\ 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \\ 2\Delta_{u,i}/m_i^3 p & 0 & \dots & 0 \\ 0 & 2\Delta_{d,i}\Phi_{d,i1}^2/\gamma_{d,i}^3 & \dots & 2\Delta_{d,i}\Phi_{d,i1}\Phi_{d,iK_d}/\gamma_{d,i}^3 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 2\Delta_{d,i}\Phi_{d,iK_d}\Phi_{d,i1}/\gamma_{d,i}^3 & \dots & 2\Delta_{d,i}\Phi_{d,iK_d}^2/\gamma_{d,i}^3 \end{bmatrix}. \quad (6.13)$$

Then, the characteristic polynomial of $\nabla^2 t_i(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i)$ is determined as

$$\det(\nabla^2 t_i(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i) - \lambda\mathbb{I}) = (-1)^{K_u+K_d-1}\lambda^{K_u+K_d-2}\left(2\Delta_{u,i}m_i^{-3}p^{-1}+\lambda\right)* \\ \left(\lambda^2 + 4\Delta_{u,i}\Delta_{d,i}\gamma_{u,i}^{-2}\gamma_{d,i}^{-2}* \left(\Phi_{u,i1}^2\Phi_{d,i1}^2 + \dots + \Phi_{u,iK_u}^2\Phi_{d,iK_d}^2\right) + \right. \\ \left. 2\lambda\left(\Delta_{u,i}\gamma_{u,i}^{-2}\left(\Phi_{u,i1}^2 + \dots + \Phi_{u,iK_u}^2\right) + \Delta_{d,i}\gamma_{d,i}^{-2}\left(\Phi_{d,i1}^2 + \dots + \Phi_{d,iK_d}^2\right)\right)\right). \quad (6.14)$$

Lastly, the eigenvalues of the Hessian $\nabla^2 t_i(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i)$ can be found to be

$$\lambda_1, ..., \lambda_{K_u+K_d-2} = 0, \quad (6.15a)$$
$$\lambda_{K_u+K_d-1} = 2\Delta_{u,i}\gamma_{u,i}^{-3}\left(\Phi_{u,i1}^2 + \dots + \Phi_{u,iK_u}^2\right), \quad (6.15b)$$
$$\lambda_{K_u+K_d} = 2\Delta_{d,i}\gamma_{d,i}^{-3}\left(\Phi_{d,i1}^2 + \dots + \Phi_{d,iK_d}^2\right), \quad (6.15c)$$
$$\lambda_{K_u+K_d+1} = 2\Delta_{u,i}m_i^{-3}p^{-1}. \quad (6.15d)$$

Thus, the Hessian $\nabla^2 t_i(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i)$ is positive semidefinite and the function $t_i(\boldsymbol{I}_{u,i},\, \boldsymbol{I}_{d,i},\, m_i)$ is convex, because all the eigenvalues of the Hessian are greater than or equal to zero. □

Theorem 16. *The integer-relaxed optimization* (6.1) *is a convex optimization problem.*

Proof. Given the linearity of (6.1c)-(6.1g) as well as Lemmas 14 and 15 proves that (6.1) is a convex optimization problem. □

For the next main step of proving the polynomial-time solvability of the integer-relaxed optimization problem, (6.1) is rewritten into a convex optimization problem with generalized inequality constraints. For the following derivations, define the n-dimensional rotated quadratic cone as

$$\mathcal{Q}_r^n = \left\{ \boldsymbol{x} \in \mathbb{R}^n \,|\, 2x_1x_2 \geq x_3^2 + \cdots + x_n^2,\, x_1,\, x_2 \geq 0 \right\}, \tag{6.16}$$

and recall the definitions of the n-dimensional power cone $\mathcal{P}_\zeta^n$ (5.17) as well as the exponential cone $\mathcal{E}$ (5.18).

To start with, introduce the slack variables s_{ki}, $k \in \{1,2,3\} = \mathcal{L}_{ud}$, $i \in \mathcal{U}$ and write the relaxed optimization problem in epigraph form, such that it reads as

$$\begin{aligned}
\min_{g,\boldsymbol{I}_u,\boldsymbol{I}_d,\boldsymbol{m},\boldsymbol{s}} \quad & g && \text{(6.17a)}\\
\text{s.t.} \quad & -\sum_{i=1}^{N} h_i^\alpha(s_{1i},\, s_{2i},\, s_{3i},\, g) \leq 0, && \text{(6.17b)}\\
& \frac{\Delta_{u,i}}{s_{1i}} + \frac{\Delta_{u,i}}{s_{2i}} + \frac{\Delta_{d,i}}{s_{3i}} - T_{max} \leq 0, \quad \forall i \in \mathcal{U}, && \text{(6.17c)}\\
& \text{(6.1c), (6.1d), (6.1e)}, && \text{(6.17d)}\\
& 0 \leq I_{\{u,d\},ij} \leq 1, \quad \forall i \in \mathcal{U},\, j \in \mathcal{K}_{\{u,d\}}, && \text{(6.17e)}\\
& 1 - m_i \leq 0, \quad \forall i \in \mathcal{U}, && \text{(6.17f)}\\
& s_{1i} = \sum_{j=1}^{K_u} I_{u,ij}\Phi_{u,ij}, \quad \forall i \in \mathcal{U}, && \text{(6.17g)}\\
& s_{2i} = m_i p, \quad \forall i \in \mathcal{U} && \text{(6.17h)}\\
& s_{3i} = \sum_{j=1}^{K_d} I_{d,ij}\Phi_{d,ij}, \quad \forall i \in \mathcal{U}, && \text{(6.17i)}
\end{aligned}$$

where

$$h_i^\alpha(s_{1i},\, s_{2i},\, s_{3i},\, g) = \begin{cases} \frac{1}{1-\alpha}\left(s_{1i}^{1-\alpha} + s_{2i}^{1-\alpha} + s_{3i}^{1-\alpha}\right) + g, & \alpha \neq 1 \\ \log\left(s_{1i}\right) + \log\left(s_{2i}\right) + \log\left(s_{3i}\right) + g, & \alpha = 1 \end{cases}. \tag{6.18}$$

Proceeding, conic reformulations for the constraints (6.17b) and (6.17c) are introduced.

Lemma 17. *The constraint* (6.17c) *can be written as*

$$\begin{aligned}
\sum_{k=1}^{3} u_{ki} &\leq T_{max}, && \text{(6.19a)}\\
(u_{ki},\, s_{ki};\, \Delta_{k,i}) &\in \mathcal{Q}_r^3 \quad \forall k \in \mathcal{L}_{ud}, && \text{(6.19b)}
\end{aligned}$$

where $\Delta_{k,i} = \Delta_{u,i}$ *for* $k = \{1,2\}$ *and* $\Delta_{k,i} = \Delta_{d,i}$ *for* $k = 3$.

Proof. (6.19b) is by definition transformed to

$$u_{ki} s_{ki} \geq \sqrt{\Delta_{k,i}^2},\ u_{ki},\ s_{ki} \geq 0. \tag{6.20}$$

Dividing this expression by s_{ki} and extracting the root leads to

$$u_{ki} \geq \frac{\Delta_{k,i}}{s_{ki}},\ u_{ki},\ s_{ki} \geq 0, \tag{6.21}$$

from where it can be observed that

$$\sum_{k=1}^{3} \frac{\Delta_{k,i}}{s_{ki}} \leq T_{max}. \tag{6.22}$$

Due to the positiveness of $\Delta_{k,i}$ and s_{ki}, which follows from the constraints (6.17g) to (6.17i), the additional constraints $u_{ki},\ s_{ki} \geq 0$ that are introduced with this reformulation are always met. □

By setting $\beta = 1 - \alpha$ and using the slack variable u_{ki}, the constraint (6.17b) is transformed to the constraints (6.24) for the cases $\alpha \in (0, 1)$ and $\alpha \in (1, \infty)$. Bringing the sum over s_{ki}^{β} to the right side of the inequality results in

$$-g \leq \frac{1}{\beta} \sum_{k=1}^{3} \sum_{i=1}^{N} s_{ki}^{\beta}, \tag{6.23}$$

which can be converted into

$$(6.23) = \begin{cases} -g\beta \leq \sum_{k=1}^{3} \sum_{i=1}^{N} u_{ki}, & \text{(6.24a)} \\ u_{ki} \leq s_{ki}^{\beta}, \quad \forall k \in \mathcal{L}_{ud},\ i \in \mathcal{U};\ \alpha \in (0,1) & \text{(6.24b)} \\ g\,|\beta| \geq \sum_{k=1}^{3} \sum_{i=1}^{N} u_{ki}, & \text{(6.24c)} \\ u_{ki} \geq s_{ki}^{\beta}, \quad \forall k \in \mathcal{L}_{ud},\ i \in \mathcal{U};\ \alpha \in (1,\infty) & \text{(6.24d)} \end{cases}.$$

Since (6.24b) and (6.24d) are identical to (5.26b) and (5.26d), the former two inequalities can be reformulated using power cones as declared in Lemmas 7 and 8.

For the case $\alpha = 1$, (6.17b) must be reformulated to

$$-g \leq \sum_{k=1}^{3} \sum_{i=1}^{N} \log s_{ki}, \tag{6.25}$$

by bringing the sum over the logarithms to the right side of the inequality. (6.25) can be written as

$$-g \leq \sum_{k=1}^{3} \sum_{i=1}^{N} u_{ki}, \tag{6.26a}$$

$$u_{ki} \leq \log s_{ki}, \quad \forall k \in \mathcal{L}_{ud},\, i \in \mathcal{U}, \tag{6.26b}$$

by introducing again the slack variable u_{ki}.

The inequality (6.26b) is equal to (5.34b) and thus this constraint can be reformulated using an exponential cone as proven in Lemma 9.

Finally, the following theorem can be stated:

Theorem 18. *The integer-relaxed version of the optimization problem* (6.1) *can be written as a convex optimization problem with generalized inequality constraints.*

Proof. Given lemmas 7, 8, 9, and 17 and the fact that (6.17b) is linear for $\alpha = 0$ concludes the proof. □

With the preceding derivations, the optimization problem (6.1) reads for any $\alpha \in [0, \infty)$, written in its integer-relaxed version as a convex optimization problem with generalized inequality constraints, as:

$$\min_{g,\, \boldsymbol{I}_u,\, \boldsymbol{I}_d,\, \boldsymbol{m},\, \boldsymbol{s},\, \boldsymbol{u}} \quad g \tag{6.27a}$$

$$\text{s.t.} \qquad -\sum_{i=1}^{N} e_i^{\alpha}(s_{1i}, s_{2i}, s_{3i}, u_{1i}, u_{2i}, u_{3i}, g) \leq 0, \tag{6.27b}$$

$$\begin{aligned} &\text{(6.1c), (6.1d), (6.1e), (6.17e), (6.17f),} \\ &\text{(6.17g), (6.17h), (6.17i), (6.19),} \end{aligned} \tag{6.27c}$$

where

$$\text{(6.27b)} = \begin{cases} \text{(6.17b)}, & & \alpha = 0 \\ \text{(6.24a), (5.27)}, & \forall k \in \mathcal{L}_{ud},\, i \in \mathcal{U}, & 0 < \alpha < 1 \\ \text{(6.26a), (5.35)}, & \forall k \in \mathcal{L}_{ud},\, i \in \mathcal{U}, & \alpha = 1 \\ \text{(6.24c), (5.30)}, & \forall k \in \mathcal{L}_{ud},\, i \in \mathcal{U}, & \alpha > 1 \end{cases}. \tag{6.28}$$

For the final verification of the polynomial-time solvability of the optimization problem stated in (6.27), recall the definitions of the generalized logarithms for the quadratic, power, and exponential cone introduced in (5.40), (5.42), and (5.44) and their corresponding degrees. Since the rotated n-dimensional quadratic cone can be written as an ordinary quadratic cone by a rotation of coordinates, the generalized logarithm $\Gamma_{\mathcal{Q}}(\boldsymbol{x})$ is also valid for the rotated n-dimensional quadratic cone $\mathcal{Q}_r^n$. As explained in Section 5.2, the generalized

logarithm for an ordinary inequality can be given as the ordinary logarithm that is applied to a slack variable representing the inequality.

With the preceding definitions of the generalized logarithms for all constraints in (6.27), a logarithmic barrier function $\Lambda_{ud}(\boldsymbol{w}_{ud})$ can be given as

$$\Lambda_{ud}(\boldsymbol{w}_{ud}) = -\sum_{c=1}^{Z} \Gamma_c(\boldsymbol{w}_{ud}),\ \mathbf{dom}\,\Lambda = \{\boldsymbol{w}_{ud} \,|\, f_c(\boldsymbol{w}_{ud}) \prec_{K_c} 0,\ c = 1, ..., Z\}, \tag{6.29}$$

where $Z = (7+2K_u+2K_d)N+2+K_u+K_d$ for $\alpha = 0$ and $Z = (10+2K_u+2K_d)N+2+K_u+K_d$ for $\alpha \neq 0$. The vector $\boldsymbol{w}_{ud}$ is composed of the vectorized matrices $\boldsymbol{I}_u$ and $\boldsymbol{I}_d$ as well as the vectors $\boldsymbol{m}$, $\boldsymbol{s}$, and $\boldsymbol{u}$. The function $\Gamma_c(\boldsymbol{w}_{ud})$ denotes the generalized logarithms defined for each generalized inequality constraint $f_c(\boldsymbol{w}_{ud})$ in the convex optimization problem with generalized inequalities given in (6.27). As a logarithmic barrier function can be defined for the optimization problem, the barrier method can be applied to solve the problem.

Following the same steps as in Section 5.2, a complexity analysis based on the property of self-concordance is now presented.

Lemma 19. *The logarithmic barrier function $\Lambda_{ud}(\boldsymbol{w}_{ud})$ is self-concordant.*

Proof. First, note that the sum of self-concordant functions is again self-concordant [BV04]. Hence, the logarithmic barrier for the positive orthant defined by all slack variables corresponding to linear inequalities is a self-concordant function, because $-\log x$ is self-concordant. The logarithmic barriers established using the generalized logarithms defined in (5.40), (5.42), and (5.44) are self-concordant as well, see Section 11.6 in [BV04] and Sections 2.4 and 3.1 in [Cha09], which concludes the proof. □

Lemma 20. *The number of total Newton steps excluding the initial centering step for solving* (6.27) *using the Barrier method can be bounded by [BV04]*

$$T_{Barrier} = \left\lceil \frac{\log(\bar{\theta}/(t^{(0)}\xi))}{\log \mu} \right\rceil * \left(\frac{\bar{\theta}(\mu - 1 - \log \mu)}{\chi} + \log_2 \log_2(1/\xi) \right). \tag{6.30}$$

Proof. Given the fact that (6.27a) is linear and using Lemma 19, the objective of the Barrier method, i.e., the function $tg + \Lambda_{ud}(\boldsymbol{w}_{ud})$, is self-concordant. Given the additional properties that this function is closed and the sublevel sets of the optimization problem (6.27) are bounded leads to (6.30). □

For a definition of the parameters in (6.30) the reader is referred to Lemma 12. The variable $\bar{\theta}$ stands for the sum of the degrees of the generalized logarithms Γ_c, which for the contemplated problem is calculated as

$$\bar{\theta} = \begin{cases} (10 + 2K_u + 2K_d)N + 2 + K_u + K_d, & \alpha = 0 \\ (19 + 2K_u + 2K_d)N + 2 + K_u + K_d, & \alpha \neq 0 \end{cases}. \tag{6.31}$$

This leads to the final theorem:

Theorem 21. *The complexity of solving the optimization problem* (6.27) *in terms of Newton steps is*

$$T_{Barrier} = \mathcal{O}\left(\log\left((K_u + K_d)N/\xi\right) * \left((K_u + K_d)N + \log_2\log_2\left(1/\xi\right)\right)\right). \tag{6.32}$$

Proof. Plugging (6.31) into (6.30) and simplifying this term leads to the bound (6.32). □

6.3 Conversion Algorithms

In the previous section, it was shown that the integer-relaxed optimization problem (6.27) can be solved optimally in polynomial time. In the same manner as for the uplink-only scenario, the obtained allocation allows the assignment of fractions of resources, which breaks the natural limitation that only integer parts of RAN and edge computing resources can be allocated. Hence, also for the two-way communication scenario, specific approximation algorithms for obtaining an integer solution to the optimization problem were developed for the particular values of $\alpha = 0$, $\alpha = 1$, $\alpha = 2$, and $\alpha \to \infty$. These algorithms are related to the algorithms presented in Section 5.3 and again rely on the conversion of the continuous solution to an integer resource allocation. Subsequently, the algorithms for the specific fairness cases are introduced. Throughout the following subsections, $\boldsymbol{J}_{\{u,d\}}$ indicates the $N \times K_{\{u,d\}}$ uplink/downlink RAN allocation matrix with entries $J_{\{u,d\},ij} \in \{0, 1\}$ and $\boldsymbol{n}$ denotes the $N \times 1$ edge computing resource allocation vector with entries $n_i \in \mathbb{N} \setminus \{0\}$. $\boldsymbol{I}_{\{u,d\}}$ and $\boldsymbol{m}$ are the corresponding continuous variables. Once more, it is assumed that an admission control was performed when applying the subsequently defined approximation algorithms. The development of such an admission policy was, however, outside the scope of this thesis and is left as a future work.

The approximation algorithms for the two-way communication system model follow the same procedure as the heuristics from Section 5.3. First, the continuous edge computing allocation is converted to an integer assignment by mathematical rounding according to Algorithm 2. Again, only $L - N$ resources are assigned during the optimization of the integer-relaxed problem. Next enough uplink and downlink RAN resources are allocated such that every user fulfills its latency requirement. Finally, the remaining RAN resources are allocated with the aim of meeting the specific fairness criterion.

6.3.1 No Fairness (Throughput Maximization)

As described in Subsection 5.3.2, the objective in the no fairness case is to maximize the overall throughput in the network, which is achieved when every PRB is assigned to the user who is experiencing the best channel conditions. Based on the general allocation scheme

Algorithm 8 Resource Allocation for $\alpha = 0$ in the Uplink/Downlink Scenario

Input: N, K_u, K_d, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{I}_d$, $\boldsymbol{\Phi}_u$, $\boldsymbol{\Phi}_d$
Output: $\boldsymbol{n}$, $\boldsymbol{J}_u$, $\boldsymbol{J}_d$

function AllocA0(N, K_u, K_d, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{I}_d$, $\boldsymbol{\Phi}_u$, $\boldsymbol{\Phi}_d$)
  $\boldsymbol{n} = \text{ECRAlloc}(N, L - N, \boldsymbol{m}) + \mathbf{1}$
  $\boldsymbol{J}_u = 0$, $\boldsymbol{J}_d = 0$
  for $i = 1$ to N **do**
    Calculate $w_{\{u,d\},i} = \sum_{j=1}^{K_{\{u,d\}}} I_{\{u,d\},ij} \Phi_{\{u,d\},ij}$.
  end for
  Create list z with users i ordered s.t. $\Delta_{u,i}/w_{u,i} + \Delta_{d,i}/w_{d,i}$ is decreasing.
  while list z is non-empty **do**
    for user i in list z **do**
      for uplink u and downlink d **do**
        Find $\arg\max_j I_{\{u,d\},ij} \Phi_{\{u,d\},ij}$.
        if $\exists$ more than one j **then**
          Choose randomly between those j.
        end if
        Allocate PRB j to user i, update $\boldsymbol{J}_{\{u,d\},j}$ and set $\boldsymbol{I}_{\{u,d\},j} = \mathbf{0}$.
      end for
      Calculate delay δ_i using n_i and $\boldsymbol{J}_{u,i}$, $\boldsymbol{J}_{d,i}$.
      if $\delta_i \leq T_{max}$ **then**
        Remove user i from list z.
      end if
    end for
  end while
  for all non-allocated uplink/downlink PRBs k **do**
    Find $\arg\max_i \Phi_{\{u,d\},ik}$.
    Allocate PRB k to user i and update $\boldsymbol{J}_{\{u,d\},k}$.
  end for
  return $\boldsymbol{n}$, $\boldsymbol{J}_u$, $\boldsymbol{J}_d$
end function

introduced previously, this objective is pursued in Algorithm 8 while still all resource and delay constraints are met. The complexity of Algorithm 8 is $\mathcal{O}(N + K_u + K_d)$.

6.3.2 Proportional Fairness

In the proportional fairness case the uplink and downlink RAN resources are handled separately in the approximation algorithm once every user complies with its delay constraint. The reason for this approach is that the data rate a user is assigned in the downlink is

Algorithm 9 Resource Allocation for $\alpha = 1$ in the Uplink/Downlink Scenario

Input: N, K_u, K_d, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{I}_d$, $\boldsymbol{\Phi}_u$, $\boldsymbol{\Phi}_d$
Output: $\boldsymbol{n}$, $\boldsymbol{J}_u$, $\boldsymbol{J}_d$

1: **function** AllocA1(N, K_u, K_d, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{I}_d$, $\boldsymbol{\Phi}_u$, $\boldsymbol{\Phi}_d$)
2: Follow lines 2 to 22 from Alg. 8.
3: **for** $i = 1$ to N **do**
4: Calculate $w_{\{u,d\},i} = \sum_{j=1}^{K_{\{u,d\}}} J_{\{u,d\},ij}$.
5: **end for**
6: Create lists $z_{\{u,d\}}$ with users i ordered s.t. $w_{\{u,d\},i}$ is increasing.
7: **for all** non-allocated uplink/downlink PRBs k **do**
8: Take $z_{\{u,d\}}(1)$, find $\arg\min_k \left(\max_i \left(\Phi_{\{u,d\},ik} \right) - \Phi_{\{u,d\},z_{\{u,d\}}(1)k} \right)$.
9: Allocate PRB k to user $z_{\{u,d\}}(1)$ and update $\boldsymbol{J}_{\{u,d\},k}$.
10: Set $w_{\{u,d\},z_{\{u,d\}}(1)} = \sum_{j=1}^{K_{\{u,d\}}} J_{\{u,d\},z_{\{u,d\}}(1)j}$.
11: Reorder list z with users i s.t. $w_{\{u,d\},i}$ is increasing.
12: **end for**
13: **return** $\boldsymbol{n}$, $\boldsymbol{J}_u$, $\boldsymbol{J}_d$
14: **end function**

independent of the data rate a user is assigned in the uplink once the latency requirement is fulfilled. Since the objective is a sum of the logarithms of these two data rates, the aim of the designed heuristic is to give an equal amount of uplink or downlink resources to every user and consider the two links as two separated networks. The followed strategy is detailed specified in Algorithm 9 with its complexity being $\mathcal{O}(N + K_u + K_d)$.

6.3.3 Delay Minimization

In the same manner as in Subsection 5.3.4, due to the lack of the knowledge of all allocation combinations, the objective of the proposed approximation algorithm is to minimize the maximum delay any user is experiencing after fulfilling all delay constraints. However, this strategy is followed independently for the uplink RAN and the downlink RAN, as the original goal of delay minimization is to minimize the overall delay present in the system. Because the reciprocals of the uplink and downlink data rates are again added in the objective function, the assignments in one network part do not influence the objective value from the other network part, which is the reason why the two links can be considered separately. The resulting approximation algorithm is summarized in Algorithm 10. Its complexity is once more $\mathcal{O}(N + K_u + K_d)$.

Algorithm 10 Resource Allocation for $\alpha = 2$ in the Uplink/Downlink Scenario

Input: N, K_u, K_d, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{I}_d$, $\boldsymbol{\Phi}_u$, $\boldsymbol{\Phi}_d$
Output: $\boldsymbol{n}$, $\boldsymbol{J}_u$, $\boldsymbol{J}_d$

1: **function** AllocA2(N, K_u, K_d, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{I}_d$, $\boldsymbol{\Phi}_u$, $\boldsymbol{\Phi}_d$)
2: Follow lines 2 to 22 from Alg. 8.
3: **for** $i = 1$ to N **do**
4: Calculate $w_{\{u,d\},i} = \sum_{j=1}^{K_{\{u,d\}}} J_{\{u,d\},ij} \Phi_{\{u,d\},ij}$.
5: **end for**
6: Create lists $z_{\{u,d\}}$ with users i ordered s.t. $w_{\{u,d\},i}$ is increasing.
7: **for all** non-allocated uplink/downlink PRBs k **do**
8: Take $z_{\{u,d\}}(1)$, find $\arg\min_k \left(\max_i \left(\Phi_{\{u,d\},ik} \right) - \Phi_{\{u,d\},z_{\{u,d\}}(1)k} \right)$.
9: Allocate PRB k to user $z_{\{u,d\}}(1)$ and update $\boldsymbol{J}_{\{u,d\},k}$.
10: Set $w_{\{u,d\},z_{\{u,d\}}(1)} = \sum_{j=1}^{K_{\{u,d\}}} J_{\{u,d\},z_{\{u,d\}}(1)j} \Phi_{\{u,d\},z_{\{u,d\}}(1)j}$.
11: Reorder list $z_{\{u,d\}}$ with users i s.t. $w_{\{u,d\},i}$ is increasing.
12: **end for**
13: **return** $\boldsymbol{n}$, $\boldsymbol{J}_u$, $\boldsymbol{J}_d$
14: **end function**

6.3.4 Max-Min Fairness

Lastly, also in the max-min fairness scenario, i.e., $\alpha \to \infty$, the remaining PRBs in the uplink and downlink RAN are distributed separately from each other. The motivation for this approach is again the mathematical character of the objective function, meaning that the uplink and downlink parts contribute independently to the overall utility because they are connected via a sum. Redistributions of blocks only affect the downlink/uplink data rates of other users, so both the downlink and the uplink can be considered as a self-contained system once the latency requirements are fulfilled. The corresponding heuristic following this scheme is presented in Algorithm 11 with its complexity given as $\mathcal{O}(N + K_u + K_d)$.

6.4 Performance Evaluation

Applying the adjusted approximation algorithms from Section 6.3 in simulations, their performance is evaluated in the following. First, the simulation setup is shortly recapitulated and adapted to the two-way communication scenario. The benchmark allocation procedure follows the same principle as in Chapter 5 with the only difference being that a downlink PRB is assigned in each iteration as well. Hence, an extended description including an algorithm is omitted here. Subsections 6.4.2 to 6.4.5 of this chapter address the performance of the heuristics for the specific fairness cases.

Algorithm 11 Resource Allocation for $\alpha \to \infty$ in the Uplink/Downlink Scenario

Input: N, K_u, K_d, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{I}_d$, $\boldsymbol{\Phi}_u$, $\boldsymbol{\Phi}_d$
Output: $\boldsymbol{n}$, $\boldsymbol{J}_u$, $\boldsymbol{J}_d$

1: **function** AllocAInf(N, K_u, K_d, L, $\boldsymbol{m}$, $\boldsymbol{I}_u$, $\boldsymbol{I}_d$, $\boldsymbol{\Phi}_u$, $\boldsymbol{\Phi}_d$)
2: Follow lines 2 to 22 from Alg. 8.
3: **for** $i = 1$ to N **do**
4: Calculate $w_{\{u,d\},i} = \left(\sum_{j=1}^{K_{\{u,d\}}} J_{\{u,d\},ij}\Phi_{\{u,d\},ij}\right)^{|1-\alpha|}$.
5: **end for**
6: Create lists $z_{\{u,d\}}$ with users i ordered s.t. $w_{\{u,d\},i}$ is increasing.
7: **for all** non-allocated uplink/downlink PRBs k **do**
8: Take $z_{\{u,d\}}(1)$, find $\arg\min_k \left(\max_i \left(\Phi_{\{u,d\},ik}\right) - \Phi_{\{u,d\},z_{\{u,d\}}(1)k}\right)$.
9: Allocate PRB k to user $z_{\{u,d\}}(1)$ and update $\boldsymbol{J}_{\{u,d\},k}$.
10: Set $w_{\{u,d\},z_{\{u,d\}}(1)} = \left(\sum_{j=1}^{K_{\{u,d\}}} J_{\{u,d\},z_{\{u,d\}}(1)j}\Phi_{\{u,d\},z_{\{u,d\}}(1),j}\right)^{|1-\alpha|}$.
11: Reorder list $z_{\{u,d\}}$ with users i s.t. $w_{\{u,d\},i}$ is increasing.
12: **end for**
13: **return** $\boldsymbol{n}$, $\boldsymbol{J}_u$, $\boldsymbol{J}_d$
14: **end function**

6.4.1 Simulation Setup

The 5G trace with data measured in the Republic of Ireland [RLSQ20] was again used for the input CQI values with the corresponding data rates given in Table 4.1. The same method as described in Subsection 5.4.1 was used to obtain the per-block rates of every user from the measurements. It is assumed that a user is experiencing the same channel conditions in the uplink and downlink, as the difference in the user's position and the time-dependence of the channel is negligible due to the strict latency requirement. The subcarrier spacing is again 30 kHz, such that the PRB width is 360 kHz and a frame consists of 20 slots [ETS22b]. The total number of uplink PRBs is $K_u = 80$ and the number of downlink PRBs is $K_d = 100$. Once more, at least 1 slot and at most 6, 10, or 20 slots are allocated to each user in the uplink and downlink, respectively, which follows from the latency requirement and the duration of one slot (0.5 ms). The number of edge computing resources is $L = 120$ and the processing rate per resource is 500 kbps. The size of the data sent from the user to the BS is 6 kbit, while the size of the packets sent from the BS to the users is 4 kbit. For all types of fairness, simulation data were again gathered for $N = \{5, 8, 10\}$ and for $T_{max} = \{3, 5, 10\}$ ms using MATLAB R2021b together with CVX [GB14, GB08] and the Mosek optimizer [MOS22].

6.4.2 Results for No Fairness (Throughput Maximization)

The same evaluation plots as presented in Section 5.4 are presented in this section. Figures 6.1 to 6.3 show the results for the no fairness, i.e., throughput maximization, case. Thereby (and also in the subsequent plots), *RG* denotes again the relative gap that was set when solving the integer optimization problem. In Figures 6.1 and 6.3 it is discernable that the approximation algorithm outperforms the benchmark by far and is very close to the integer and continuous (average) optimum, indicating the very good performance of the algorithm. The averages in Figure 6.3 and all following average plots are again taken over 100 measurement points. When looking at the average objective values for a specific number of users, it is observable that the average value decreases when tightening the delay constraint. This can be explained with the allocation of more PRBs to users that are experiencing worse channel conditions, which is needed to fulfill their latency requirements. In Figure 6.2 the deviation of the objective value acquired from the heuristic to the integer and continuous optimum is shown in percent. Some deviations to the integer optimum are negative, indicating that the heuristic obtained a better result than the integer optimum. Of course, this is not possible and the reason for this observation is the NP-hardness of the integer optimization problem, which sometimes makes it impossible to solve the optimization with a high enough accuracy. When comparing the objective values from the approximation algorithm and the continuous optima, the maximum deviation that can be observed among 100 data points is 1.79 %, while the average deviation is 0.47 %. This proves the very good performance of the approximation algorithm.

6.4.3 Results for Proportional Fairness

The results for the proportional fairness case are shown in Figures 6.4 to 6.6. Looking at the average objective value from 100 measurement points and at the single measurements for different CQI inputs in Figure 6.6 and Figure 6.4, respectively, it is observable that the benchmark algorithm is outperformed by the approximation algorithm. Due to the similarity of the objectives of the Round-Robin principle and the proportional fairness, i.e., providing every user with the same amount of resources, the results from the benchmark are much closer to the heuristic objective values and the optima than for $\alpha = 0$. However, since the Round-Robin algorithm neglects the per-PRB rates of each user, it shows a poorer performance than the specific algorithm. Another observation from Figure 6.6 is that the objective values do not depend on the delay constraint but rather on the number of users that are present in the network. This was already perceived for the uplink-only scenario and can be explained with the gradient characteristics of the natural logarithm, cf. Subsection 5.4.4. Finally, from Figure 6.5, it is recognizable that the approximation algorithm exhibits an excellent performance. The maximum deviation to the integer optimum among 100 data points is 0.39 %, whereas the average deviation is only 0.27 %.

6.4.4 Results for Delay Minimization

When looking at the results for the delay minimization that are shown in Figures 6.7 to 6.9, it is observable that the plots are very similar to the uplink-only scenario. Specifically, the heuristic once more outperforms the benchmark Round-Robin algorithm and the results are very close to the integer optimum. The inverse dependence of the objective value on the number of users already mentioned in Subsection 5.4.5 can also be identified for the present scenario in Figure 6.9. For the measurements with IDs 15 and 16 the performance is again poorer than compared to the average results, with the maximum deviation of the heuristic objective value from the integer optimum being 11.69 % among 100 data points, while the average is only 1.46 %. The reason is the presence of a user experiencing very bad channel conditions compared to all other users, which has a considerable impact due to the mismatch between the original goal of overall delay minimization and the objective of the approximation algorithm introduced in Subsection 6.3.3. It is noticeable that the impact gets smaller the higher the number of users in the network is. Although these outliers can be detected, the overall performance of the approximation is still very good.

6.4.5 Results for Max-Min Fairness

Finally, also for the max-min fairness, satisfying evaluation results are given in Figures 6.10 to 6.12. For the two-way communication scenario, the highest possible α that allowed for acceptable simulation outcomes was $\alpha = 12$. The benchmark allocation scheme performs very poor in the presence of a user with bad channel conditions, which can be seen in Figure 6.10 for the measurements 15 and 16. Additionally, in Figure 6.11, it is observable that also the approximation shows a much larger deviation for these measurements than for other inputs. Moreover, the deviation is much larger than for the uplink-only scenario, cf. Figure 5.11. This observation is caused by two different factors: The first reason is that the minimum number of RAN resources for scenario 1 is 120, whereas it is 80 for scenario 2. Thus, the minimum data rate that any user is experiencing is bigger for the first scenario, such that the negative influence on the objective is smaller. The second reason is that $\alpha = 12$ is only an approximation of $\alpha \to \infty$. Therefore, the presented results are only an approximation of the max-min fairness. When comparing the minimum data rate that any user is undergoing in the optimal continuous solution and in the solution from the heuristic, it is perceivable that this data rate is larger in the solution from the heuristic. Thus, the heuristic actually provides a better solution in the sense of max-min fairness. However, since $\alpha = 12$ is not equal to max-min fairness, there are allocation scenarios where the minimum data rate a user is experiencing is worse than observed in the heuristic solution but the overall objective value is still better. Concluding, this implies that the degraded performance of the approximation algorithm is only detectable due to the approximation of $\alpha \to \infty$ and will diminish the greater α gets. Besides these studies, the results shown in Figure 6.12 are similar to the uplink-only scenario. Due to the inverse dependence of the objective to the data rate of each user, the overall objective value gets smaller the higher

Table 6.1: Maximum and average deviation of the approximation algorithm objective values from the continuous/integer optimum among 100 data points in the uplink/downlink scenario

α	0	1	2	12
max. dev. from int. opt. in %	—	0.39	11.69	—
avg. dev. from int. opt. in %	—	0.27	1.46	—
max. dev. from cont. opt. in %	1.79	0.39	11.75	7.85
avg. dev. from cont. opt. in %	0.47	0.28	1.58	0.10

the number of users is. Even though the maximum deviation from the heuristic to the continuous optimum among 100 data points is 7.85 %, the average deviation is only 0.1 %, which once more certifies the excellent performance of the approximation algorithm.

6.5 Summary

Jointly allocating RAN and processing resources to vehicular users so that their delay constraints are met, while simultaneously providing certain types of fairness, was considered in this chapter. The contemplated scenario was a two-way communication scenario, implying that the users sent data to the BS, where the data was processed, and a response packet was sent back to each user. Thus, a full round-trip was considered. The integer-relaxed allocation optimization problem for this scenario is still solvable in polynomial time, and the approximation algorithms with polynomial-time complexity introduced in Section 5.3 could be adjusted for all types of fairness. The performance of the heuristics is very close to the optimum for all cases. Conclusively, the key results regarding the deviation of the objective values from the integer and continuous optimal values are given in Table 6.1.

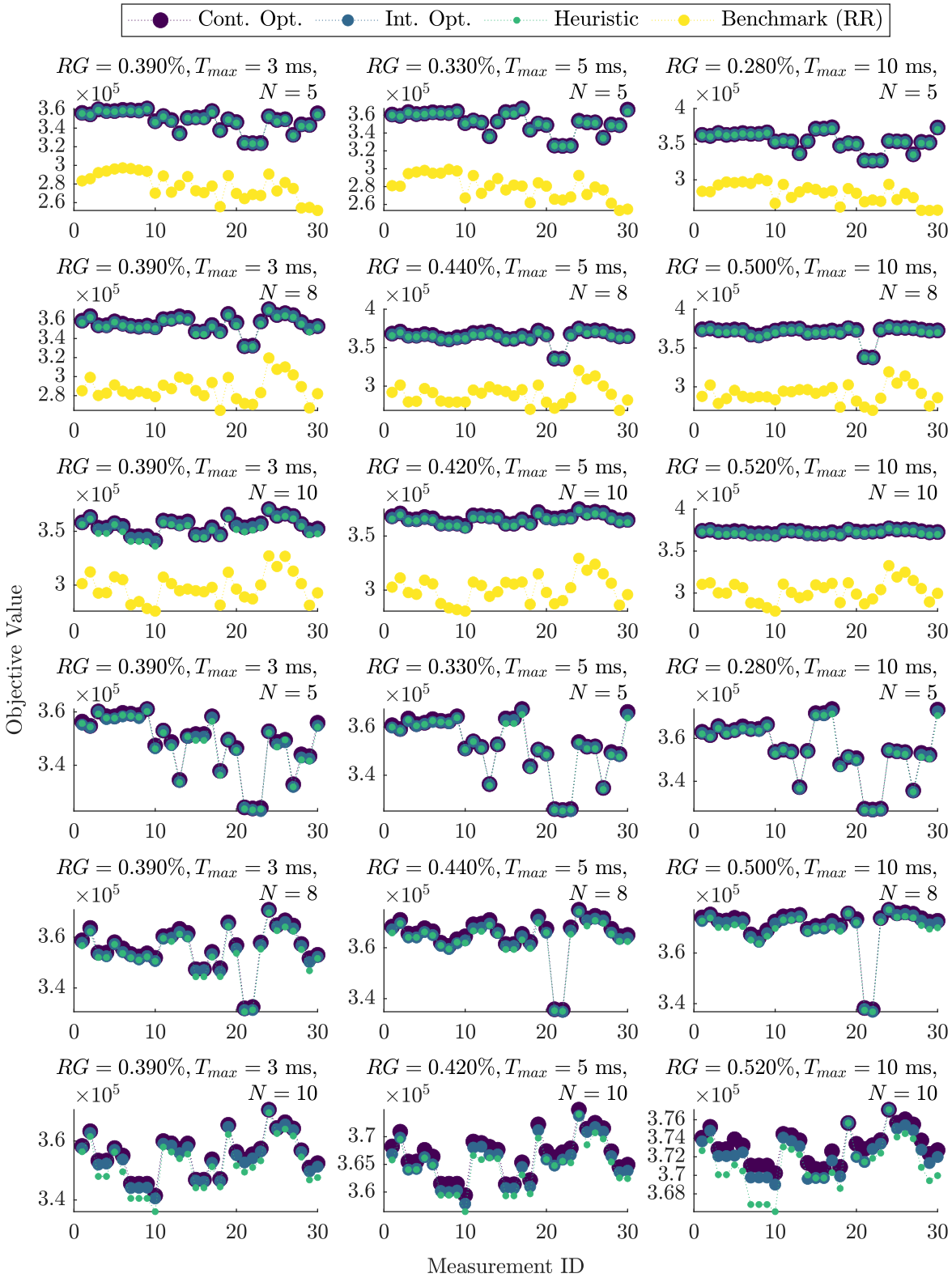


Figure 6.1: Obj. values for various CQI inputs for $\alpha = 0$ in the up-/downlink scenario.

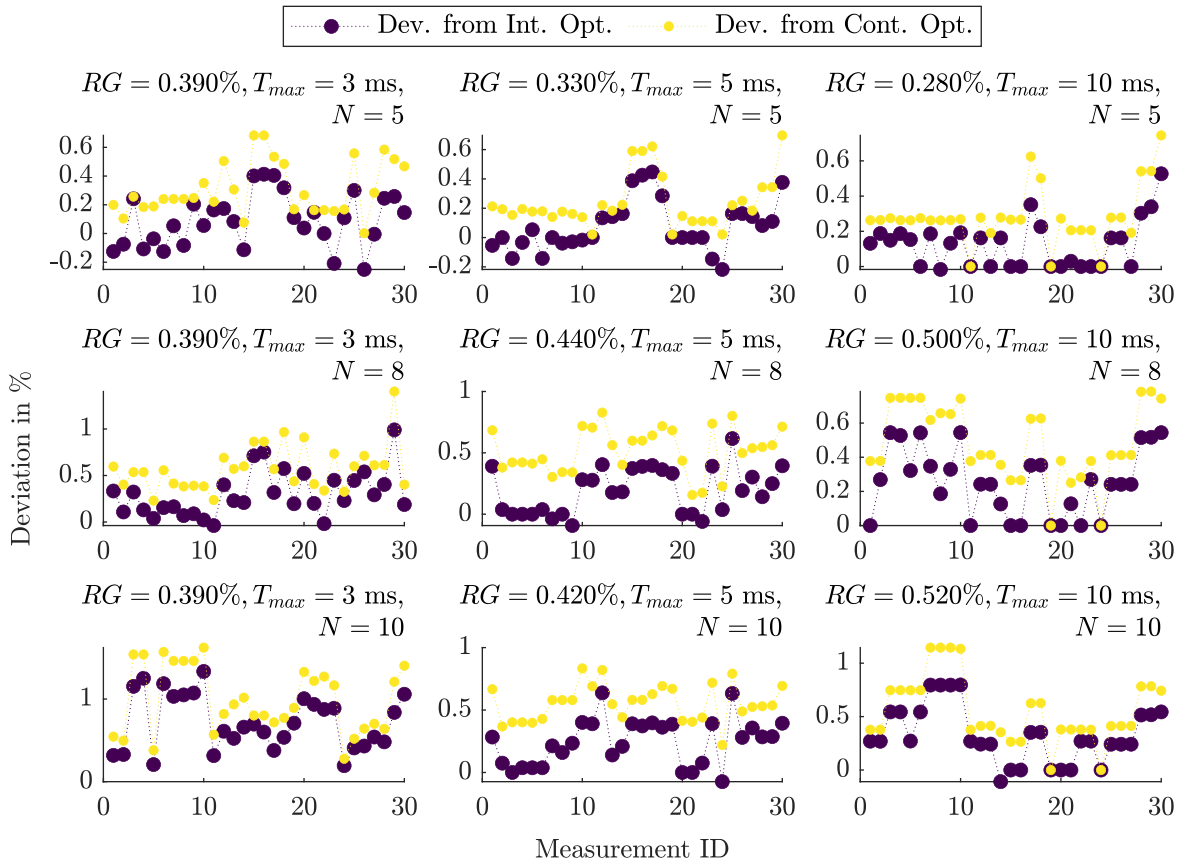


Figure 6.2: Deviation of the heuristic solution from the continuous/integer optimum for $\alpha = 0$ in the up-/downlink scenario.

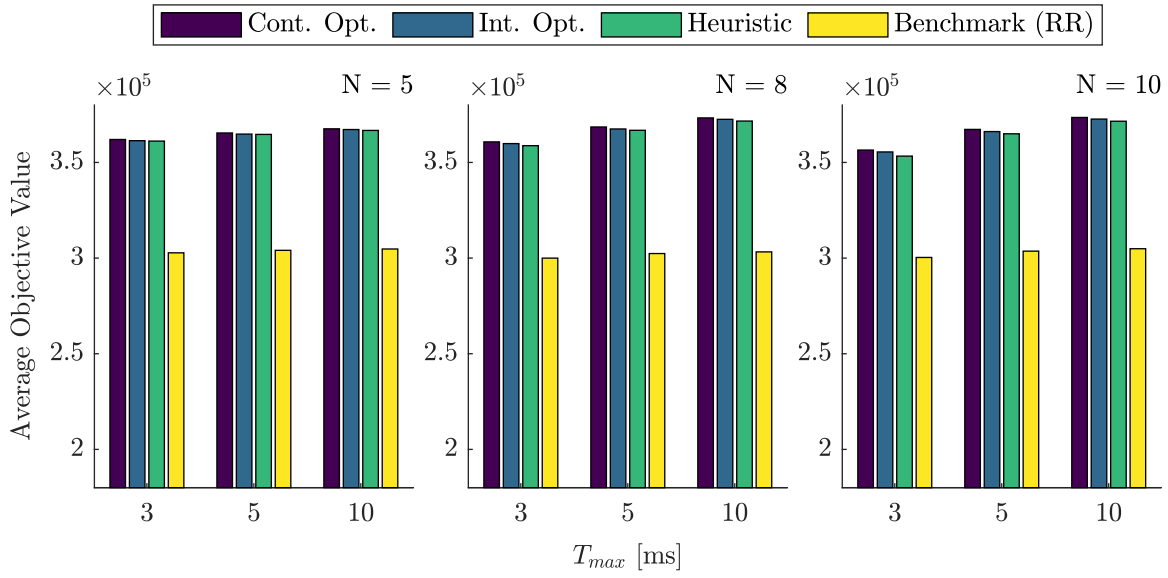


Figure 6.3: Average objective values for $\alpha = 0$ in the up-/downlink scenario.

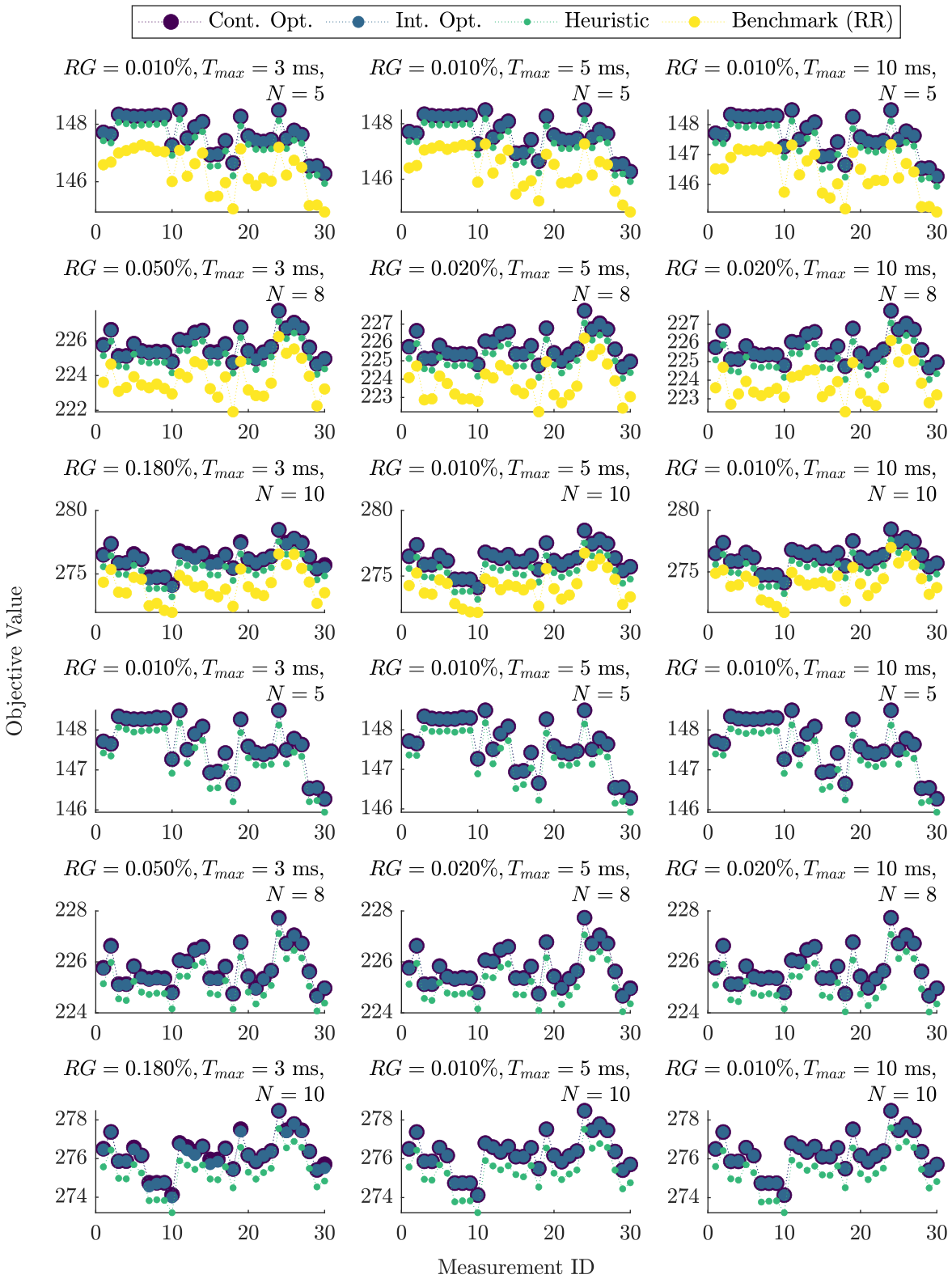


Figure 6.4: Obj. values for various CQI inputs for $\alpha = 1$ in the up-/downlink scenario.

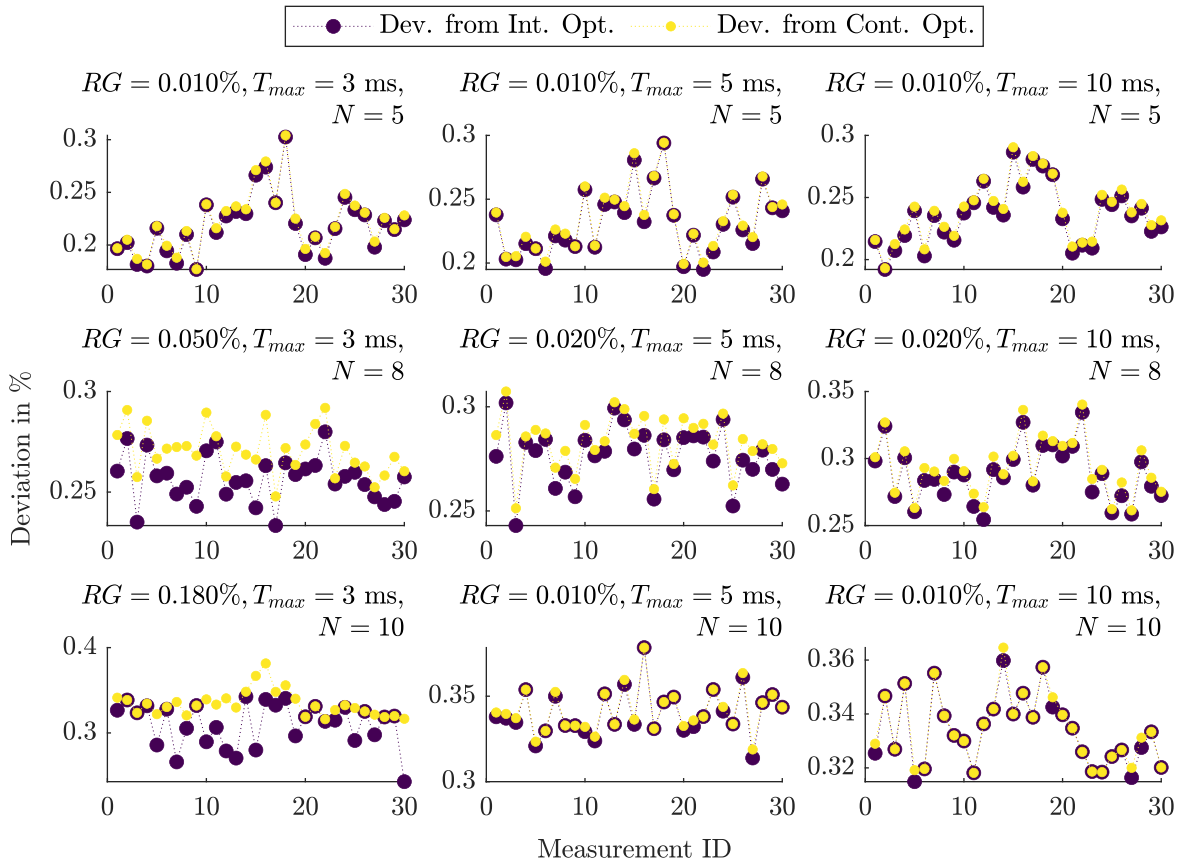


Figure 6.5: Deviation of the heuristic solution from the continuous/integer optimum for $\alpha = 1$ in the up-/downlink scenario.

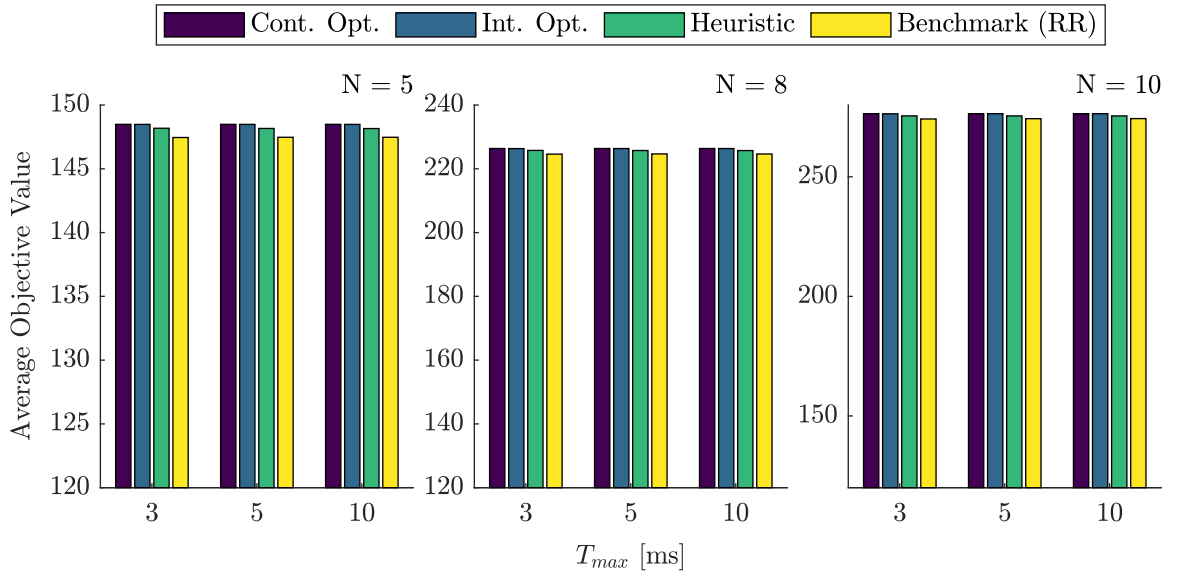


Figure 6.6: Average objective values for $\alpha = 1$ in the up-/downlink scenario.

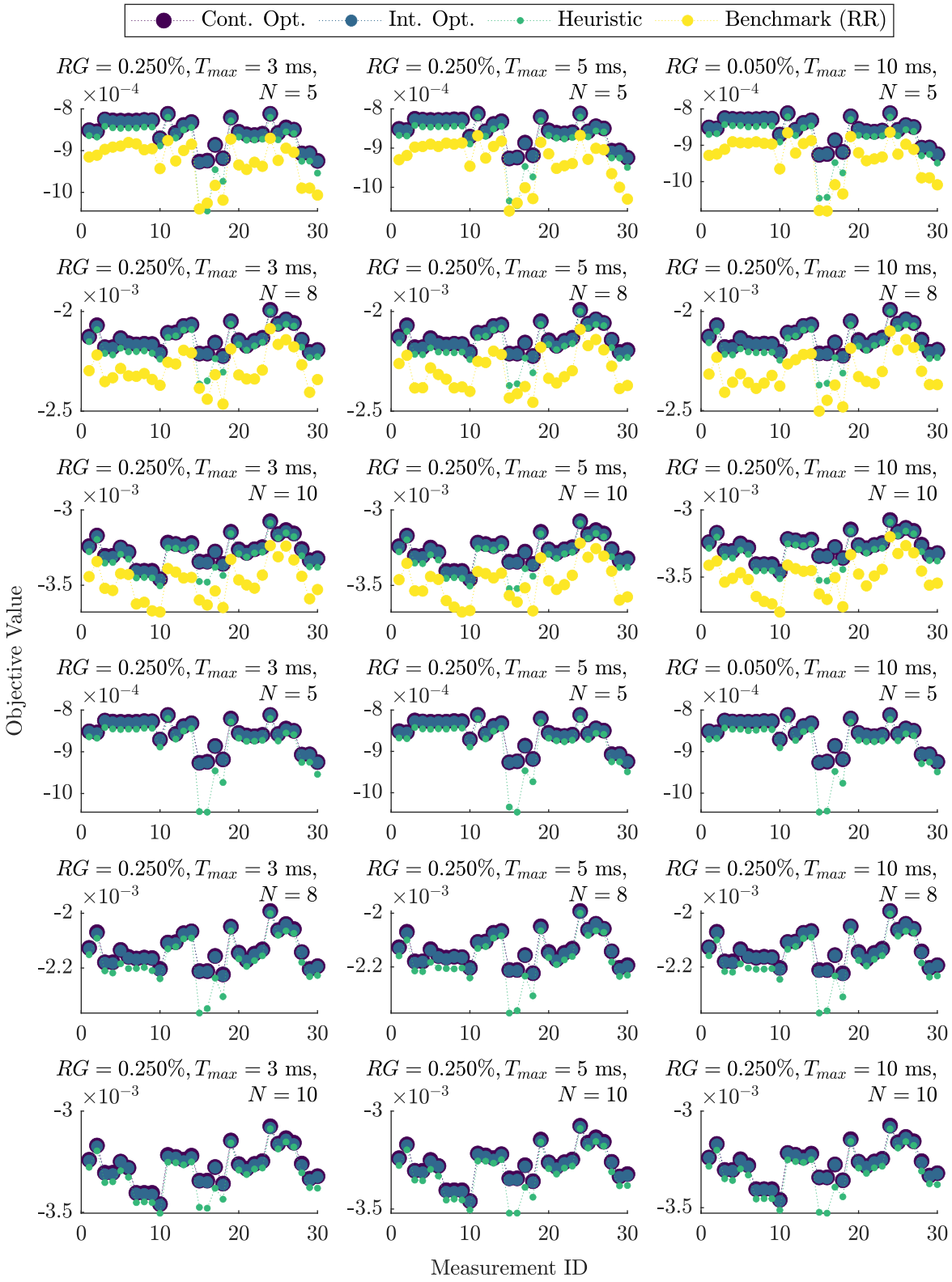


Figure 6.7: Obj. values for various CQI inputs for $\alpha = 2$ in the up-/downlink scenario.

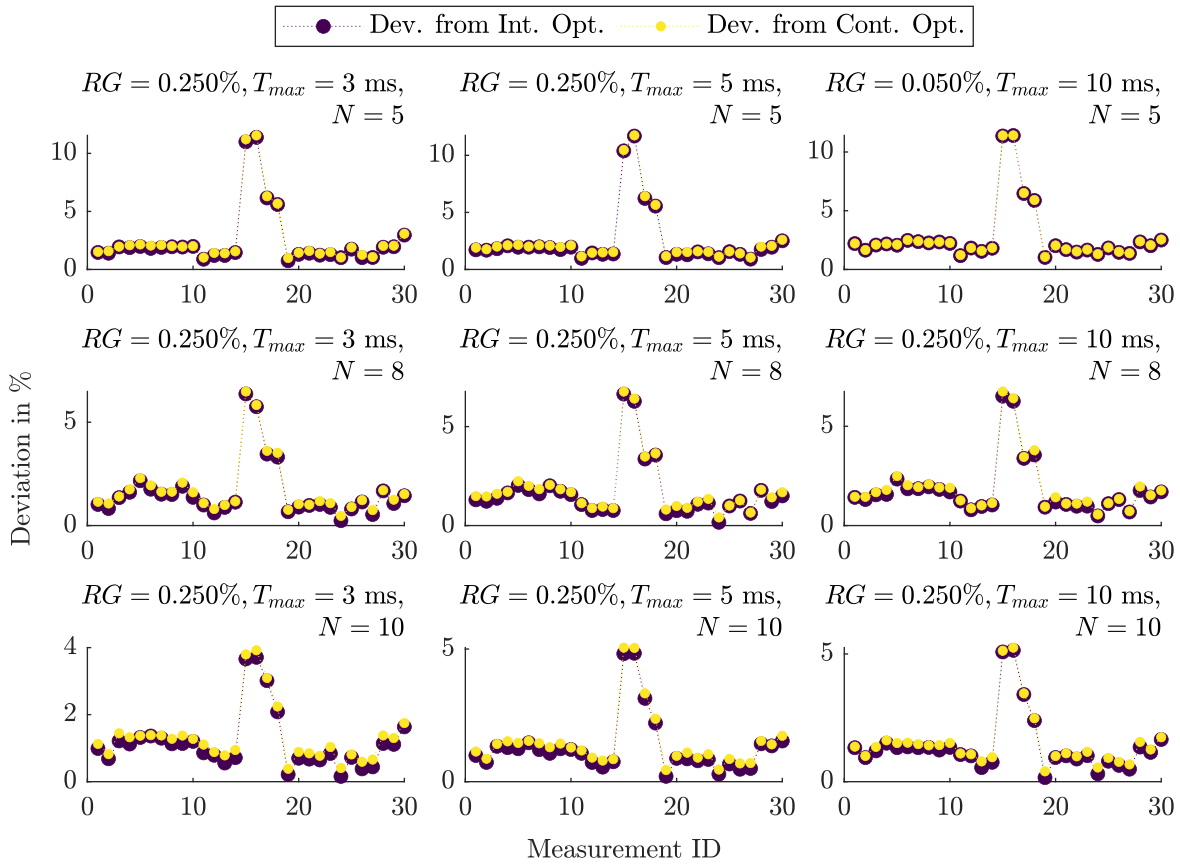


Figure 6.8: Deviation of the heuristic solution from the continuous/integer optimum for $\alpha = 2$ in the up-/downlink scenario.

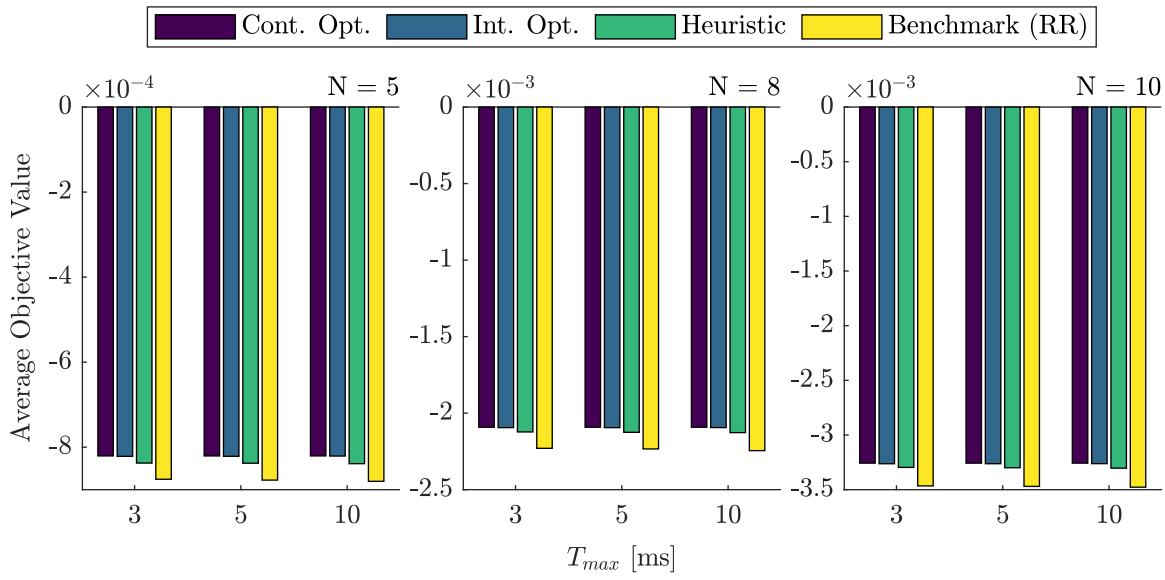


Figure 6.9: Average objective values for $\alpha = 2$ in the up-/downlink scenario.

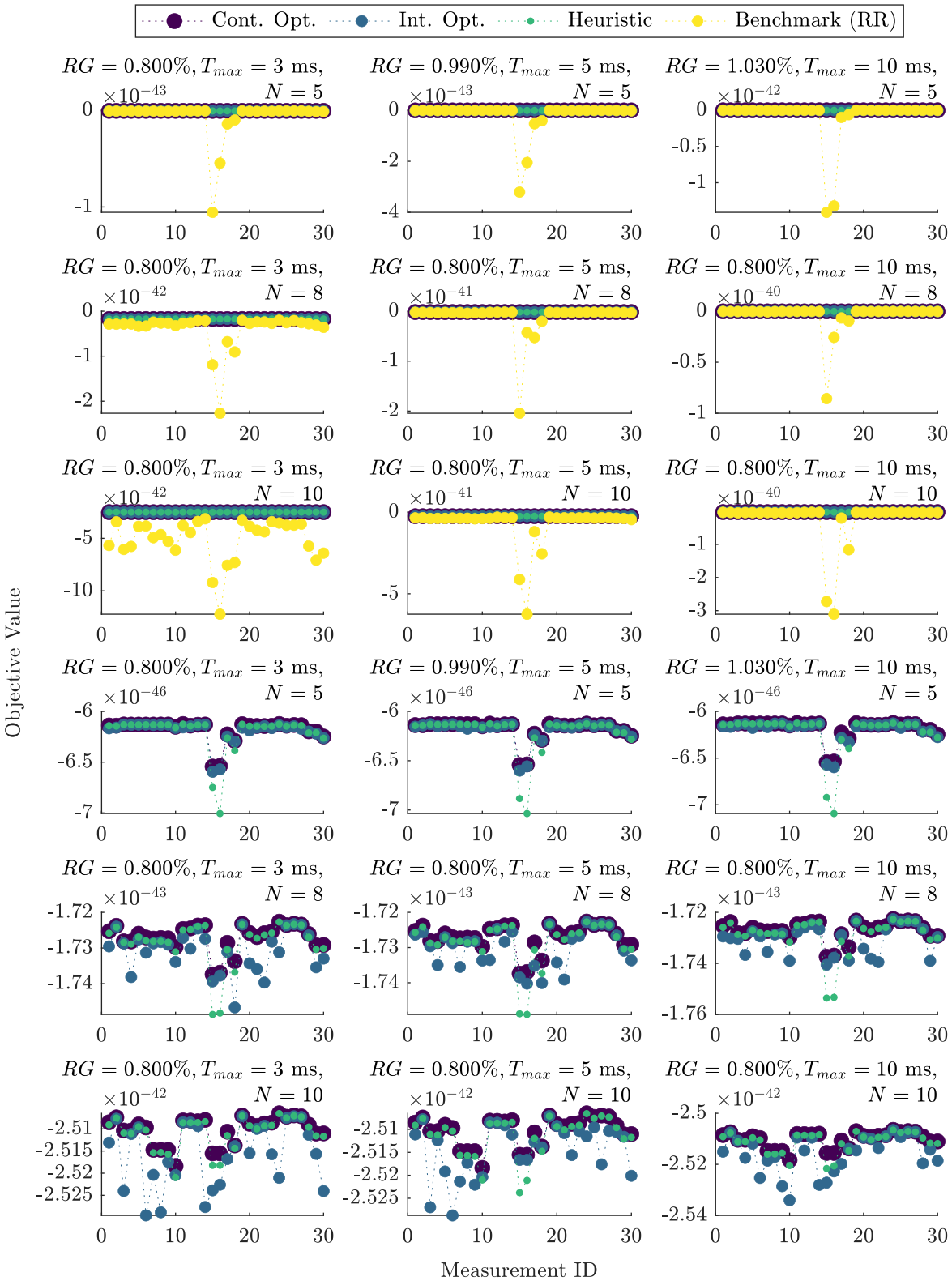


Figure 6.10: Obj. values for various CQI inputs for $\alpha = 12$ in the up-/downlink scenario.

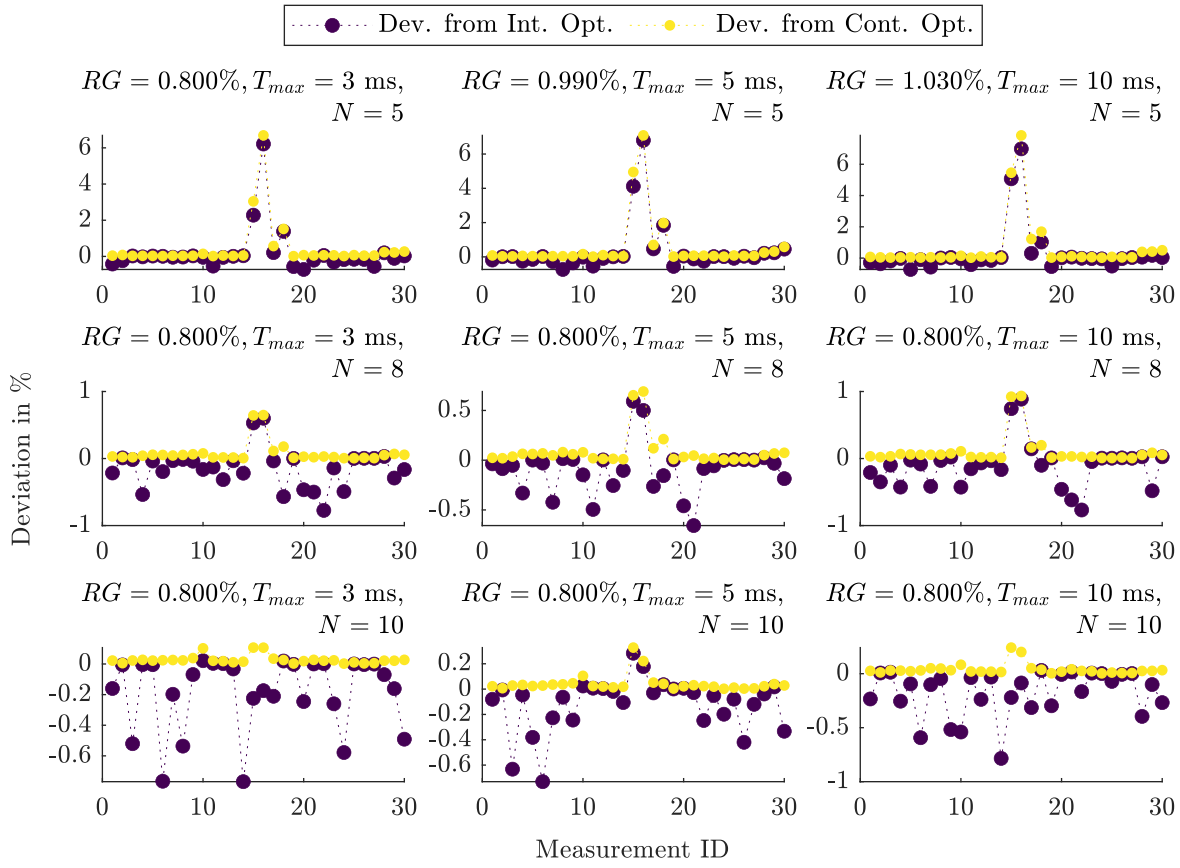


Figure 6.11: Deviation of the heuristic solution from the continuous/integer optimum for $\alpha = 12$ in the up-/downlink scenario.

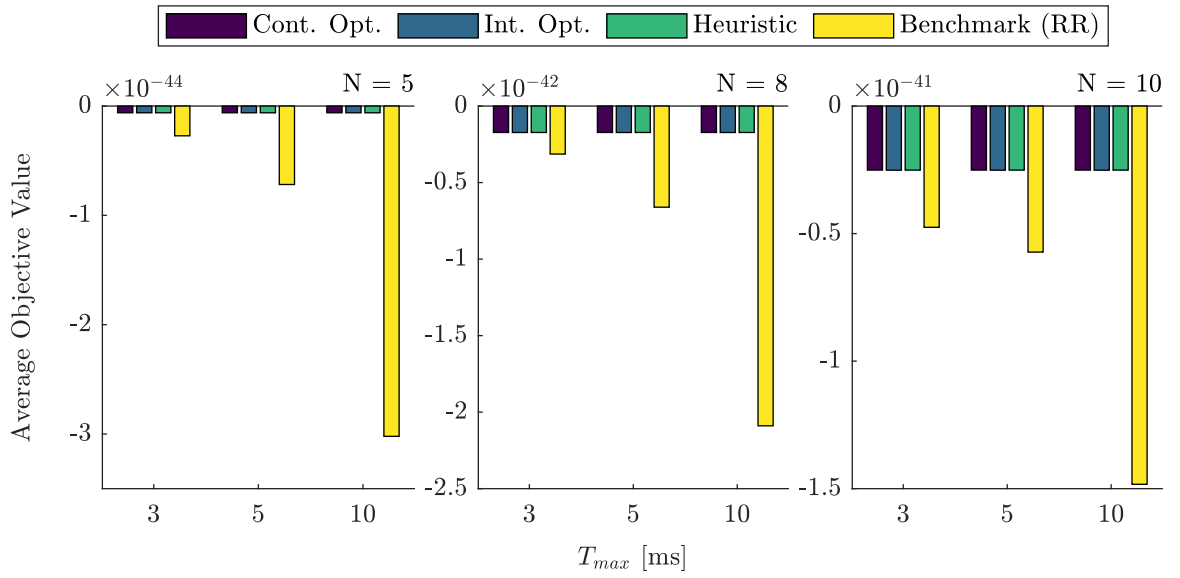


Figure 6.12: Average objective values for $\alpha = 12$ in the up-/downlink scenario.

7 Comparison: Mobile Edge versus Centralized Cloud Computing

Lastly, after introducing a MEC system and developing resource allocation schemes that provide fairness among the users, this MEC system is compared to a CCC system employed by an automotive OEM. To this end, real data from vehicles was gathered that are explained in more detail in the following. Eventually, the overall delays achieved with the CCC system are compared to the latencies experienced using the devised MEC system. The CCC system for vehicular users requesting a service is depicted in Figure 7.1.

7.1 Detailed Data Analysis

Two different types of data were gathered to model the CCC system, as data for the entire process of transmitting a service request, processing this request, and receiving a response were not available. On the one hand, the average round trip ping duration from a vehicle located anywhere in Europe, the Middle East, or Africa (EMEA) to the CCC servers was evaluated to get insights into the transmission delays. On the other hand, the processing times of two different services provided to the vehicular users were assessed to exemplarily assess the processing times of URLLC services. The following two subsections present a detailed analysis of these data.

7.1.1 Ping Durations

For the ping durations, 99 average ping duration values of various vehicular users were collected for 20 consecutive days. These average values are automatically and periodically reported and saved in one of the OEM's databases. One average value is thereby taken over 5 single measurements, where the 5 measurements stem from a single vehicle. The roundtrip ping duration is determined by a simple ping test conducted automatically and periodically by the vehicle, where the answering entity is a proxy server that is upstream of the actual processing server. A violin plot of the 99 average values is depicted for every day in Figure 7.2, where it is marked whether the values were collected on a weekday (Monday to Friday) or on the weekend (Saturday or Sunday). First, it can be noticed that there is no difference between the days, i.e., the violin plots have the same shape for every

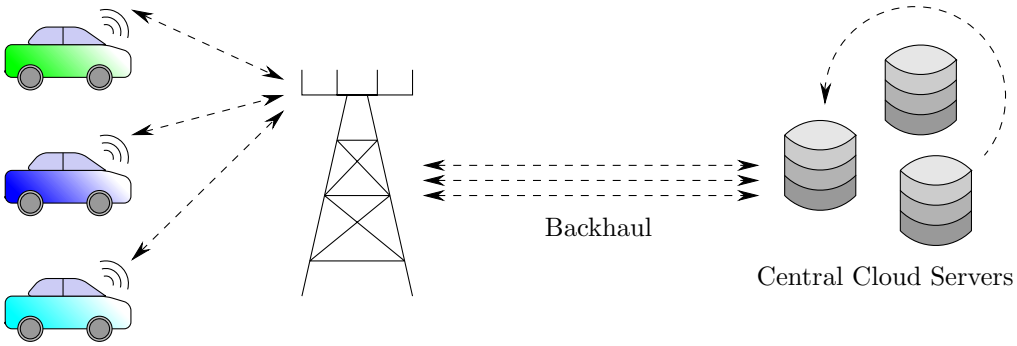


Figure 7.1: Illustration of the centralized cloud computing scenario.

Table 7.1: Mean, median and standard deviation (in ms) of the average ping duration measurement values per day

Day	1	2	3	4	5	6	7	8	9	10
Mean	80.41	79.22	86.17	77.64	92.79	79.12	90.59	98.20	88.09	87.38
Median	67.0	71.0	72.0	68.0	72.0	65.0	72.0	74.0	71.0	67.0
Std. Dev.	57.48	41.33	54.03	54.29	82.23	62.23	68.10	105.92	63.10	70.21
Day	11	12	13	14	15	16	17	18	19	20
Mean	92.83	93.57	85.29	83.01	85.54	82.92	83.17	75.40	85.88	70.65
Median	75.0	69.0	70.0	72.0	72.0	69.0	66.0	65.0	68.0	66.0
Std. Dev.	78.77	80.67	63.31	53.99	64.53	53.36	57.04	53.35	75.91	33.12

day. Additionally, it is observable that there are always some outliers, which usually reach values in the range around 400 ms. Only in three cases the outliers significantly exceed this value. The large number of outliers can be explained by the wide geographic area in which the vehicular users are located. This wide geographic area is also reflected when looking at the standard deviations of the 99 average measurement points per day given in Table 7.1. The mean and median values (see Table 7.1), are however quite stable, as they stay in the range between 70 and 99 ms or 65 and 75 ms, respectively. These comparably low values indicate a bias in the distribution of the vehicular users within the EMEA region, meaning that there are probably more users closer to the CCC server located in central Europe. Due to privacy reasons, the locations of the vehicular users corresponding to an average ping duration value were not available and could thus not be analyzed. The maximum average value of 5 measurements that can be observed is 889 ms, while the minimum value is only 20 ms, which again reflects the extent of the EMEA region. The average value of all measurement data is 84.9 ms, the median of the data is 70 ms, and the standard deviation is 65.5 ms, indicating once more the bias in the distribution of the users in the contemplated geographic area.

7.1.2 Processing Rates

The cloud server structure consists of an upstream proxy server and processing servers. The proxy server is distributing the service requests to the actual processing servers where the requests are handled. Therefore, values for two different time spans were gathered: Firstly, the pure processing times of the processing servers that actually process the service requests, and secondly, the time span accounting for the entire time period between the reception and the response of the service request at the proxy server. Additionally, the data sizes of the requests associated with these processing times were collected. For 20 consecutive days, data were gathered for a time span of one hour each. As the smallest resolution in time was 30 s, it could happen that the average of multiple requests constitutes one data point, as multiple requests can occur during these 30 s. Hence, for every day, 120 data points (processing rates) were calculated by dividing the data sizes by the processing times in the optimal case. However, since it can happen that a service is not requested during a 30 s interval, it could be that less than 120 data points were calculated per day. The processing rates must be interpreted such that they describe the rate of handling a service request of one single user, i.e., every data point is the processing rate experienced by one single user.

Two different services were contemplated during the data acquisition. The first one is called hybrid voice dialog (HVD) and enables an intelligent personal assistant service like Amazon's Alexa or Apple's Siri. The second service is called secure time (ST) and provides a monitored time base in a secure manner, which is essential for the operation of the vehicle. Figures 7.3 to 7.6 depict violin plots with the processing rates per day for both services as well as for both time spans, i.e., the pure processing time as well as the entire time span accounting for the forwarding and processing. Thereby, in the title of each violin plot, WE denotes that this day was on the weekend, and WD denotes that this day was a workday. The number in brackets denotes the number of 30 s intervals in which at least one service request was received.

Multiple observations can be made from these plots: First, it can be seen that the pure processing rates are much larger than the processing rates that include the forwarding of the requests. This is an apparent observation, as the time spans including the forwarding are of course larger than the time span accounting for the pure processing of the data. Secondly, both for the HVD and for the ST service, the pure processing rates on day 7 are much lower than for all other days. Additionally, large parts of the processing rates on day 20 are very close to 0. When checking the validity of these data, it became clear that the reason for these values is a database failure. Thus, these values (days) will not be considered any further or when calculating statistical parameter values of the entire data.

When comparing the HVD service with the ST service, it is noticeable that the number of 30 s intervals where at least one request was received is much larger for the ST service than for the HVD service. The reason for these incidences is the character of the services. The HVD service is a human triggered service, whereas the ST service is an automated service

that is requested periodically by the vehicle. Additionally, due to the dependency of the HVD service on humans, the number of 30 s intervals with at least one request is lower for weekend days than for workdays for the HVD service, as more people use their car during the week. Analyzing the pure processing rates of the HVD and the ST service, it is observable that the average processing rates of the HVD service request are much bigger than the processing rates of the ST service, cf. Tables 7.2 and 7.4. Note the different units of the two tables. The reason for this deviation is a configuration of the servers, as the HVD service is more latency critical than the ST service. However, when looking at the processing rates calculated including the forwarding (see Tables 7.3 and 7.5), it is discernable that the processing rates are of the same order for both services. This reveals the large influence of the forwarding overhead of a service request.

Looking at the shapes of the violins (related to the units) and the values for the standard deviations for both the pure processing rates as well as the processing rates including forwarding, it is perceptible that the processing rates for the ST service are much more stable than for the HVD service. The standard deviations of all HVD data are 9.14 Mbps or 62.89 kbps for the pure processing time and the processing time including forwarding, respectively, while they are $0.25 \cdot 10^5$ bps or 0.67 kbps for the ST service. A reason for this difference could be that the ST service characteristics are always the same and predictable. In contrast, the HVD service depends highly on the actual spoken request by the human, which explains the large variations in the processing time of this service.

Lastly, when looking at the outliers, for the HVD service, a couple of outliers can be observed in the positive direction, meaning that much higher processing rates were experienced by the user. Opposed to that, for the ST service, few outliers are discernable mainly in the negative direction, i.e., the user experienced a slightly lower processing rate. The median and mean values of the processing rates highlight this observation. While the overall mean value (8.21 Mbps) is larger than the median value (4.87 Mbps) for the HVD service for the pure processing rates, for the ST service the median ($2.48 \cdot 10^5$ bps) is almost equal to the mean value ($2.46 \cdot 10^5$ bps). Similar proportions can be recognized for the processing rates including forwarding, i.e., for the HVD service, the average of all data (36.27 kbps) is again larger than median (32.98 kbps), while for the ST service the median (24.16 kbps) is slightly larger than the average value of the dataset (24.05 kbps).

7.2 Latency Comparisons

For a comparison of the minimum achievable latencies for a URLLC service request using either the MEC or the CCC system, the entire procedure of transmitting a request, forwarding the request at the proxy server, processing the request, and transmitting a response packet is modelled for the CCC server in the following. For the transmission parts, the average ping duration value is used to model the latency, i.e., the duration of sending a packet to the proxy server and receiving a packet from the proxy server is assumed to be

Table 7.2: Mean, median and standard deviation (in Mbps) of the pure processing rates of the hybrid voice dialog service per day

Day	1	2	3	4	5	6	7	8	9	10
Mean	5.51	3.6	10.77	6.6	6.28	6.08	0.02	5.26	8.13	6.98
Median	3.2	1.83	3.94	2.95	4.22	3.78	0.02	3.6	4.55	5.02
Std. Dev.	7.51	4.99	12.61	8.18	6.41	6.05	0.03	6.3	8.0	7.43
Day	11	12	13	14	15	16	17	18	19	20
Mean	9.08	9.58	7.79	9.27	8.64	14.42	9.94	11.56	8.89	3.41
Median	4.82	7.55	6.07	6.59	4.54	12.01	6.05	8.5	7.02	0.27
Std. Dev.	9.89	8.93	6.84	9.13	10.31	12.46	12.96	10.04	6.57	7.94

Table 7.3: Mean, median and standard deviation (in kbps) of the processing rates of the hybrid voice dialog service per day when considering the entire processing time between the reception and the response of the service request at the proxy server

Day	1	2	3	4	5	6	7	8	9	10
Mean	22.43	15.29	45.92	30.81	21.96	42.66	45.41	28.75	38.65	24.51
Median	24.09	7.98	41.18	37.4	13.98	37.7	34.33	28.64	40.52	20.87
Std. Dev.	15.69	16.69	36.94	26.44	20.7	50.3	130.81	16.16	20.58	18.42
Day	11	12	13	14	15	16	17	18	19	20
Mean	35.71	34.27	64.07	33.78	40.22	45.09	34.61	51.01	37.65	30.14
Median	40.98	32.62	32.62	33.68	41.98	48.17	33.65	42.08	36.55	30.26
Std. Dev.	24.44	24.81	214.92	26.22	23.76	23.34	25.67	76.39	23.26	22.73

Table 7.4: Mean, median and standard deviation (in 10^5 bps) of the pure processing rates of the secure time service per day

Day	1	2	3	4	5	6	7	8	9	10
Mean	2.7	2.27	2.4	2.56	2.55	2.34	0.01	2.26	2.36	2.42
Median	2.69	2.26	2.41	2.54	2.53	2.35	0.01	2.26	2.36	2.43
Std. Dev.	0.17	0.11	0.13	0.14	0.14	0.14	0.0	0.14	0.13	0.23
Day	11	12	13	14	15	16	17	18	19	20
Mean	2.61	2.37	2.34	2.6	2.39	2.46	2.52	2.67	2.09	0.83
Median	2.63	2.54	2.47	2.59	2.4	2.47	2.61	2.7	2.13	0.08
Std. Dev.	0.23	0.61	0.54	0.15	0.14	0.12	0.46	0.29	0.32	1.08

$t_{CCC,t} = 84.9\,\text{ms}$. Although the ping measurements are conducted using quite small ping packets, this value gives a lower bound on the achievable minimum latency, as normal data packets with larger size would of course experience larger delays. For the latency caused by the CCC server due to forwarding and processing of the service request, the average processing rate including forwarding of the HVD service is used to model the processing

Table 7.5: Mean, median and standard deviation (in kbps) of the processing rates of the secure time service per day when considering the entire processing time between the reception and the response of the service request at the proxy server

Day	1	2	3	4	5	6	7	8	9	10
Mean	24.36	24.43	24.43	24.1	24.29	24.09	24.27	24.11	24.17	23.95
Median	24.4	24.47	24.5	24.15	24.27	24.17	24.26	24.26	24.23	24.12
Std. Dev.	0.89	0.6	0.51	0.39	0.33	0.44	0.37	1.13	0.46	0.53
Day	11	12	13	14	15	16	17	18	19	20
Mean	23.75	23.61	24.11	23.95	24.21	24.31	23.95	23.77	23.56	23.21
Median	23.92	23.82	24.15	24.06	24.26	24.32	24.04	24.0	23.78	23.63
Std. Dev.	0.68	0.86	0.34	0.48	0.54	0.42	0.44	0.63	0.81	1.35

rate for a URLLC service, as the latency requirements for the HVD service are stricter than for the ST service. The processing rate is hence assumed to be 36.27 kbps. This processing rate is roughly one order of magnitude smaller than the processing rate that was assumed for the MEC system. Opposed to that, the average of the pure processing rate of the HVD service (8.21 Mbps) is roughly one order of magnitude larger than the processing rates of the MEC system. Naturally, due to the high amount of available computing resources, the experienced processing rate when using the CCC server is larger than for the MEC system. However, due to the forwarding overhead and the high amount of incoming service requests, the overall experienced processing rate from the CCC server is smaller than the processing rate experienced when using the MEC system. With the packet sizes assumed to be 6 kbit and 4 kbit in the uplink and downlink respectively, the latency caused by the CCC server is on average equal to

$$t_{CCC,p} = \frac{6\,\text{kbit} + 4\,\text{kbit}}{36.27\,\text{kbps}} = 275.71\,\text{ms}. \tag{7.1}$$

Adding up the transmission and processing delays, the average overall delay that a service request experiences when it is served by the CCC server is equal to

$$t_{CCC,tot} = t_{CCS,p} + t_{CCS,t} = 275.71\,\text{ms} + 84.9\,\text{ms} = 360.61\,\text{ms}. \tag{7.2}$$

Finally, this leads to the conclusion that the latencies achievable with the CCC system are two orders of magnitude larger than the latencies achievable with the MEC system. This large overhead has two main reasons. First, the transmission latencies are much larger, as the CCC server is a lot farther away from the vehicular user than the MEC server that is co-located with the nearest BS. The second reason is the forwarding overhead caused by the proxy server that is caused by the large amount of service requests that need to be handled. Even if the assumption that a MEC server is associated with every BS is idealized, the experienced latencies could definitely be reduced when a more distributed computing setup is employed.

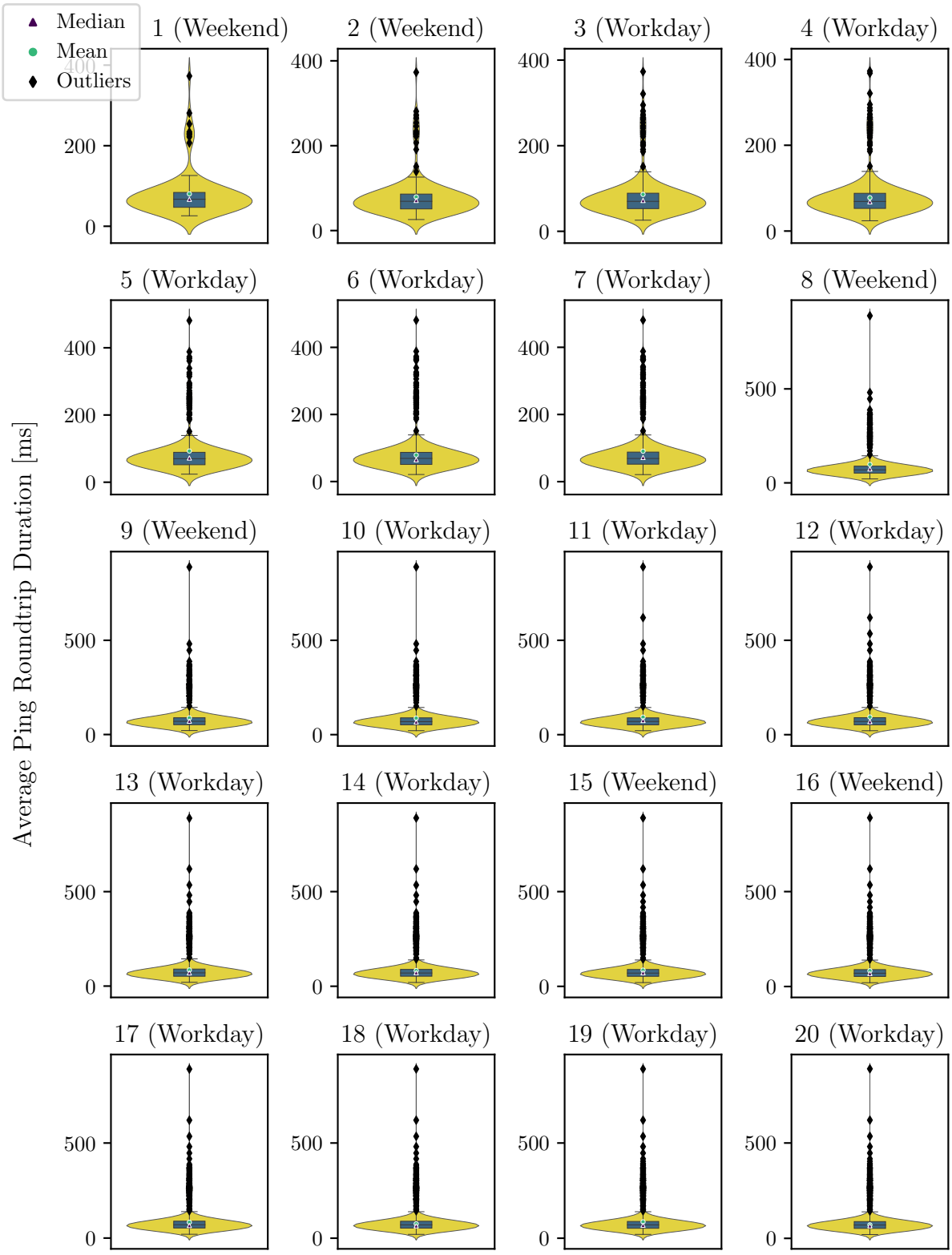


Figure 7.2: Violin plots for the average ping durations per day.

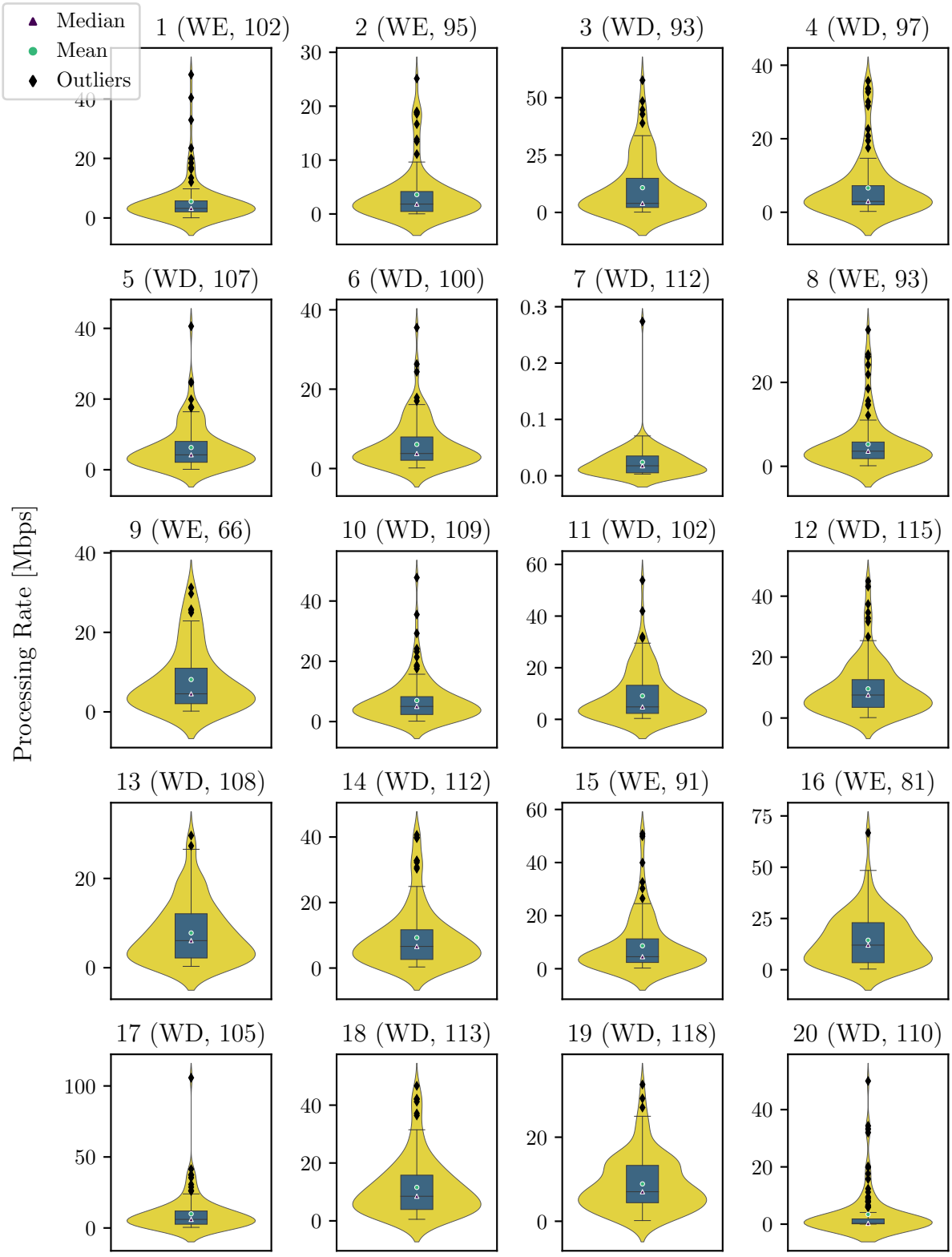


Figure 7.3: Violin plots for the processing rates per day for the hybrid voice dialog when considering the pure processing time.

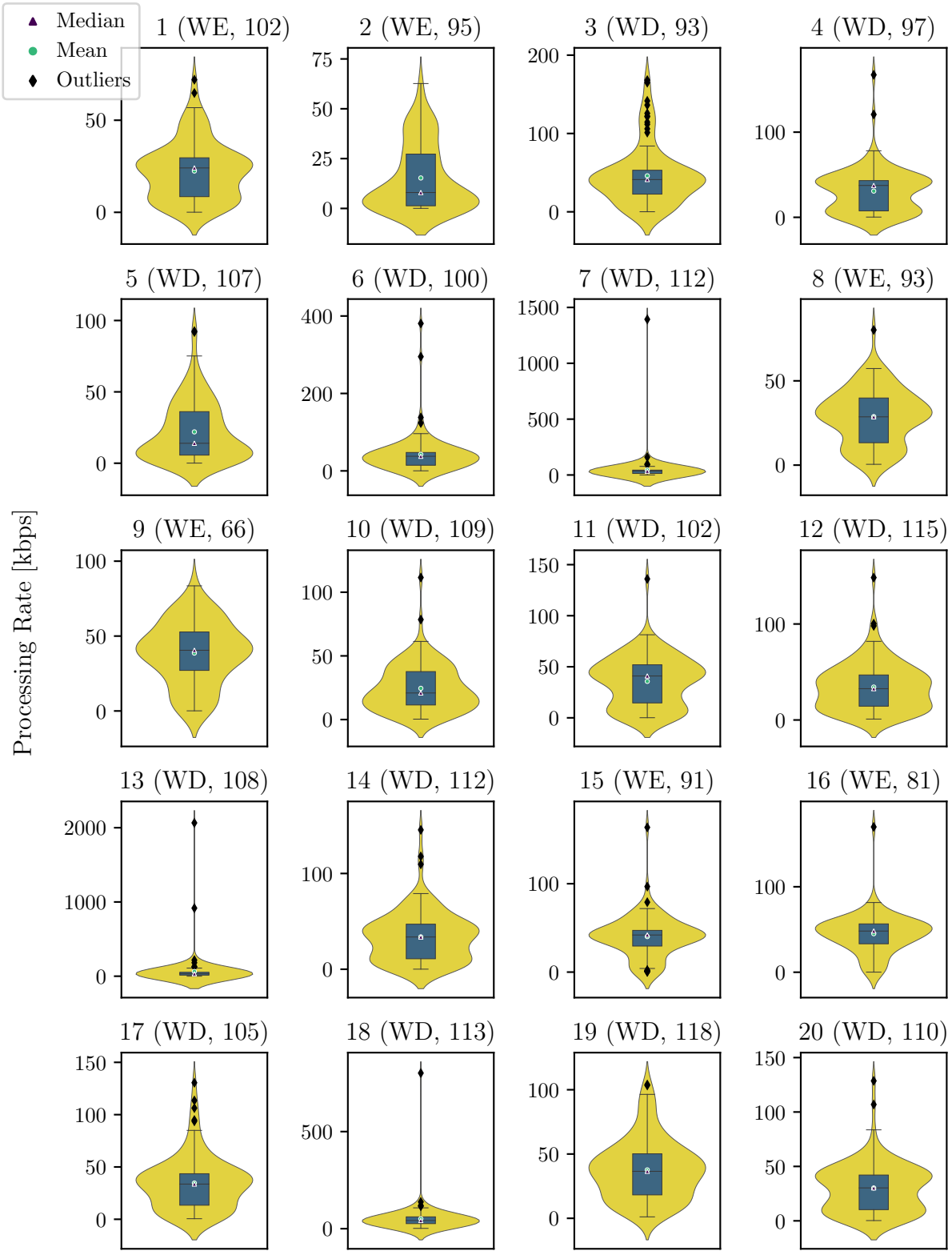


Figure 7.4: Violin plots for the processing rates per day for the hybrid voice dialog when considering the entire processing time between the reception and the response of the service request at the proxy server.

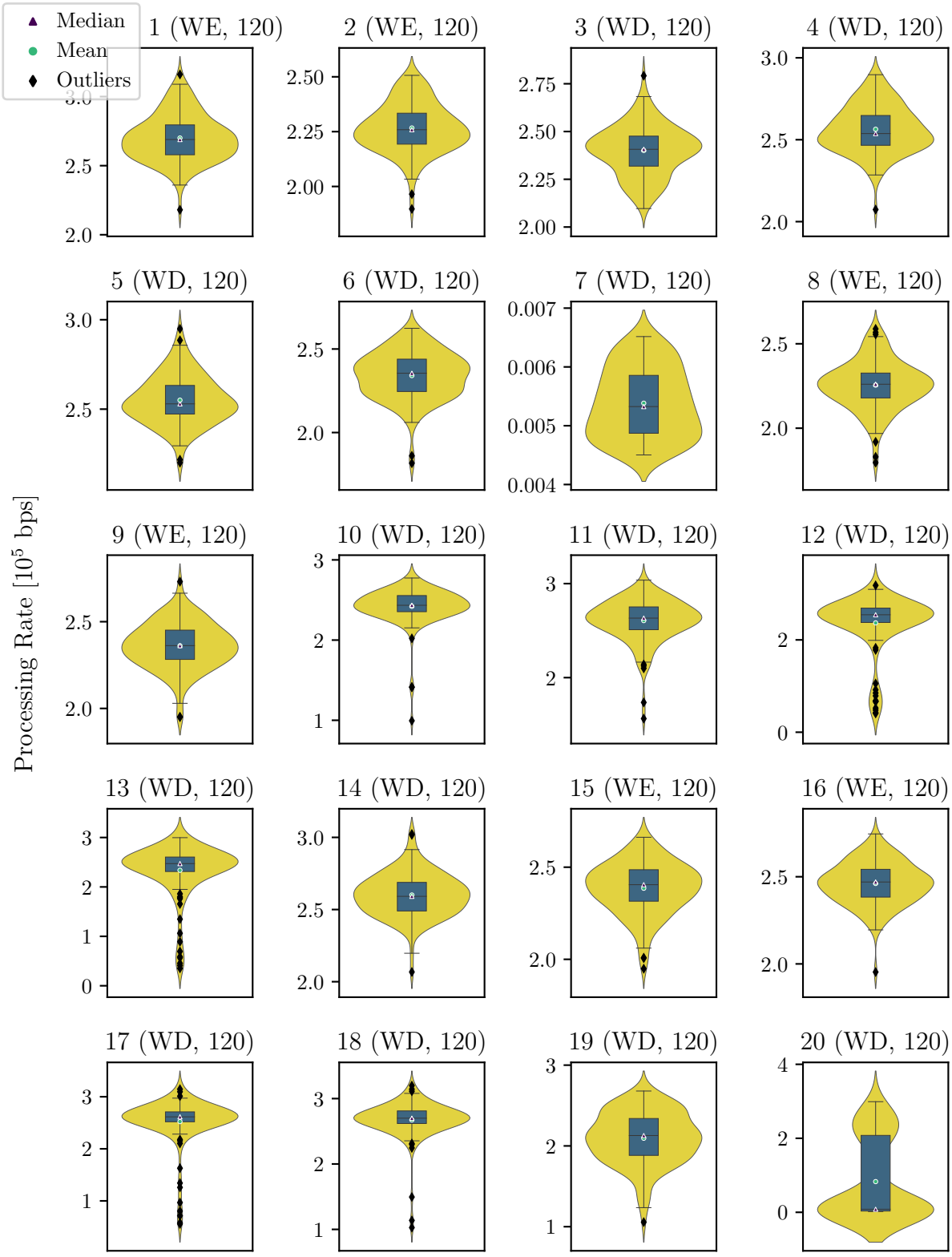


Figure 7.5: Violin plots for the processing rates per day for the secure time when considering the pure processing time.

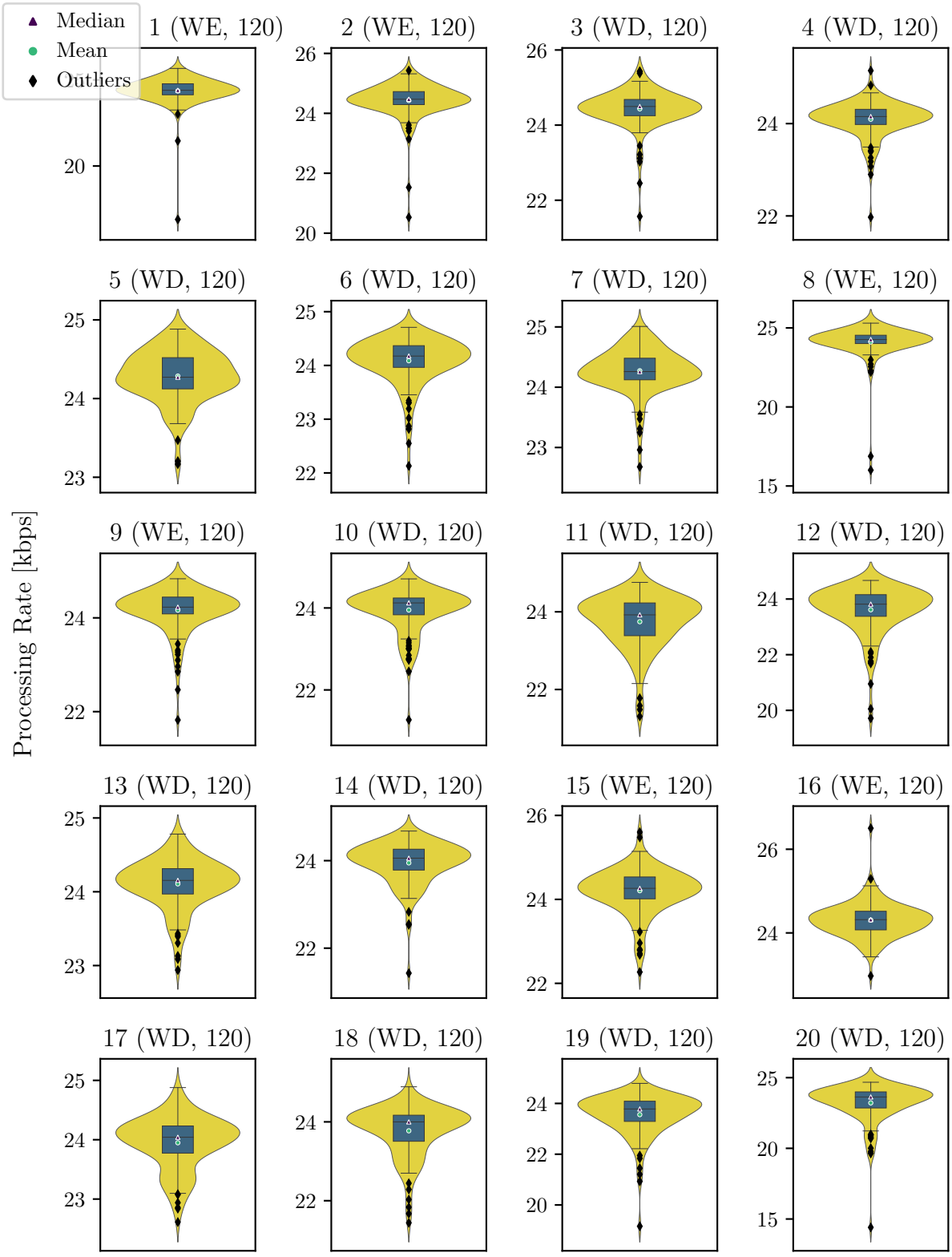


Figure 7.6: Violin plots for the processing rates per day for the secure time when considering the entire processing time between the reception and the response of the service request at the proxy server.

8 Conclusions

In the previous chapters, approaches on near-optimal RAN and edge computing resource allocation were presented. After outlining related work on this topic, the system model that was considered in this thesis was introduced. Next, admission control policies for homogenous and heterogenous sets of users were derived for a scenario consisting of uplink communication and edge processing. Given the availability of sufficient resources guaranteed by applying the admission policy, near-optimal resource allocation schemes for the uplink-only scenario were developed for the cases no fairness, proportional fairness, delay minimization, and max-min fairness. Subsequently, the analyses and approximation algorithms were extended to a scenario including downlink communication. Finally, the achieved MEC delays were compared to a CCC system from an automotive OEM.

8.1 Summary

After proving the analytical intractability of the equal-share approach, the maximum number of users from a homogenous set that can be served by the network with a reliability of 100 % was found along the Pareto Frontier as a trade-off between RAN and computing resources. Based on this result, an algorithm was designed to find the maximum number of users from a homogenous set that can be handled with general reliability $1 - \epsilon$. For a heterogenous set of users, specific conditions that a newly arriving user must fulfill in order to be admitted to the network were determined. The theoretical results showed perfect concordance with conducted simulations and comparisons to an approach based on separate consideration of network and computing resources indicated the superior performance of the devised policies.

By reformulating the continuous relaxation of the original integer nonlinear optimization problem for the joint resource allocation of RAN and edge computing resources into a convex optimization problem with generalized inequality constraints, it can be proven that an optimal solution to this problem can be found in polynomial time for both the uplink-only and the two-way communication scenario. Employing insights from the continuous resource assignment results, approximation algorithms relying on the conversion of the optimal continuous allocation results to integer assignments can be designed for near-optimal integer resource allocation. These approximation algorithms operate in two steps: First,

all processing resources and the amount of RAN resources needed to fulfill a user's latency requirement are assigned. Then, the remaining PRBs are allocated according to the specified fairness metric. For both the uplink-only and the two-way communication scenario, the conducted simulations prove a superb performance of the devised approximation algorithms, where the largest deviation from the continuous optimum among all fairness metrics and scenarios is 11.75 %, while the largest average deviation is 1.58 %.

Two types of data were analyzed from an existing CCC system employed by an automotive OEM. On the one hand, the average transmission latency from vehicular users located in the EMEA region was determined, and on the other hand, the average processing rate of a server setup including a proxy server was investigated by looking at two different services. Using these discovered values, it was shown that the devised MEC system allows for a reduction in the experienced delay by two orders of magnitude.

8.2 Prospects

In order to fulfill the reliability and latency demands of users requesting services based on URLLC, MEC solutions will play a major role in the development of the network architectures of the future. On the one hand, network operators need to employ appropriate admission policies suited for these systems to maximize the number of users that can be served in order to maximize their revenue. On the other hand, these admission policies need to ensure the availability of sufficient resources, such that the operators can adhere to the SLAs. In combination with these admission policies, suitable resource allocation schemes need to be devised, that are based on already existing specifications of the working principles of 5G NR. The interplay of fronthaul and backhaul networks, computing resources, and storage resources will thereby be an important aspect and pose a significant challenge.

It is the responsibility of both the economic and the academic community to develop such admission control policies and resource allocation schemes. As already stated in Chapter 6, the design of an admission policy for the two-way communication scenario constitutes a future work that should be followed. Additionally, investigating guidelines that also take varying channel conditions in the frequency domain into account represent an interesting research approach. Regarding the presented resource allocation schemes, the implementation of the approximation algorithms in a testbed could prove the actual applicability in a real setup and also provide interesting insights in the performance in reality. Finally, as outlined previously, even more enhanced scenarios than the present one, including, e.g., network allocated storage resources, pose research problems that should be addressed in the future.

Bibliography

[BDP18] Mehdi Bennis, Mérouane Debbah, and H Vincent Poor. Ultrareliable and low-latency wireless communication: Tail, risk, and scale. *Proceedings of the IEEE*, 106(10):1834–1853, 2018.

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[Cha09] Robert Chares. *Cones and interior-point algorithms for structured convex optimization involving powers and exponentials*. PhD thesis, Université Catholique de Louvain Louvain-la-Neuve, Louvain, Belgium, 2009.

[CLLW19] Hsu-Tung Chien, Ying-Dar Lin, Chia-Lin Lai, and Chien-Ting Wang. End-to-end slicing with optimized communication and computing resource allocation in multi-tenant 5G systems. *IEEE Transactions on Vehicular Technology*, 69(2):2079–2091, 2019.

[CVS20] Marco Centenaro, Lorenzo Vangelista, and Stephan Saur. Analysis of 5G radio access protocols for uplink URLLC in a connection-less mode. *IEEE Transactions on Wireless Communications*, 19(5):3104–3117, 2020.

[DP19] Apostolos Destounis and Georgios S Paschos. Complexity of URLLC scheduling and efficient approximation schemes. *arXiv preprint arXiv:1904.11278*, 2019.

[DPS20] Erik Dahlman, Stefan Parkvall, and Johan Skold. *5G NR: The next generation wireless access technology*. Academic Press, 2020.

[EJAGS19] Salah Eddine Elayoubi, Sana Ben Jemaa, Zwi Altman, and Ana Galindo-Serrano. 5G RAN slicing for verticals: Enablers and challenges. *IEEE Communications Magazine*, 57(1):28–34, 2019.

[ETS22a] ETSI. 5G NR base station (BS) radio transmission and reception: 3GPP TS 38.104 version 17.6.0 release 17. `www.etsi.org`, 2022. Technical Specification.

[ETS22b] ETSI. 5G NR physical channels and modulation: 3GPP TS 38.211 version 17.1.0 release 17. `www.etsi.org`, 2022. Technical Specification.

[ETS22c] ETSI. 5G NR physical layer procedures for data: 3GPP TS 38.214 version 17.1.0 release 17. `www.etsi.org`, 2022. Technical Specification.

[GB08] Michael Grant and Stephen Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008. `http://stanford.edu/~boyd/graph_dcp.html`.

[GB14] Michael Grant and Stephen Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. `http://cvxr.com/cvx`, March 2014.

[GG07] Sally A Goldman and Kenneth J Goldman. *A practical guide to data structures and algorithms using Java*. Chapman and Hall/CRC, 2007.

[GMRLa20] Nipuni Uthpala Ginige, K B Shashika Manosha, Nandana Rajatheva, and Matti Latva-aho. Admission control in 5G networks for the coexistence of eMBB-URLLC users. In *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, pages 1–6. IEEE, 2020.

[Gol05] Andrea Goldsmith. *Wireless communications*. Cambridge university press, 2005.

[HDD+18] Bin Han, Antonio DeDomenico, Ghina Dandachi, Anastasios Drosou, Dimitrios Tzovaras, Roberto Querio, Fabrizio Moggio, Omer Bulakci, and Hans D Schotten. Admission and congestion control for 5G network slicing. In *2018 IEEE Conference on Standards for Communications and Networking (CSCN)*, pages 1–6. IEEE, 2018.

[HEGS+18] Yishu Han, Salah Eddine Elayoubi, Ana Galindo-Serrano, Vineeth S Varma, and Malek Messai. Periodic radio resource allocation to meet latency and reliability requirements in 5G networks. In *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, pages 1–6. IEEE, 2018.

[HMCK22a] Valentin Thomas Haider, Fidan Mehmeti, Ana Cantarero, and Wolfgang Kellerer. Joint α-fair allocation of RAN and computing resources to URLLC users in 5G. unpublished, October 2022.

[HMCK22b] Valentin Thomas Haider, Fidan Mehmeti, Ana Cantarero, and Wolfgang Kellerer. Joint α-fair allocation of RAN and computing resources to vehicular users with URLLC traffic. unpublished, September 2022.

[HNLF19] Vu Nguyen Ha, Ti Ti Nguyen, Long Bao Le, and Jean-François Frigon. Admission control and network slicing for multi-numerology 5G wireless networks. *IEEE Networking Letters*, 2(1):5–9, 2019.

[LL11] Jon Lee and Sven Leyffer. *Mixed integer nonlinear programming*, volume 154. Springer Science & Business Media, 2011.

[MHK22] Fidan Mehmeti, Valentin Thomas Haider, and Wolfgang Kellerer. Admission control for URLLC traffic with computation requirements in 5G and beyond. unpublished, October 2022.

[ML19] Fidan Mehmeti and Thomas F La Porta. Admission control for consistent users in next generation cellular networks. In *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, pages 1–7. IEEE, 2019.

[ML21a] Fidan Mehmeti and Thomas F La Porta. Admission control for mMTC traffic in 5G networks. In *Proceedings of the 17th ACM Symposium on QoS and Security for Wireless and Mobile Networks*, pages 79–86, 2021.

[ML21b] Fidan Mehmeti and Thomas F La Porta. Admission control for URLLC users in 5G networks. In *Proceedings of the 24th International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pages 199–206, 2021.

[ML21c] Fidan Mehmeti and Thomas F La Porta. Analyzing a 5G dataset and modeling metrics of interest. In *2021 17th International Conference on Mobility, Sensing and Networking (MSN)*, pages 81–88. IEEE, 2021.

[ML22] Fidan Mehmeti and Thomas F La Porta. Reducing the cost of consistency: Performance improvements in next generation cellular networks with optimal resource reallocation. *IEEE Transactions on Mobile Computing*, 2022.

[MOS22] MOSEK ApS. *The MOSEK optimization toolbox for MATLAB manual. Version 10.0.20.*, 2022.

[NORDS+20] Jorge Navarro-Ortiz, Pablo Romero-Diaz, Sandra Sendra, Pablo Ameigeiras, Juan J Ramos-Munoz, and Juan M Lopez-Soler. A survey on 5G usage scenarios and traffic models. *IEEE Communications Surveys & Tutorials*, 22(2):905–929, 2020.

[OF20] Mourice O Ojijo and Olabisi E Falowo. A survey on slice admission control strategies and optimization schemes in 5G network. *IEEE Access*, 8:14977–14990, 2020.

[ON19] Anuar Othman and Nazrul Anuar Nayan. Efficient admission control and resource allocation mechanisms for public safety communications over 5G network slice. *Telecommunication Systems*, 72(4):595–607, 2019.

[RLSQ20] Darijo Raca, Dylan Leahy, Cormac J Sreenan, and Jason J Quinlan. Beyond throughput, the next generation: a 5G dataset with channel and context metrics. In *Proceedings of the 11th ACM multimedia systems conference*, pages 303–308, 2020.

[SIL+16] Hamidreza Shariatmadari, Sassan Iraji, Zexian Li, Mikko A Uusitalo, and Riku Jäntti. Optimized transmission and resource allocation strategies for ultra-reliable communications. In *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pages 1–6. IEEE, 2016.

[Sri04] Rayadurgam Srikant. *The mathematics of Internet congestion control*. Springer, 2004.

[Sta21] William Stallings. *5G Wireless: A Comprehensive Introduction*. Addison Wesley, 2021.

[YZR20] Hao Yin, Lyutianyang Zhang, and Sumit Roy. Multiplexing URLLC traffic within eMBB services in 5G NR: Fair scheduling. *IEEE Transactions on Communications*, 69(2):1080–1093, 2020.

List of Figures

List of Tables

List of Algorithms