Investigating and Modeling Uplink Processing in 5G NR Multisector Scheduler Simulator

OSKAR NYGREN & AXEL SNEITZ-BJÖRKMAN
MASTER´S THESIS
DEPARTMENT OF ELECTRICAL AND INFORMATION TECHNOLOGY
FACULTY OF ENGINEERING | LTH | LUND UNIVERSITY

Investigating and Modeling Uplink Processing in 5G NR Multisector Scheduler Simulator

Oskar Nygren
Axel Sneitz-Björkman

Department of Electrical and Information Technology
Lund University

Abstract

As the latest generation of wireless access technology called Fifth Generation (5G) New Radio (NR) evolves, computational efficiency is key to keep low cost and flexible deployments for vendors. Understanding how the Digital Signal Processing (DSP) load behaves in the digital units is one aspect to enable this efficiency. This thesis investigates how the DSP load in the base station behaves for a cell in high band, where frequencies are in the range 24.25 GHz - 52.6 GHz, when varying the number of allocated Resource Blocks (RBs) in both a single and multisector configuration. The computational behaviour is also characterized using regression models and implemented into an internal simulator at Ericsson. Additionally, the investigation is complemented with data from real User Equipments (UEs) and a more real-life scenario where error occurs in the transmission.

The results show that the load scales with the number of RBs in such a way that it is possible to have more Connected Users (CUs) by limiting the allocated RBs for all users. Furthermore, the load generated by the uplink data should be considered on a Time Division Duplex (TDD) pattern basis. When doing so, it can be approximated as a linear or higher order function of the number of allocated resource blocks. When implemented into the simulator, the model is validated against data from real hardware. The validation showed that the model generates accurate normal distributed load values: it captures the essential behaviour but it is hard to predict correct mean values, due to the generated load being too stochastic in nature.

These insights on the load behaviour provides input and considerations on how to design solutions that does not exceed the maximum computation utilization. For future work, the behaviour from downlink data and other traffic scenarios could be investigated to fully explore the limitations and requirements.

Populärvetenskaplig sammanfattning

5G NR är den senaste generationen mobilnät och är efterföljaren till Fourth Generation (4G) Long-Term Evolution Advanced (LTE-A). I mobilnäten kan användare koppla upp sig mot basstationer med sina enheter som kan vara en mobiltelefon eller dator. Basstationerna består av antenner som kan både skicka och ta emot information samt en radioenhet som hanterar trafik till och från stationen. Eftersom antennen på en basstation placeras högt upp för bäst täckning går det att känna igen basstationer genom att antennerna ofta sitter på en mast eller höga hustak i större städer. Från antennen går det sedan en kabel till radioenheten som placeras på en lämplig plats, men den kan även vara integrerad tillsammans med antennen. Runtom basstationen delas topografin upp i hexagonformade områden, ungefär i samma utseende som bins vaxkakor. Alla användare som befinner sig inom samma hexagon tillhör ur basstationens perspektiv en och samma cell.

För varje cell finns det en viss mängd resurser tillgängliga i både tids- och frekvensdomänen. I tidsdomänen består en resurs genom att tiden går att dela upp i olika intervall vilket tillåter att fler användare delar på den. För användarnas enheter är dessa intervall tydliga men för oss människor märks det inte av utan det upplevs ändå som ett jämnt flöde av kommunikation. På samma sätt som att vi inte märker att glödlampor blinkar 50 gånger per sekund utan ljuset upplevs som konstant. På samma sätt som tiden delas upp kan frekvenser också delas upp i olika intervall vilket bildar frekvensresurser. Kombineras båda dessa uppdelningar går det att ha flera användare samtidigt som delar på de olika resurserna. Förutom att dela in frekvenserna i olika intervall finns det också begränsningar över hur stort frekvensområde varje cell förfogar över vilket kallas bandbredd. Det begränsar i sin tur hur många intervall frekvensområdet går att dela in i.

När det är många användare samtidigt i en och samma cell, vilket kan hända vid större folksamlingar som konserter, idrottsevenemang eller stadsparken i Lund under valborg, tar resurserna i cellen slut. Om då användarens enhet inte fått tillgång till resurserna på ett tag kommer den tro att kontakten med basstation har försvunnit och försöker då koppla upp sig igen mot samma eller en ny basstation. En lösning för att kunna ha fler användare uppkopplade samtidigt mot samma basstation utan att tappa dem är att minska de enskilda cellernas områden, vilket

ger en möjlighet att öka antalet celler runtom basstationen. Att ha flera celler samtidigt är kostsamt för radioenheten då dess processor måste arbeta hårdare för att kunna hantera all data som skickas till den från varje cell.

En annan typ av resurs i 5G NR kallas *resource block* eller på svenska: resursblock. Enkelt förklarat går det att beskriva antalet resursblock som antalet intervall för en viss bandbredd som finns i en cell. Genom att begränsa antalet resursblock som kan användas går det att minska mängden frekvensresurser som används för en cell trots att det finns mer tillgängligt. I det här arbetet har det undersökts hur belastningen för processorn i radioenheten har förändrats beroende på hur många såna resursblock som finns tillgängliga för upplänken. Upplänken är den kommunikation som sker från användarenheterna till basstationen.

Med hjälp av en särskild testmiljö på Ericsson som kan simulera mobiltelefoner har belastningen för en verklig radioenhet kunnat undersökas. Genom att begränsa antalet resursblock har data samlats in för olika antal resursblock, antal celler och två olika bandbredder. För varje resurblocksvärde som data samlats in för har en statistik modell skapats. Genom att interpolera parametrarna i den statistiska modellen för alla resursblocksvärden har en funktion för varje parameter tagits fram så att värden för dessa kan skapas för ett godtyckligt antal resursblock.

Därefter har den statistiska modellen implementerats i en intern simulator på Ericsson och data för tre olika resursblocksvärden som inte användes för att skapa modellen samlades in. I syfte att validera modellen samlades även testdata för de motsvarande tre resursblocksvärdena in från testmiljön som sedan användes för att validera simulatorns prestanda. Valideringen visade att modellen i många fall lyckas fånga utseendet på lastens distributioner men att dessa distributioner i några fall inte alltid har samma medelvärde som datan från testmiljön.

Med resultaten från simulatorn går det att dra slutsatser om hur belastningen för radioenheten med upplänkstrafik kommer att se ut och när den infaller. För att kunna möjliggöra fler scenarion och slutsatser från simulatorn kan även belastningen från nedlänkstrafik undersökas, modelleras och implementeras i simulatorn. Detta skulle ge en fullständig bild av hur belastning beter sig i radioenheten.

Acknowledgements

Table of Contents

List of Figures

List of Tables

List of Acronym and Abbreviations

3GPP   3rd Generation Partnership Project

4G   Fourth Generation

5G   Fifth Generation

AMC   Adaptive Modulation and Coding

BBU   Baseband Unit

BLER   Block Error Rate

CP   Cyclic Prefix

C-RAN   Cloud Radio Access Network

CSIM   Component Based Simulators

CU   Connected User

DL   Downlink

DSP   Digital Signal Processing

DU   Digital Unit

EMCA   Ericsson Many Core Architecture

eNB   eNodeB

FSUE   Full Stack User Equipment

gNB   gNodeB

ICI   Intercarrier Interference

ISI   Intersymbol Interference

KLD   Kullback-Liebler Divergence

L1   Physical Layer 1

LA   Link Adaptation

LAT	Logic Analyzer Tool
LTE-A	Long-Term Evolution Advanced
MAC	Medium Access Control
MCS	Modulation Coding Scheme
NR	New Radio
NSA	Non-standalone
OFDM	Orthogonal Frequency-Division Modulation
OLS	Ordinary Least Squares
OTA	Over-the-Air
PBCH	Physical Broadcast Channel
PDCCH	Physical Downlink Control Channel
PDSCH	Physical Downlink Shared Channel
PRACH	Physical Random-Access Channel
PUCCH	Physical Uplink Control Channel
PUSCH	Physical Uplink Shared Channel
QoS	Quality of Service
RA	Random Access
RAN	Radio Access Network
RB	Resource Block
RE	Resource Element
RTT	RealTimeTraces
RU	Radio Unit
SC	Sector Carrier
SR	Scheduling Request
SSB	Synchronization Signal Block
TDD	Time Division Duplex
TTI	Transmission Time Interval
UE	User Equipment
UL	Uplink
vRAN	Virtual Radio Access Network

Introduction

1.1 Background and Motivation

Given the increased demand for greater throughput in 5G mobile networks, vendors of Radio Access Network (RAN) equipment are constantly looking for ways to increase coverage while remaining efficient both in cost and computation. By using several Radio Units (RUs), connected to a single Digital Unit (DU), it is possible to cover a larger geographical area with fewer resources. This generates higher demand on the efficiency of the baseband software which executes in the DU. Specifically, the main processing load is generated in the Physical Layer 1 (L1) from processing UL and DL data.

The RUs are often integrated with the antennas forming a system and together they cover one sector. A sector is thus a geographical area spanned by the transmission angle from the antenna system. When connected to a DU, together with other baseband equipment, they form a base station. In 4G LTE-A, the base station is referred to as eNodeB (eNB) and in, 5G NR as gNodeB (gNB). If a base station has more than one antenna system, it can cover additional sectors illustrated in Figure 1.1.

Additionally, each sector is a set of cells where the cells can transmit on different frequencies illustrated in Figure 1.2. Each User Equipment (UE), belonging to one sector, can use the resources from either one or multiple cells by aggregating them.

Allocation of computational resources in the base station is controlled by a scheduler, which assigns resources based on Scheduling Requests (SRs) from UEs. The performance of a multisector configuration, where several RUs are connected to the same DU, is therefore highly dependent on the effectiveness of the scheduler. Specifically, it is interesting to analyze how certain parameters affect the scheduler impact on system load, to further improve performance. One of the parameters of special interest is the number of assigned Resource Block (RB) per cell, which essentially sets the maximum data rate a cell can achieve.

1
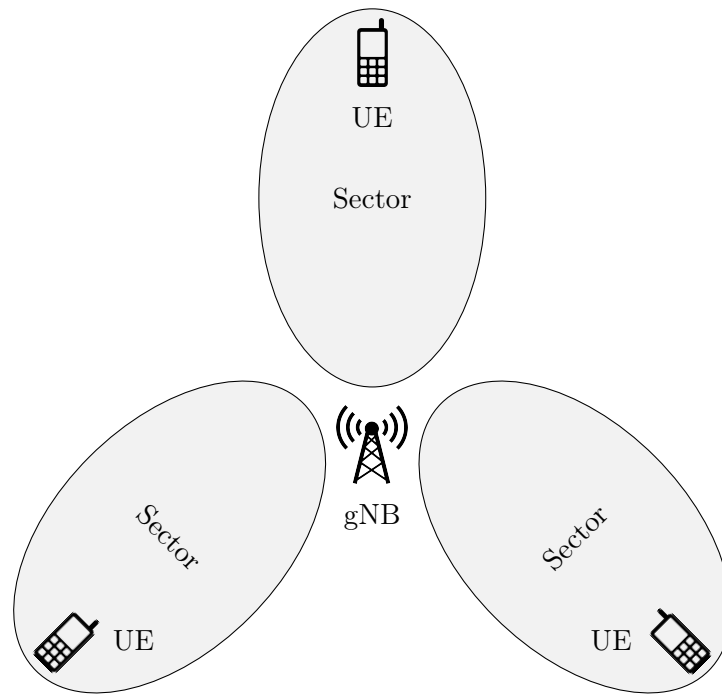
Figure 1.1: Different sectors around one gNodeB.
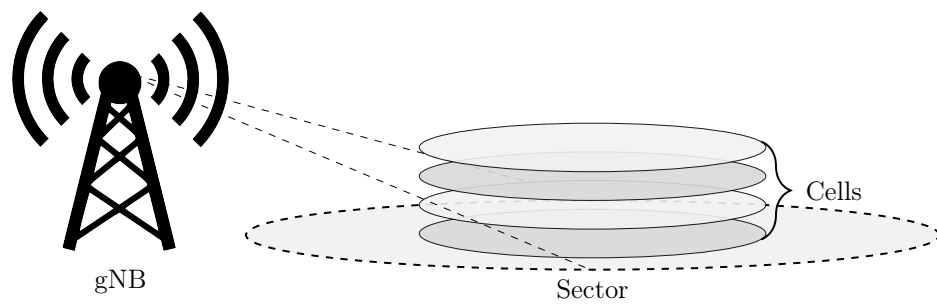
Figure 1.2: How cells relate to one sector from a gNodeB.

1.2   Objectives

The purpose of this thesis is to explore and model how the scheduler entity in high band, where frequencies are in the range 24.25 GHz - 52.6 GHz, behaves and affects the system load depending on the number of RBs that are utilized for each scheduled entity. To limit the scope of the thesis, only the load generated by the UL traffic, when data is flowing from the UEs to the base station, is considered.

If a UE goes for a long period without any UL transmission, the synchronization will be lost and the UE will drop the connection to the base station. To keep a high number of Connected Users (CUs) it is important to not throttle UL resources in a such a way that the UEs will believe the link is dead and drop the connection. More specifically, the following items are addressed:

- How the DSP load generated by UL data scales with the number of RBs assigned per cell.

- How the number of RBs assigned per cell affects the system load in a multisector/cell environment. Is it for example possible to support a larger number of primary cells, and thus more CUs, by assigning a lower number of RBs per cell in a high load scenario?

- How the system load is affected in a more real-life scenario different from the high load. For example, how does the Link Adaptation (LA) algorithm change the processing requirements?

- How data from real-world hardware tests can be used to build a model which indicates the performance of different configurations.

- How adding model parameters to an internally developed simulation tool can extend functionality to that tool.

Due to time constraints, the collection of data that were to be analyzed needed to be limited. Thus, only data from cells with 50 and 100 MHz bandwidth were collected. Also, due to limitations in the testing software, only configurations with up to three sectors were investigated.

1.3   Previous Work

Many system providers run their software on special proprietary hardware and thus considers the DSP load behavior as confidential. Therefore, the open literature in this area is quite scarce, but there are some articles relating to the work in this thesis. In [1], Pramanik et al. investigates the computational and memory requirements of Virtual Radio Access Networks (vRANs) for an LTE-A emulation system. vRANs lets the operators run their baseband functions as software on general-purpose hardware. In the article, the authors build a vRAN test bed using general-purpose computers to profile the computing and memory behavior when sending DL data. Among other results, the authors show that the CPU utilization and throughput increases with the MCS index for a single user. They

also show that the CPU utilization increases with the number of occupied RBs for a fixed MCS index of 27. They find that these increases in CPU utilization can be approximated as linear functions of both MCS index and number of occupied RBs. The last interesting result from their article is that the CPU utilization also increases for each additional connected UE for a fixed number of RBs and the MCS index set to 27.

Somewhat similar, Tran et al. [2] look at the processing behavior for a virtual Baseband Unit (BBU) in a Cloud Radio Access Network (C-RAN) test bed. In the article the authors investigate and show how the CPU utilization increases with the MCS index for three different configurations of 25, 50 and 100 RBs sending DL data. By knowing how many bits there are in each symbol for a given MCS index value, the CPU utilization was calculated for a given throughput. It was found that the CPU utilization could be approximated as a linear function of the DL throughput or the number of occupied RBs.

There are two key differences from these articles comparing to the work in this thesis. First, the baseband software is simulated in the articles whereas real hardware is utilized in this thesis. Secondly, the articles investigate load in the previous generation telecommunication generation, 4G, whereas this thesis focus on the current generation (i.e., 5G). Therefore, it is not certain that their results can be directly compared to the results of the work in this thesis.

Theoretical Background

2.1 5G NR

5G NR is the successor to the previous telecommunication generation 4G LTE-A. The technical specifications for 5G are developed by the organization 3rd Generation Partnership Project (3GPP). Based on the same technology as LTE-A, NR reuses many features from LTE-A without the restrictions of retaining backwards compatibility, allowing new features and technical solutions.

One of the main features of NR is that it allows utilization of the mm-Wave frequencies in the range 24.25 GHz - 52.6 GHz [3]. There are also three distinct use cases for 5G often talked about which are: Enhanced Mobile Broadband, Massive Machine-type Communication and Ultra-Reliable and Low-Latency Communication.

2.2 Orthogonal Frequency-Division Multiplexing

Orthogonal Frequency-Division Modulation (OFDM) [4] is a modulation scheme which divides the data on multiple carrier frequencies. In conventional frequency division duplex schemes, the spacing between the carriers must be sufficiently large to avoid Intercarrier Interference (ICI). In OFDM, the subcarriers are orthogonal to each other and spaced in such a way that when sampled, in theory, only the contribution from one subcarrier is used, allowing them to be more closely spaced without interfering with one another and thus increasing the spectral efficiency. In practice there is always a small (but minimal) interference. Figure 2.1 illustrates orthogonal carriers in the time- and frequency domains.

However, OFDM still suffers from Intersymbol Interference (ISI) where parts of the previous symbol(s) are leaking over in the time domain and interfering with the desired symbol. OFDM symbols is a time domain modulation of a complex number, representing the amplitude and phase of the modulated bits. One way to combat ISI is the use of a guard interval in the time domain. With OFDM, a Cyclic Prefix (CP) is often used. The CP is applied by copying the last part of

the OFDM symbol and attaching it to the front of the symbol.
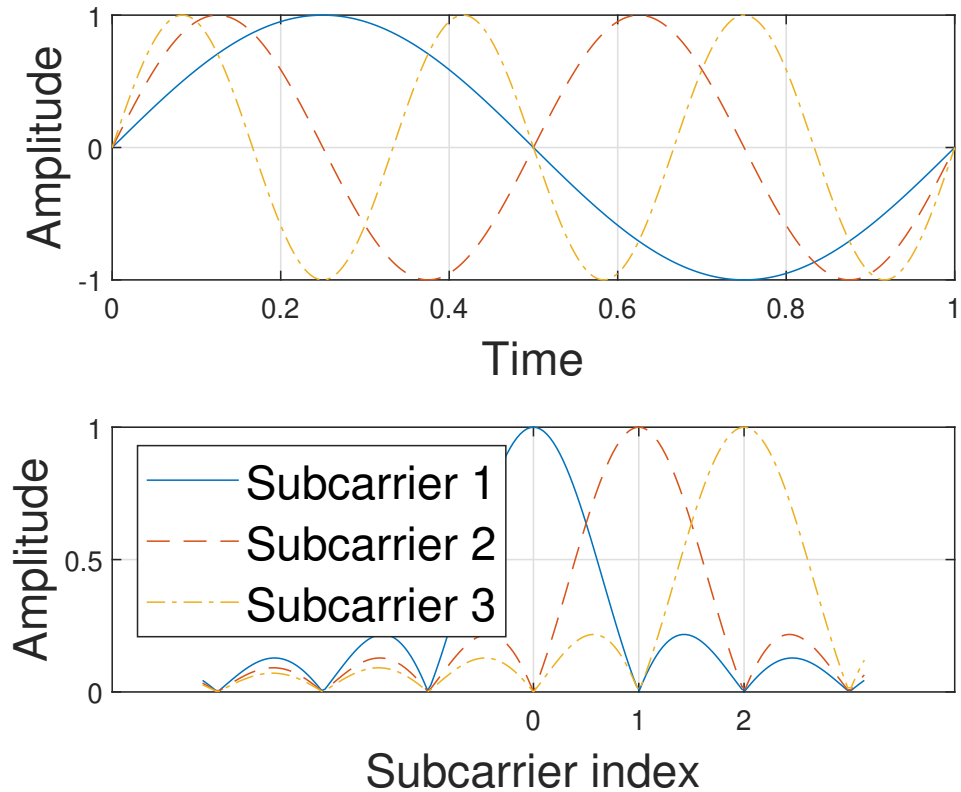
Figure 2.1: Orthogonal subcarriers in the time- and frequency do-
mains.

The modulation scheme is what determines how many bits there are in each OFDM symbol, from one bit per symbol using binary phase-shift keying up to eight bits per symbol using 256 quadrature amplitude modulation. The modulation order is controlled by MCS which also determines the coding rate, the latter is a ratio of how much redundancy there is in the data stream. Typically a 10 % Block Error Rate (BLER) [5] is targeted by the Adaptive Modulation and Coding (AMC) algorithm which allocates different MCS values. It may seem paradoxical to allow errors in the communication to happen but trying to achieve a extremely low error rate is costly and doesn't necessarily provide the best performance. A selection of modulation orders, target code rates and spectral efficiencies are shown in Table 2.1 for PUSCH, recreated from [5, table 6.1.4.1-1]. Spectral efficiency is a measure of how many bits per second that can be transmitted over a given bandwidth. Generally both the modulation order and code rate increase with the MCS index.

MCS Index	Modulation Order	Target code rate (R x 1024)	Spectral efficiency [bit/(s·Hz)]
$\vdots$	$\vdots$	$\vdots$	$\vdots$
15	4	616	2.4063
16	4	658	2.5703
$\vdots$	$\vdots$	$\vdots$	$\vdots$
26	6	910	5.3320
27	6	948	5.5547

Table 2.1: MCS index Table for PUSCH.

2.3 Time/Frequency Structure in 5G NR

2.3.1 Time Domain

In the time domain the length of one OFDM symbol varies with the numerology. The numerology is defined by the sub-carrier spacing and the cyclic prefix length and is given by $\Delta f = 2^{\mu} \cdot 15$ kHz [6, table 4.3-1]. Extended CP, which is used to further minimize ISI, is only supported for numerology two. Table 2.2 shows some of the different numerologies for NR.

μ	Δf (kHz)	CP
0	15	Normal
1	30	Normal
2	60	Normal, Extended
3	120	Normal
4	240	Normal

Table 2.2: Different numerologies in NR.

Also, in the time domain, the OFDM symbols are divided into radio frames, subframes and slots. In NR, one frame is always 10 ms and is divided into ten subframes of 1 ms each [7, p. 5.3]. These subframes are then divided into slots where each slot consists of 14 OFDM symbols for normal CP and 12 for extended CP. Each numerology will have a different OFDM symbol length in time which in turn corresponds to the number of slots in a subframe. For example, in numerology three, the useful OFDM symbol time is 8.92 μs which means that one slot is 0.125 ms and one subframe consists of eight slots. Table 2.3 and Figure 2.2 illustrates the number of slots per subframe and frame for different numerologies.

Lastly there is the concept of Transmission Time Interval (TTI), which corresponds to consecutive OFDM symbols in the time domain [8, p. 5.4.7]. By using different number of symbols in a TTI, different durations can be defined. One way to look at TTI is that it is the length of one scheduled transmission. Together with the numerology, the TTI determines how the transmission is to be made on the

μ	Symbols	Slots / subframe	Slots / frame
0	14	1	10
1	14	2	20
2	14	4	40
3	14	8	80
4	14	16	160

Table 2.3: Slots per subframe and frame for different numerologies.

physical layer. In this thesis, one TTI is always equal to one slot or 14 OFDM symbols.

2.3.2 Frequency Domain

Like the time domain, the frequency domain also has subdivisions that are related to each other. Frequency ranges are grouped into different bands and one carrier is a contiguous band of 50, 100, 200 or 400 MHz in NR [9, table 5.3.2-1]. Depending on the numerology, the subcarriers have different widths. For example, numerology three has subcarriers which are evenly spaced by 120 kHz. These subcarriers lie next to each other except at the edges of the carrier where guard bands are used to reduce ICI.

In NR, 12 subcarriers are always grouped into a RB, irrespectively of the numerology used [6]. Thus, the carriers will contain different number of RBs depending on the channel bandwidth used. Ericsson currently uses numerology three in their NR high band implementation and this report will assume that same numerology unless stated otherwise. Table 2.4 shows the maximum number of RBs for different channel bandwidths with 120 kHz sub-carrier spacing [9, table 5.3.2.-1].

Bandwidth [MHz]	50	100	200	400
RBs	32	66	132	264

Table 2.4: Maximum number of RBs for different channel bandwidths with 120 kHz sub-carrier spacing.

2.3.3 Time-Frequency Grid

Combining the time and frequency domains results in a time-frequency grid. Figure 2.3 shows the combination of those domains in where each block in the frequency axis represents a sub-carrier frequency and each block on the time axis represents a pulse. This is often referred to as the resource grid.

The smallest unit in the resource grid is 1 OFDM symbol × 1 sub-carrier which is called a Resource Element (RE). Another unit is 1 slot × 1 RB, that is, $14 \times 12 = 168$ REs and in the context of the previous telecommunication generation (4G), this is often referred to as a *resource block*, which has the same same frequency-

One frame, $T_{\text{frame}} = 10$ ms

One subframe, $T_{\text{subframe}} = 1$ ms

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

$\Delta f = 15$ kHz

One slot, 1 ms

$\Delta f = 30$ kHz

One slot, 0.5 ms

$\Delta f = 60$ kHz

One slot, 0.25 ms

$\Delta f = 120$ kHz

One slot, 0.125 ms

$\Delta f = 240$ kHz

One slot, 0.0625 ms
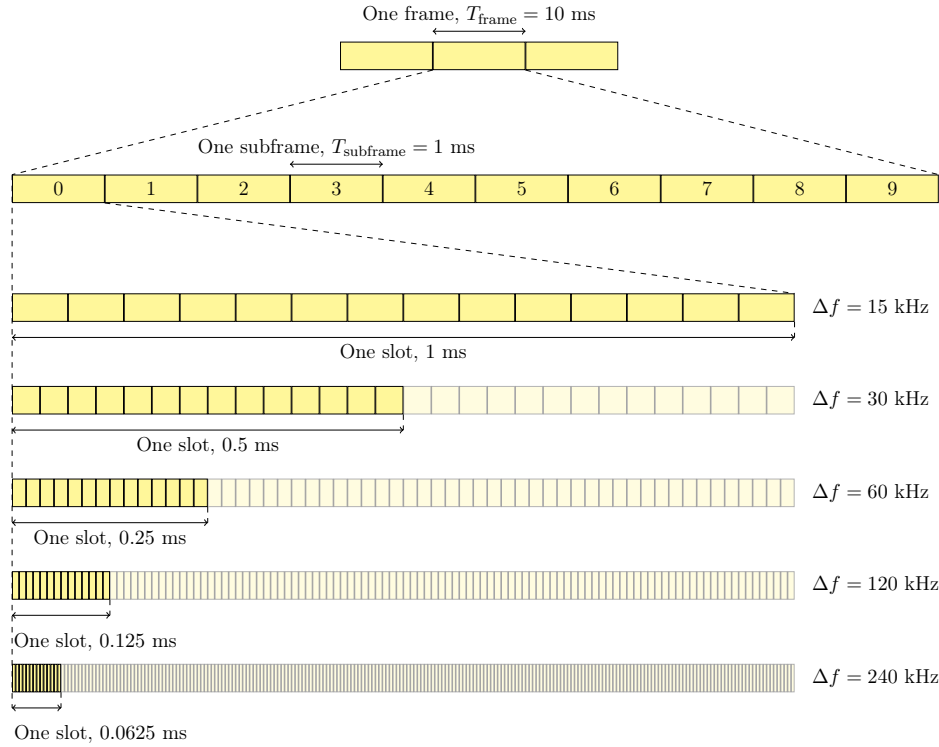
Figure 2.2: Frames, subframes, and slots in NR.

domain unit of 12 subcarriers as NR. This is most likely because the definition of RBs differ between LTE-A and NR [10]. In this report the NR frequency-domain definition of 12 sub-carrier is implied when referring to RBs unless stated otherwise.

2.3.4 Physical Channels

There are different physical-channel types defined for NR [11]. The main channels for unicast data in down- and uplink are called Physical Downlink Shared Channel (PDSCH) and Physical Uplink Shared Channel. Control information are sent on the Physical Downlink Control Channel (PDCCH) and Physical Uplink Control Channel (PUCCH) respectively. There is also the Physical Broadcast Channel (PBCH) that carries information, for example synchronization data, necessary for UEs to access the network and lastly, there is the Physical Random-Access Channel (PRACH) used for random access.

2.3.5 Scheduling

In NR the resource blocks, symbols and slots in Figure 2.3 are dynamically allocated and shared between users. The entity that controls this assignment of

Frequency

Resource Element

Resource
Block
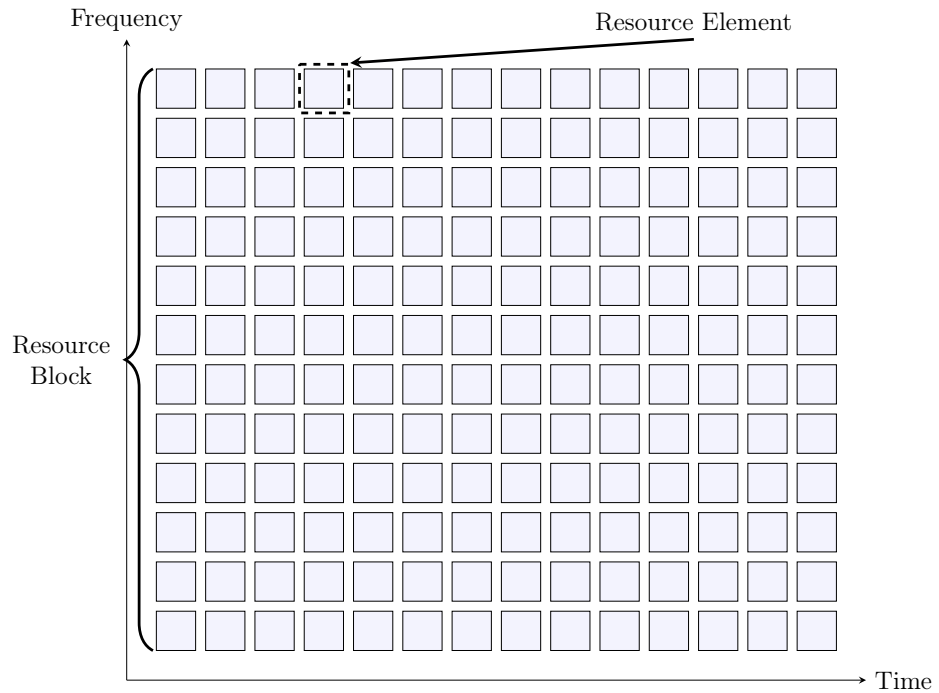
Time

Figure 2.3: Time-frequency resource grid

resources in uplink and downlink is called the scheduler and although it is part of the Medium Access Control (MAC) layer, it is often viewed as a separate entity operating alongside the layer [11].

The parameters by which the scheduler assigns uplink resources are UE buffer status, Quality of Service (QoS) requirements and radio conditions [12, p. 10]. It then signals its decisions to the UEs, who in turn, knows which resources and when it has been granted. The schedulers strategy to take decisions is not specified by 3GPP and the implementation is specific to the system manufactures. Most scheduler implementations favor scheduling users with the most advantageous channel conditions in both time and frequency, which is referred to as channel-dependent scheduling [11].

Scheduling and execution can not be performed simultaneously as there is a delay from when the scheduler has taken a decision to when it has been communicated to a UE. There are also other considerations such as Random Access (RA), when new UEs are allowed to enter a cell and reference signals that needs resources. With a purely reactive scheduler, this planning would become too complex. One solution is to use a fixed pattern which covers several TTIs so that the scheduler knows in advance what types of requests it can serve in that specific TTI.

The current pattern that Ericsson uses is called the 4:1 pattern where four consecutive download slots (PDSCH) are followed by one uplink slot (PUSCH). This

pattern is repeated over 160 TTIs and within those 160 TTIs, some of the slots are allocated for other services such as reference signals or RA. 12 out of the 32 PUSCH slots are shared with RA and might have fewer REs devoted to PUSCH. This pattern is shown in Figure 2.4.

It is important to understand that the pattern in Figure 2.4 shows the Over-the-Air (OTA) pattern. The processing for the data that is to be sent in a specific slot has to be completed before that slots begins. This means that the load that corresponds to a specific slot has been generated in a previous slot.

2.3.6   Synchronization and Random Access

Periodically the scheduler will send out signals to enable initial access for devices looking to find a cell. This happens through the Synchronization Signal Block (SSB) which is broadcasted and consists mainly of three parts: Two synchronization signals called Primary Synchronization Signal and Secondary Synchronization Signal and the PBCH. The SSB contains, among other things, information such as the cell carrier frequency, frequency offset and the system timing. After the synchronization is completed a device can begin the RA procedure to access the found cell.

The RA procedure consists of four steps, often referred to as "Message 1, 2, 3 and 4", where the device and network exchange these messages between each other. The RA procedure is not only for new devices but can also be used in other contexts. For example, when there has been a period of no UL transmission, the synchronization to the base station will be lost and the UE can use the procedure to reestablish the synchronization [13].

2.3.7   Link Adaptation

LA is a functionality which dynamically matches the current channel conditions to the modulation order, coding rate and other signal and protocol parameters. The UEs sends channel quality indicators to the base station as an estimated recommendation of what MCS that it can reliably receive. In NR there are four types of LA supported: Adaptive Transmission Bandwidth, Adaptive Transmission Duration, Transmission Power Control and the previously mentioned AMC [12].

Slot 0	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7	Slot 8	Slot 9
DL	DL	SSB	DL	UL	SSB	SSB	DL	DL	UL RA

Slot 10	Slot 11	Slot 12	Slot 13	Slot 14	Slot 15	Slot 16	Slot 17	Slot 18	Slot 19
DL	DL	SSB	DL	UL	SSB	SSB	DL	DL	UL RA

Slot 20	Slot 21	Slot 22	Slot 23	Slot 24	Slot 25	Slot 26	Slot 27	Slot 28	Slot 29
DL	DL	DL	DL	UL	DL	DL	DL	DL	UL RA

Slot 30	Slot 31	Slot 32	Slot 33	Slot 34	Slot 35	Slot 36	Slot 37	Slot 38	Slot 39
DL	DL	DL	DL	UL	DL	DL	DL	DL	UL RA

Slot 40	Slot 41	Slot 42	Slot 43	Slot 44	Slot 45	Slot 46	Slot 47	Slot 48	Slot 49
DL	DL	DL	DL	UL	DL	DL	DL	DL	UL RA

Slot 50	Slot 51	Slot 52	Slot 53	Slot 54	Slot 55	Slot 56	Slot 57	Slot 58	Slot 59
DL	DL	DL	DL	UL	DL	DL	DL	DL	UL RA

Slot 60	Slot 61	Slot 62	Slot 63	Slot 64	Slot 65	Slot 66	Slot 67	Slot 68	Slot 69
DL	DL	DL	DL	UL	DL	DL	DL	DL	UL RA

Slot 70	Slot 71	Slot 72	Slot 73	Slot 74	Slot 75	Slot 76	Slot 77	Slot 78	Slot 79
DL	DL	DL	DL	UL	DL	DL	DL	DL	UL RA

Slot 80	Slot 81	Slot 82	Slot 83	Slot 84	Slot 85	Slot 86	Slot 87	Slot 88	Slot 89
DL	DL	DL	DL	UL	DL	DL	DL	DL	UL RA

Slot 90	Slot 91	Slot 92	Slot 93	Slot 94	Slot 95	Slot 96	Slot 97	Slot 98	Slot 99
DL	DL	DL	DL	UL	DL	DL	DL	DL	UL RA

Slot 100	Slot 101	Slot 102	Slot 103	Slot 104	Slot 105	Slot 106	Slot 107	Slot 108	Slot 109
DL	DL	DL	DL	UL	DL	DL	DL	DL	UL RA

Slot 110	Slot 111	Slot 112	Slot 113	Slot 114	Slot 115	Slot 116	Slot 117	Slot 118	Slot 119
DL	DL	DL	DL	UL	DL	DL	DL	DL	UL RA

Slot 120	Slot 121	Slot 122	Slot 123	Slot 124	Slot 125	Slot 126	Slot 127	Slot 128	Slot 129
DL	DL	DL	DL	UL	DL	DL	DL	DL	UL

Slot 130	Slot 131	Slot 132	Slot 133	Slot 134	Slot 135	Slot 136	Slot 137	Slot 138	Slot 39
DL	DL	DL	DL	UL	DL	DL	DL	DL	UL

Slot 140	Slot 141	Slot 142	Slot 143	Slot 144	Slot 145	Slot 146	Slot 147	Slot 148	Slot 149
DL	DL	DL	DL	UL	DL	DL	DL	DL	UL

Slot 150	Slot 151	Slot 152	Slot 153	Slot 154	Slot 155	Slot 156	Slot 157	Slot 158	Slot 159
DL	DL	DL	DL	UL	DL	DL	DL	DL	UL

Figure 2.4: TDD pattern 4:1.

Mathematical Background

3.1 Regression Fitting

3.1.1 Linear Least Squares

When modeling a set of data as a linear polynomial function, linear least squares fitting can be applied. Depending on the underlying data, different formulations can be applied. The most common estimator, Ordinary Least Squares (OLS), minimizes the sum of the squared residuals of a model, assuming uniform variance and no bias in the data. More formally, for n samples of input data X with corresponding output data y, an overdetermined linear system can be constructed as,

$$\begin{pmatrix} X_1 & 1 \\ X_2 & 1 \\ \vdots & \vdots \\ X_n & 1 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}. \tag{3.1}$$

With the general model as,

$$\mathbf{y} = \mathbf{X}\beta + \epsilon, \tag{3.2}$$

where $\epsilon \in N(0, \sigma^2)$. Approximating β, (given there is no exact solution to the over determined system in (3.1)), simplifies to minimizing the residual vector

$$||\mathbf{y} - \mathbf{X}\beta||^2, \tag{3.3}$$

with respect to β. The problem has a unique closed-form expression, $\hat{\beta}$, which can be obtained by solving, [14]

$$(\mathbf{X}^T\mathbf{X})\hat{\beta} = \mathbf{X}^T\mathbf{y} \Leftrightarrow \hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}. \tag{3.4}$$

The OLS estimator can also be used for interpolating higher order polynomials, for example quadratic expressions such as,

$$f(x, \beta) = \beta_1 x_1^2 + \beta_2 x_1 + \beta_3. \tag{3.5}$$

3.1.2 Goodness of Fit Metrics

In order to evaluate how well a model applies to a set of observations, it is desirable to use some sort of evaluation metric. One simple metric is to observe the the sum of residual squares for the generated model, which sums all of the errors for the data points as,

$$SS_{res} = \sum_i (y_i - f_i(x, \beta))^2. \tag{3.6}$$

This is one of the most basic metrics of model fit, where a high sum of squared residuals indicates worse model fit compared to low sum of squared residuals. But it can be deceptive. Since residuals nearing zero often means that the model is overfitted to the data. Therefore, another metric taking into account how well the model accounts for the variation in the data is needed. The R^2 statistic tries to do just that, by indicating the proportion of total variation around the mean that can be attributed to the regression model.

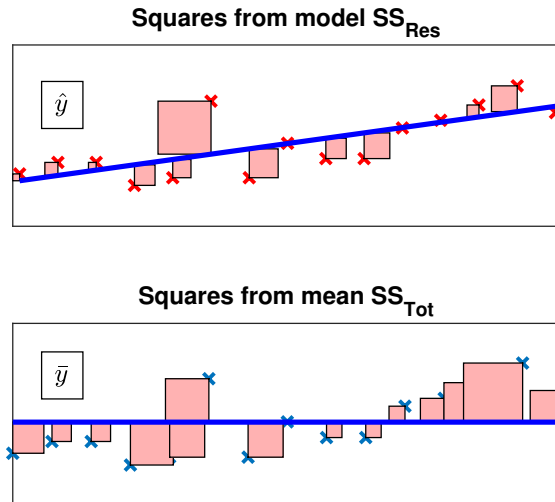

Figure 3.1: Top Figure shows the residual squares from the fitted regression model on a sample from Fishers 1936 Iris dataset, lower Figure shows residuals from mean values on the same data.

More formally, it is defined as,

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}, \tag{3.7}$$

where SS_{res} is the sum of squared residuals from the generated model and SS_{tot} is the summed squared errors around the mean. Intuitively, a value close to one would mean that a significant amount of the variation can be explained by the model. Figure 3.1 displays the different quantities SS_{res} and SS_{tot} for Fisher's 1936 Iris data set, and is intended to give the reader better intuition on what the R^2 metric actually measures.

3.1.3 Nonlinear Least Squares

Wanting to fit a nonlinear model, for instance a bimodal distribution on a data set, is a well studied problem in the field of regression analysis. Since the sought after model is nonlinear, there seldom exists a closed-form expression for the parameter estimation, and often iterative, numerical methods converging to a local solution have to be used. More formally, for the set of n data points $(x_1, x_2, \ldots, x_n)$, with corresponding output data points $(y_1, y_2, \ldots, y_n)$, there exists a set of parameters $(\beta_1, \beta_2, \ldots, \beta_n)$ which minimizes the residual vector between the model function $f(x, \beta) = y$ for the given data points, where the individual residuals are defined as $r_i = y_i - f(x_i, \beta)$.

Specifically for MATLAB's nonlinear least squares solver, it uses an algorithm called the Trust-Region method [15], iteratively finding better parameter values by stepping inside a trust-region, and evaluating the function value at each step, finally converging at a solution which minimizes the loss function (residual vector in this case). Mathematically the problem can be formulated as the iteration, [16]

$$\min\{\frac{1}{2}s^T H s + s^T g, \text{ such that } ||Ds|| \leq \Delta\}, \tag{3.8}$$

where the loss function has been approximated with the second order Taylor expansion using matrix equivalents. H and g denotes the Hessian matrix [17] and gradient to f at point x with step matrix s. D is a diagonal scalar matrix, and Δ is a positive scalar, defining the boundaries of the trust region. The minimization is then given by solving

$$||s|| - \Delta = 0, \tag{3.9}$$

finding the eigenvalues of the Hessian, and then applying Newtons method for step direction. The optimization stops after a predefined error threshold is reached.

3.2 Kernel Density Estimation

To estimate a probability density function from sampled data the non-parametric approach of kernel density estimation can be used. The underlying principle is

straightforward, for all $x \in \mathbb{R}$, weigh the surrounding data points with a kernel function K, to generate a continuous probability density function. More formally, the estimated probability distribution at point x, given data $[x_1, x_2, \ldots, x_i]$ is,

$$\hat{f}_h(x) = \frac{1}{nh} \sum_{i=1}^{n} K(\frac{x - x_i}{h}),$$ (3.10)

where K is the kernel function and h the bandwidth. The choice of kernel function is arbitrary, and is generally a smooth, unimodal function where $\int K(t)dt = 1$, to ensure that the area under the function is unity. Of all the various kernels available, Gaussian kernels are most commonly used. Choosing a proper bandwidth is key to generating a good distribution, since choosing a bandwidth that is too narrow will create an undersmoothed distribution and vice versa. B. W. Silverman [18] introduces an adequate choice for initial bandwidth selection as,

$$h_{\mathrm{opt}} = 0.9 \cdot \min(\sigma, \mathrm{IQR}/1.34)n^{\frac{-1}{5}},$$ (3.11)

where σ is the standard deviation of the samples, IQR is the interquartile range (mid spread) and n is the number of samples. This selection strikes a good balance in being able to properly smooth distributions regardless of uni- or bimodality in the underlying distribution, this is however no guarantee for optimal bandwidth.

3.3 Kullback-Liebler Divergence

To validate an approximation of an underlying distribution some sort of metric is desirable. The Kullback-Liebler Divergence (KLD) [19] (or relative entropy) provides an indicator of the information lost in an approximated distribution compared to the underlying distribution. For the discrete distributions $P(x)$ and $Q(x)$, KLD is defined as,

$$D_{KL}(p||q) = \sum_{x \in \mathcal{X}} P(x) \log \frac{P(x)}{Q(x)}.$$ (3.12)

Due to Gibbs' inequality [20], the KLD is always nonnegative and only zero if $P(x) = Q(x)$, meaning a lower value indicates an approximation close to the underlying distribution. The divergence does not, however, generalize linear distance in the distribution space, and can therefore not be used as a metric indicating goodness of fit, but rather as a way to compare different approximations to each other. Also note that the divergence is not symmetric, meaning that $D_{KL}(P||Q) \neq D_{KL}(Q||P)$

Experimental Setup

4.1 User Equipment Simulator

To measure DSP load for different numbers of RBs on real DU hardware, an Ericsson internal UE simulator called Full Stack User Equipment (FSUE) was used. The FSUE uses multiple simulated components or Component Based Simulators (CSIM), together with real baseband hardware to realize a digital testing environment in the context of 5G development. The simulations are performed with ideal channels, that is, at virtually no loss, since all communication is done over physical cables rather than OTA interfaces. The testing environment allows for manually overriding system constants in order to achieve desirable testing conditions. Disabling link adaptation for baseband traffic allows the user to set an upper limit to the number of RBs as well as fix the MCS index. Currently only Non-standalone (NSA), where the 5G network depends on the LTE-A network to operate, is supported. Therefore the FSUE used for data collection was also configured as NSA. Extraction is only done on the gNB however, since only UL in NR is investigated in this thesis.

When measuring DSP load, the main interest is the processing load that originate from the main processing units of the baseband hardware, which are called Ericsson Many Core Architectures (EMCAs). The EMCA is a proprietary ASIC design, executing software in Ericsson basebands. It is characterized by a high number of processing cores performing tasks in parallel, enabling high capacity, which is critical in real time communication systems. Having a processing load that exceeds the EMCAs processing capacity, for example during a high traffic scenario, will result in poor system performance and increased latency, but throttling the system too much would not utilize full system capacity.

Figure 4.1 illustrates the different subsystems in CSIM FSUE. Being configured as NSA, the FSUE simulates all components connected to the eNB and gNB hardware required for full stack testing. Dashed lines indicate logical connection, thick lines indicate physical connection and interfaces are specified in bold. Among the simulated components, a traffic generator based on open source software TRex can be found connected to the simulated core network. From the core network,

17

connections to the NSA DUs are established. From these DUs, simulated radio interfaces are then connected. Finally, the simulated UE is connected via a L1 termination layer.
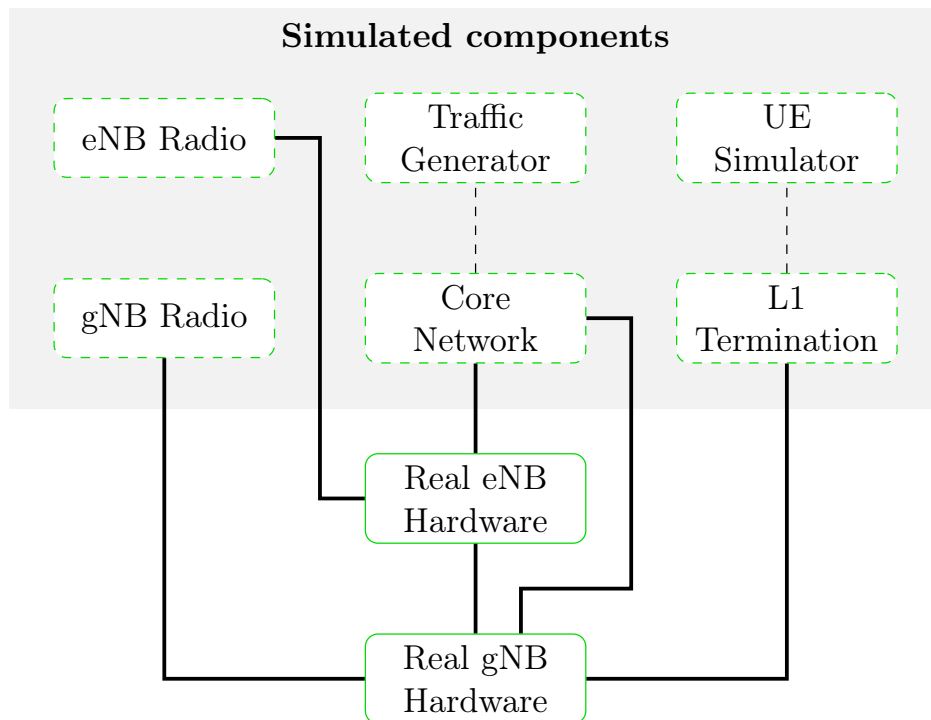

Figure 4.1: System overview of the Ericsson FSUE and its subsystems.

4.2   Ericsson Internal Simulator

The model from the measurements was incorporated into a Python simulator. This simulator was originally developed by Ericsson for examining different scheduling algorithms for DL SRs. Most of the code structure was kept but the simulator was rewritten to only schedule UL data.

Figure 4.2 shows a flowchart of the Python simulator. The simulator iterates on a TTI basis since the measured load from the DU in gNB was measured on a TTI basis. First, it runs the scheduling algorithm and updates the queues. Right after, the DSP load for each EMCA is calculated before the allocated EMCA resources (number of allocated UL sector carriers and their corresponding ID) are reset. Lastly it adds the SRs to the UEs before it starts over on the next iteration/TTI.

When calculating the load the simulator fetches the interpolated distribution parameters for the specified RB configuration and randomly generates a number with the Numpy function `random.normal`, which draws random samples from a

Gaussian distribution with specified parameters. In order to choose between the two Gaussian distributions, the interpolated amplitude determines the ratio of choosing each distribution by setting a threshold equal to the ratio. Then using the Numpy function `random.random`, a random float in the interval $[0.0, 1.0)$ is generated and a distribution is chosen based on the outcome.
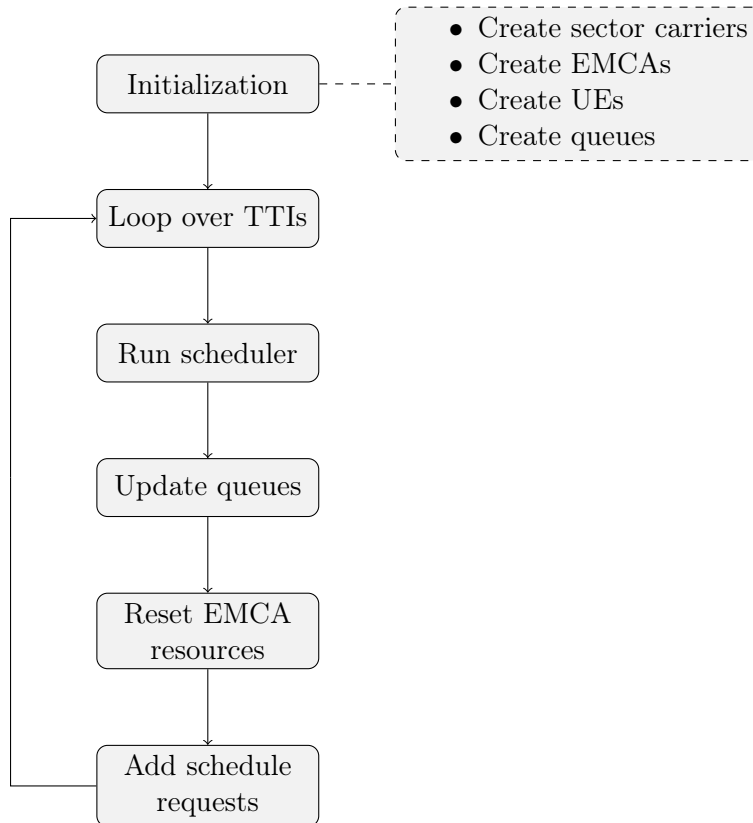

Figure 4.2: Flowchart of internal simulator.

Methods

This chapter describes the different techniques and approaches used in generating the results and models of this thesis.

5.1 User Equipment Simulator

5.1.1 Choice of Measurements

In order to accurately model the behaviour of the load when changing the number of RBs, a proper data set had to be collected. The initial decision was to investigate points spaced with power of two, that is, 2, 4, 8, 16, et cetera. Realizing this did not fully encapsulate the behaviour of the load distribution, evenly spaced points on the interval along with points at the outer edges of the range of RBs were added. For these RB values, a collection of data from one, two and three sectors was performed. Higher sector count is theoretically possible, but due to limitations in extracting DSP load on the gNB accurately for higher counts, this thesis will be limited to the collected sector configurations.

To evaluate the performance of the constructed model, validation measurements were also collected. These measurements were taken at a low, medium and high number of RBs, aiming to verify performance representative of the entire RB range.

5.1.2 Comparing User Equipment Simulator Against Real UEs

To verify realistic performance of the simulated environment, data from another test node with real UEs instead of simulated was UEs used. This node had actual UE hardware with OTA transmission in a controlled environment. In theory the channel conditions could vary but due to the controlled environment, the difference was in practice immeasurable.

5.1.3 Traffic Scenarios

Three different traffic scenarios was investigated. Too high DSP load in the system is undesirable since it can result in the system running out of computational re-

sources and might generate undefined behaviour, therefore a worst case scenarios was sought. Since the link adaptation algorithm could interfere with this scenario it was determined that it should be disabled and also that the MCS index in Table 2.1 should be set to 27 (highest possible MCS value), maximizing the throughput and thus, the load. Because of the flexibility in the system constants two additional scenarios were investigated: One where LA was enabled and another where LA was disabled with a 10 % block error rate. Since LA is enabled by default it was of interest to see what difference disabling it would have. With link adaptation enabled, the AMC algorithm, which allocates different MCS values, should affect the throughput and thus the DSP load in a nondeterministic way. 10 % BLER was investigated to see how a "real-life" scenario, where the UE has to retransmit data, could affect the DSP load in any way. For all traffic scenarios, only one UE per sector was used to generate the traffic.

5.1.4   Simulator Output Analysis

In the real gNB hardware described in Section 5.1, a Logic Analyzer Tool (LAT), which tracks and logs program events, was used to record DSP usage. The tool could separate when individual threads start and stop and on which DSP and write it to a LAT file. Measurements on the hardware were done by logging the mean load for the duration of a TTI slot (0.125 μs for numerology three) as time series data on one of the systems' EMCAs.

The load measured was a total utilization of the processing capacity, meaning all threads generated, regardless of origin, affected the total system load. Using a tool called RealTimeTraces (RTT), the data could be filtered by name specific threads. This enabled two things: extraction of only the UL generated load in L1, which is the main interest of this thesis. Secondly, extraction of the load generated by the signal synchronization blocks, which occurs at predetermined slots, in order to accurately synchronize measured load and compare the data on a TTI basis.

5.2   Formulation of Statistical Model

The data acquired from the custom end-to-end test was used to create a statistical model capable of interpolating UL generated DSP load based on system settings, such as the number of RBs and sector configuration. Initially, MATLAB was used to study the occurring patterns and load distributions in the data by using plots and statistical methods such as data variance, mean value etc. Later on, a system was constructed and used for the duration of the project, which consisted of the two following main parts:

- A parser, which reads the data from generated files, synchronizes them according to a predefined SSB pattern and outputs desired model-parameters.

- A validation script, generating diagnostic plots and testing the generated model against defined validation data.

In order to generate a model, probability distributions were first created for the different load measurements by generating kernel density estimations as described in Section 3.2. A Gaussian model was then fitted on the resulting distribution using nonlinear regression techniques described in Section 3.1.3. The statistical parameters of these distributions were then interpolated according to linear or cubic methods described in Sections 3.1. Both metrics describing goodness of fit as well as visual inspection of formulated models were used to determine validity. The model parameters could then be exported to be used in the internal simulator.

Verifying the simulated values, the KLD between verification data and simulated data was calculated in order to get an overview of modelling performance. This was done by binning the samples in order to calculate distribution probability, and then using (3.12).

5.3   Ericsson Internal Simulator Parameters

Table 5.1 lists the parameters given to the simulator. The parameter Max UL is the maximum allowed allocated UL Sector Carrier (SC) per EMCA. Each SR occupies a whole cell and the UEs were always created with a "Full buffer" type of traffic, that is, occupying all available SCs in a sector. With only one SC per sector, this meant that a maximum of 4 sectors with UL data per EMCA was possible. To validate the model, the simulator was given three different RB values of which the model had not been formulated upon.

Number of iterations:	15 000
Number of sectors:	1, 2 or 3
SCs per sector:	1
Max UL:	4
UEs per sector:	1
Number of EMCAs:	1
Cell BW:	50 or 100 MHz
Number of allocated RBs:	Low, mid and high

Table 5.1: Internal simulator parameters.

Results and Discussion

In this Chapter, trends of DSP loads for different RB configurations are analyzed, for one sector and a 100 MHz cell. It would have been advantageous to also show absolute numbers in the results, but this is unfortunately not possible due to commercial reasons. For the complete set of Figures over all configurations, see Appendix A.

6.1 Comparing Real Versus Simulated UEs

Upon testing the difference in DSP load between real and simulated UEs, it was deemed to be quite small, with mean values being only 1.7 % higher for the real UEs. When discussing this with others at Ericsson, their experience was that the difference would be reduced even further when the number of UEs is increased. From this measurement along with the empirical experience from others, the assumption is made that the simulated results are realistic enough to make conclusions regarding the performance of real UEs.

6.2 Investigating Real-Life Scenarios

Running the test with LA enabled resulted in lower DSP load. This was expected since the LA algorithm controls, among other things, the MCS index and it was also the case in the previous works of [1] and [2]. Lowering the MCS affects the throughput and thus the DSP load. Specifically which other parameters the LA algorithm changed to lower the DSP load than MCS was not investigated. With LA disabled and a 10 % BLER enabled the difference in load was insignificant. Both scenarios were only investigated on a mean basis for the whole data and further analyses were not pursued.

6.3 Initial User Equipment Simulator Results

After initial gathering of DSP load over a number of different RB configurations in cell bandwidth, the data were structured and analyzed. Using MATLAB, mean

values for every data set were calculated in order to interpolate a linear model
for DSP load versus RBs. Figure 6.1 shows the measured mean load and the
interpolated linear model for a one sector, 100 MHz cell bandwidth configuration.
The linear regression lines aligns well with the measured mean values indicating a
linear relationship between the DSP load and number of RBs. This is similar to
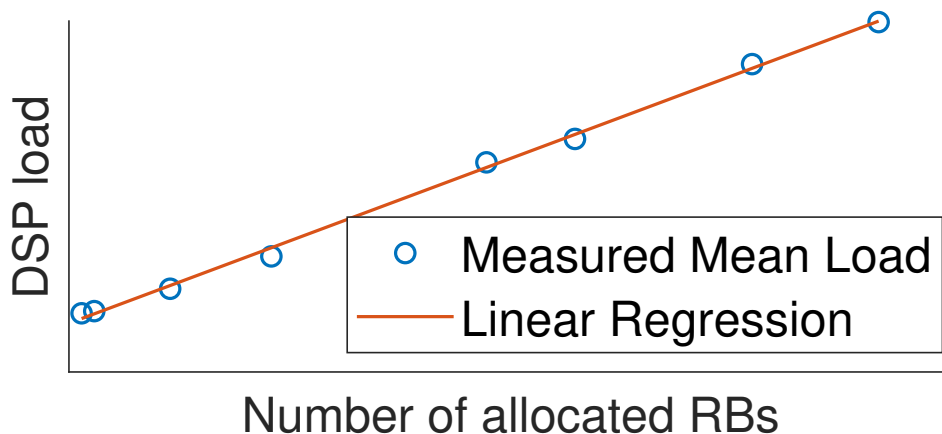what the previous works in [1] and [2] found for DL data.

Figure 6.1: Measured mean load and linear regression versus RBs
for one sector and a 100 MHz cell configuration.

To investigate how the load differed for all sector configurations the load was
divided by the number of sectors and plotted in the same Figure which can be
seen in Figure 6.2. For a higher number of sectors, the DSP load was slightly
lower for a low number of RBs but slightly higher for a high number of RBs
compared to one sector. Expected results would be perfectly linear, since there
is a linear increase in number of threads with similar duration with increasing
sector configuration, meaning linear increase in processing load in an EMCA. The
slight mismatch between regression lines is believed to be random, since the linear
regression is fitted to a selection of points affected by stochastic error terms.

With the assumption that the mismatch between the regression lines are random,
lowering the number of RBs for cells should decrease the load linearly even for a
multisector configuration. Therefore, it should be possible to support more cells,
and thus more CUs, by lowering the number of RBs.

100 MHz BW cell configuration

50 MHz BW cell configuration
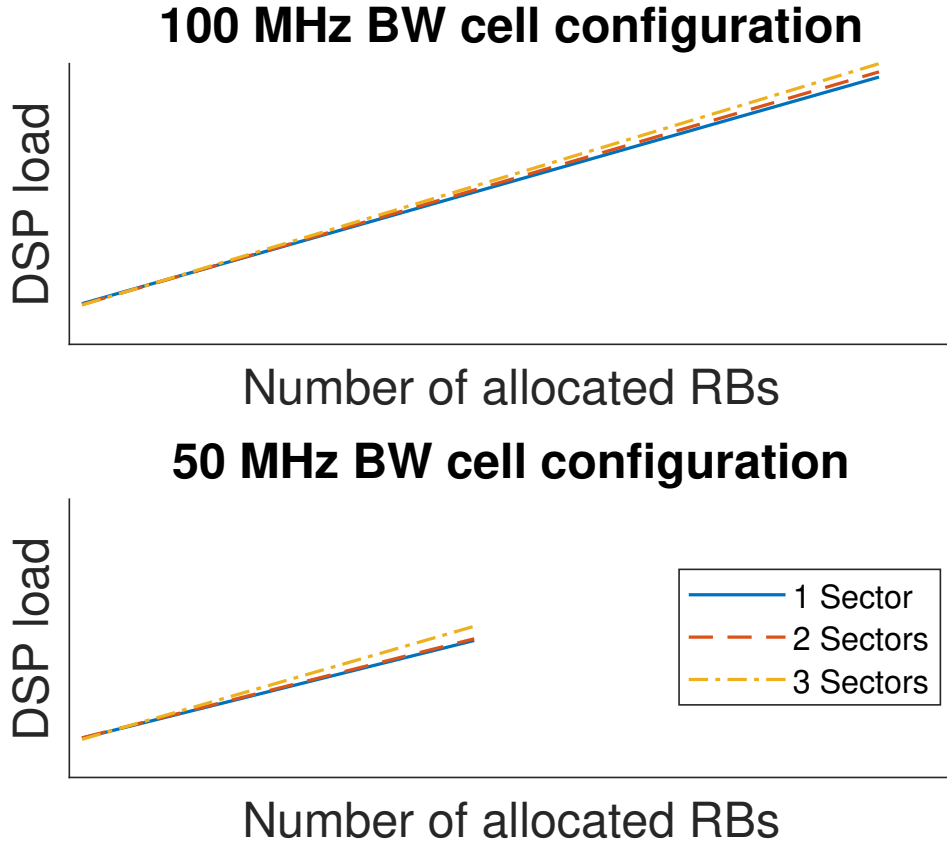
Figure 6.2: DSP load for different sector and cell bandwidth configurations.

6.4 Distribution of UE Simulator Measured Data

Further analysis of the load distribution by plotting the data points in a histogram indicated that there is a collection of different underlying distributions as seen in Figure 6.3. This meant that the previous approach of generating mean values for all of the data in order to model DSP load excluded a lot of the characteristics in the underlying data. Upon inspecting the time series data, the origin of these distributions became clear, as a very distinct periodicity emerged. A sample of the time series data can be seen in Figure 6.4.

To verify the periodicity of the data, the frequency contents was investigated with spectral density estimation. Since the load was sampled over one TTI, which is $T = 0.125$ μs in numerology three according to Table 2.2, the sampling frequency is $f_s = 8$ kHz. Figure 6.5 shows two peaks, one at $f_1 = 1.6$ kHz and another smaller at $f_2 = 3.2$ kHz. Since $f_s/f_2 = 2.5 \notin \mathbb{Z}$ and $f_2 = 2f_1$, the second peak is a harmonic caused by the discrete-time Fourier transform generating amplitude for all integer harmonics N with frequencies N/T, the only frequency of interest
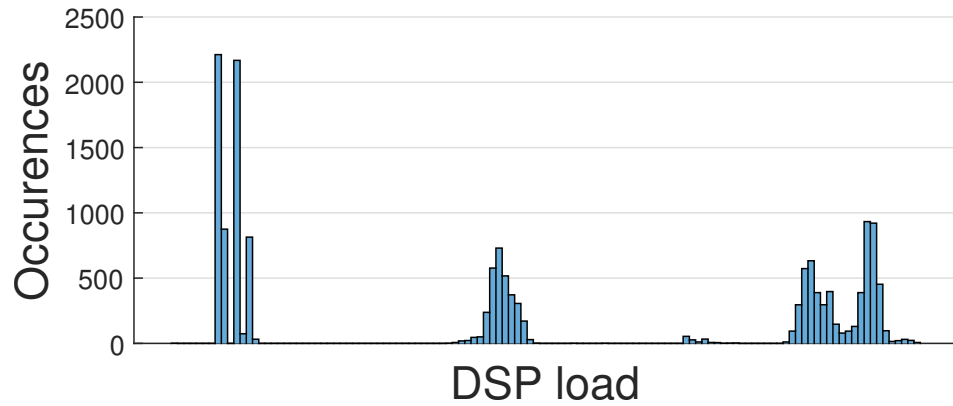
Figure 6.3: Histogram for DSP load measurements for one sector
and a 100 MHz cell configuration.
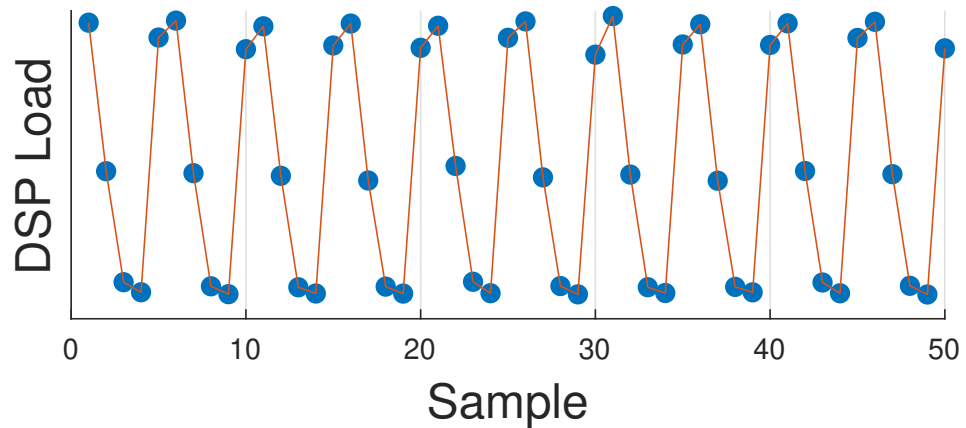
Figure 6.4: DSP load in the time domain for one sector and a 100
MHz cell configuration.

is 1.6 kHz. This frequency aligns well with the fact that the TDD pattern used is
a 4:1 pattern, meaning four DL slots followed by one UL slot, which results in the
pattern repeating after every fifth slot as in Figure 2.4.

Picking out every fifth element in the slot aligned data allowed isolation of the
different distributions in Figure 6.3. Figures 6.6 and 6.7 show the histograms for
the different slot types with a low and high number of RBs. Comparing the two
TDD separated distributions, a significant increase in DSP load for DL 1, DL 2
and UL can be observed with the increased number of RBs, while DL 3 and DL
4 remains unchanged. This was surprising, since most of the UL processing is
supposed to start in either UL or the first DL slot. This led to the belief that
the threads generated in the UL slot might affect the following slots in some way,
since the load is measured as average DSP core utilization for the duration of the
slot.
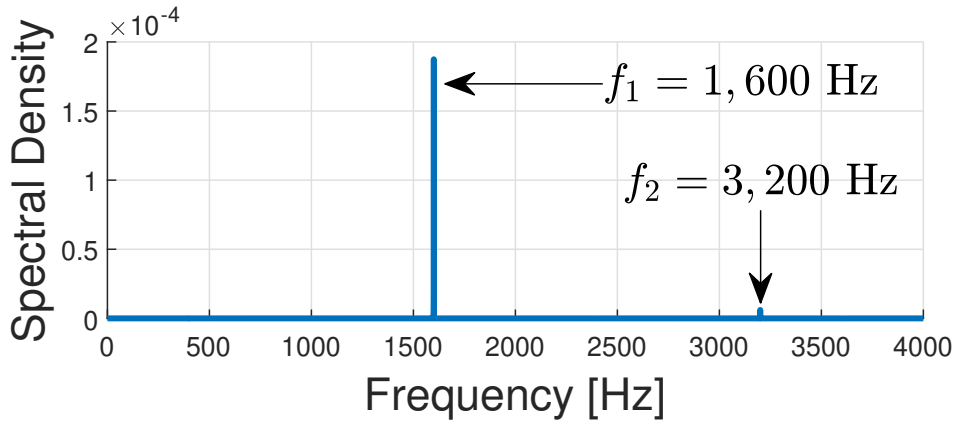
Figure 6.5: Perodicity of the DSP load data for one sector and a 100 MHz cell configuration.
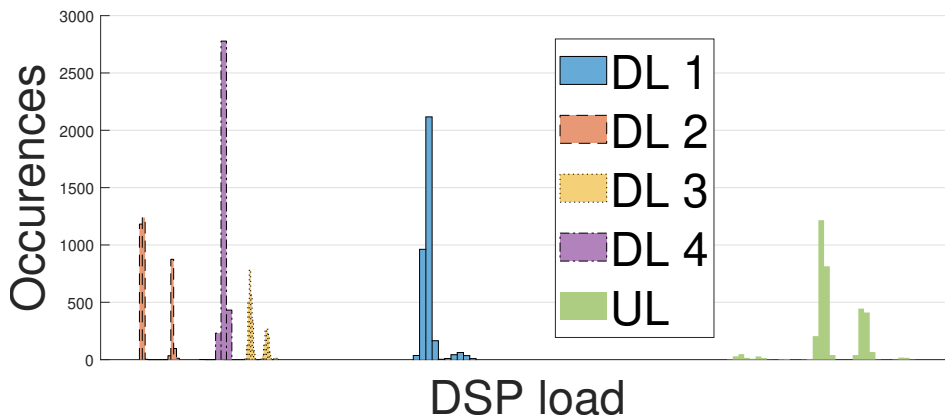
Figure 6.6: Histograms of DSP load for the different slot types over the 4:1 TDD pattern for low number of RBs.

Manually looking at the RTT logs in the form of a flamegraph verifies the hypothesis. The UL threads processing time extends from DL 1 into DL 2, causing a very small increase in load for a low number of RBs, but a substantial increase for high number of RBs. This also clarifies why the DL 1 slot has a higher load than the UL slot itself for a high number of RBs, as can be seen in Figure 6.7. Since the L1 processing does not start until some symbols into the UL slot, the average DSP core utilization is higher for DL 1 as a result of the full slot being occupied with threads generating UL load. This concept of processed threads having a duration long enough to occur during the following slot is illustrated in Figure 6.8. This extension effect causes a non-linear characteristic in generated load in the DL 2 slot, where it has a small, constant load up until a certain number of RBs, then it starts to increase.
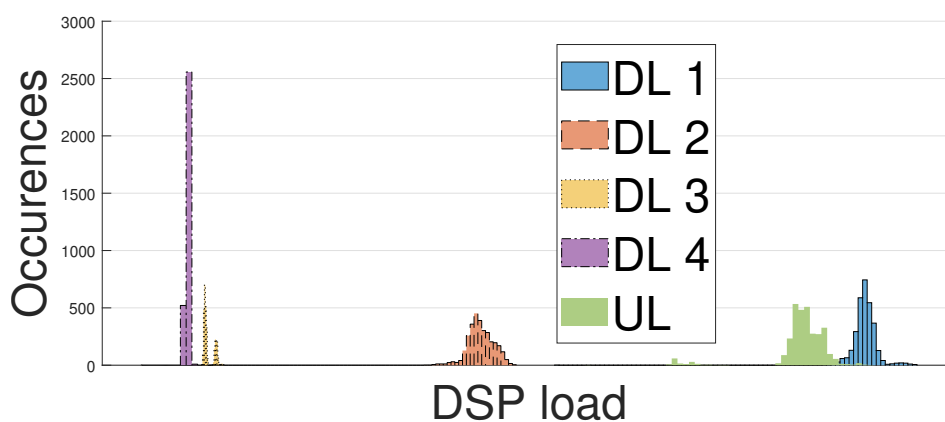
Figure 6.7: Histograms of DSP load for the different slot types over the 4:1 TDD pattern for high number of RBs.
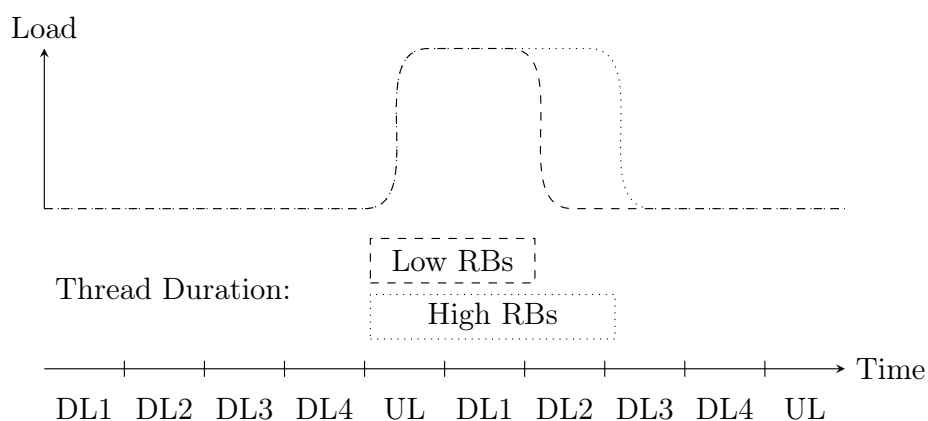
Figure 6.8: Illustration of the load and threads in the flamegraph.

6.5   Statistical Model Formulation

Having thoroughly inspected the generated load data, but now on a time slot basis, a pattern of mixture distributions started to emerge. Many of the generated histograms show that the data often have dual Gaussian peaks, but at other times they converge to unimodal distributions. See Figure 6.9 for six distributions showcasing this behaviour.

Looking at how the distributions changed for a fixed TDD slot when altering the number of RBs, a linear increase in distribution were observed in Figure 6.10. This led to the formulation of a statistical model for the distributions as follows: for a set number of RBs, interpolate a set of parameters for a mixture of two Gaussian distributions with the following continuous probability density function,

$$y(x) = a_1 e^{-\left(\frac{x-b_1}{c_1}\right)^2} + a_2 e^{-\left(\frac{x-b_2}{c_2}\right)^2}. \tag{6.1}$$

This model was chosen, partly because of the observed bimodality, but also for its ability to capture unimodal distributions, only requiring equal mean for both distributions ($b_1 = b_2$).

To fit the model onto the data, MATLAB's `fit` function from the curve-fitting toolbox was used. Applying the function directly onto the data yielded non-satisfactory results, due to noise. Therefore, kernel density estimation was applied to the data to estimate a smoothed distribution, as described in Section 3.2. Since the resulting distributions are very sensitive to kernel bandwidth, different values were tested by trial and error. No better value than the bandwidth described in (3.11) was found. The consequent continuous distributions could then be used for actual distribution fitting.

The MATLAB function `fit` could then, using the trust-region method described in Section 3.1.3, fit the formulated model in (6.1) to the data. By extracting the generated model parameters, a data set for interpolation was constructed (See Figure 6.13 for visualization of the fittings).

The final model was then created by interpolating the parameters for the distributions, that is, the amplitude (a_1, a_2), mean (b_1, b_2) and standard deviation ($\frac{c_1}{\sqrt{2}}, \frac{c_2}{\sqrt{2}}$) for the range of RBs over one, two and three sectors. Initial linear regression as described in Section 3.1 was made to investigate linearity. Linearity seemed to hold for most TDD slots, but upon inspection of characteristics of the second DL slot, a linear model was deemed incorrect because of the sudden load increase observed earlier. Therefore, linear parameter models were assumed for DL slots 1, 3, 4 and UL, but for DL 2, a second degree polynomial model was used instead.

Plots of mean and standard deviation along with goodness of fit metrics for one sector and a 100 MHz cell can be seen in Figures 6.11 and 6.12. In the plots, good R^2 values can be observed for the approximation in slots DL 1, 2, and UL, along with very small residual errors. For the mean load in slots DL 3 and 4, the scores drop significantly. This is to be expected, since the observed values for slots DL 3 and DL 4 were static regardless of RB configuration. In some cases it would have almost certainly sufficed using mean values for these slots instead.

These measurements showcase the model on collected data for one sector at 100 MHz, but generally the same behaviour was observed for two and three sectors. The result for these multi-sector cases can be seen in Section A.5.

For the 50 MHz cell, the same behaviour occurs predominantly, meaning the same interpolation model could be used. Upon further inspection of the data at 50 MHz, an unexpected outlier with low UL load at the theoretical maximum RB was observed in Figure A.21, skewing the model and also opposing the initial assessment of linearity. When investigating the impact on throughput, it was deemed negligible, which was unexpected. Redoing the load measurement confirmed this was

not a contingency. Discussing this outlier with Ericsson employees, no apparent reason could be found.
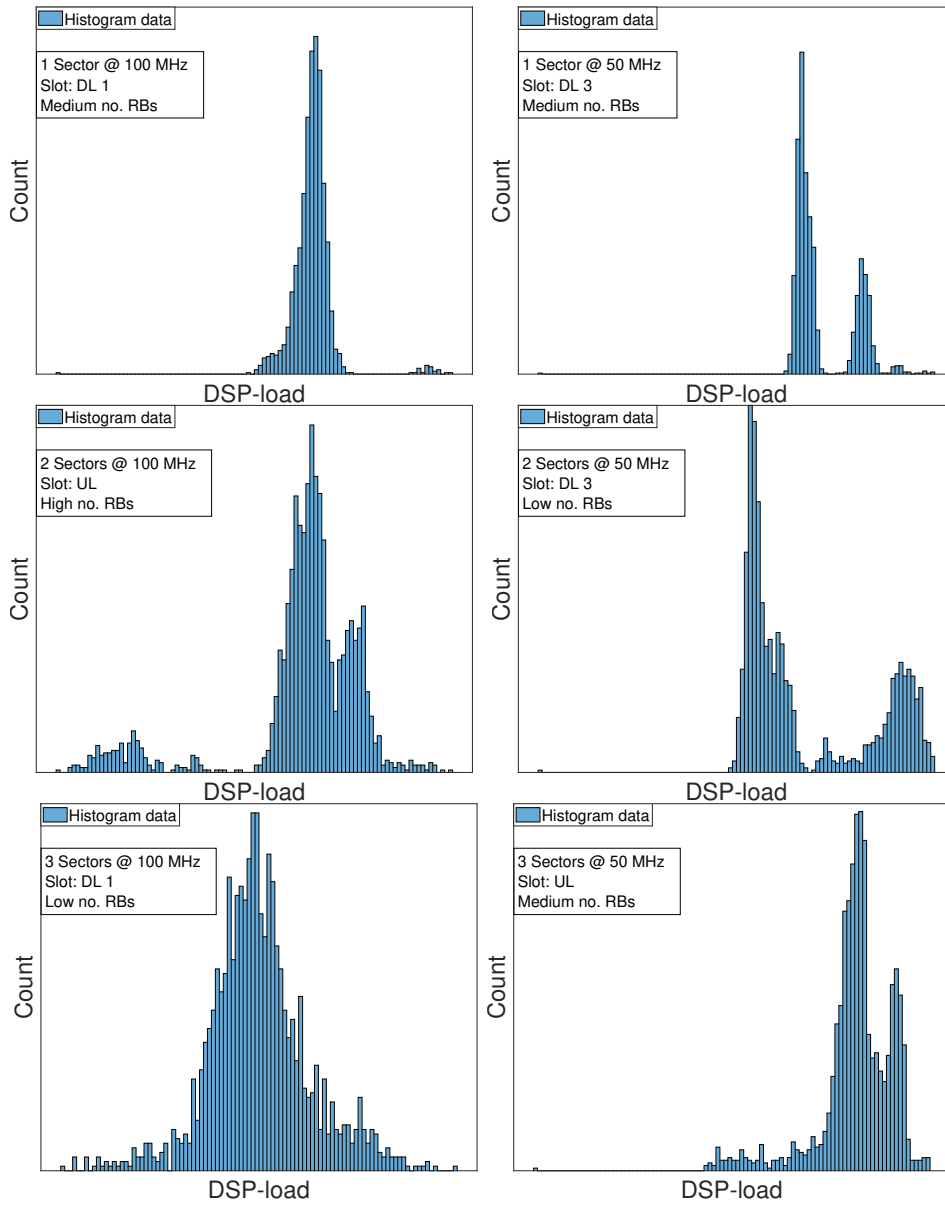
Figure 6.9: A collection of randomly picked histograms of DSP load
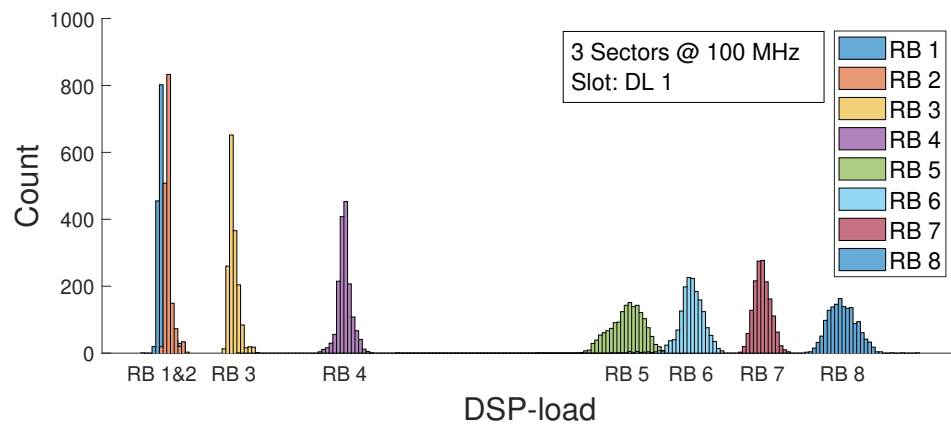data for all measured sector configurations.

Figure 6.10: Distributions from full range of RB-configurations for
TDD slot DL 1, note that the values for RBs are not evenly
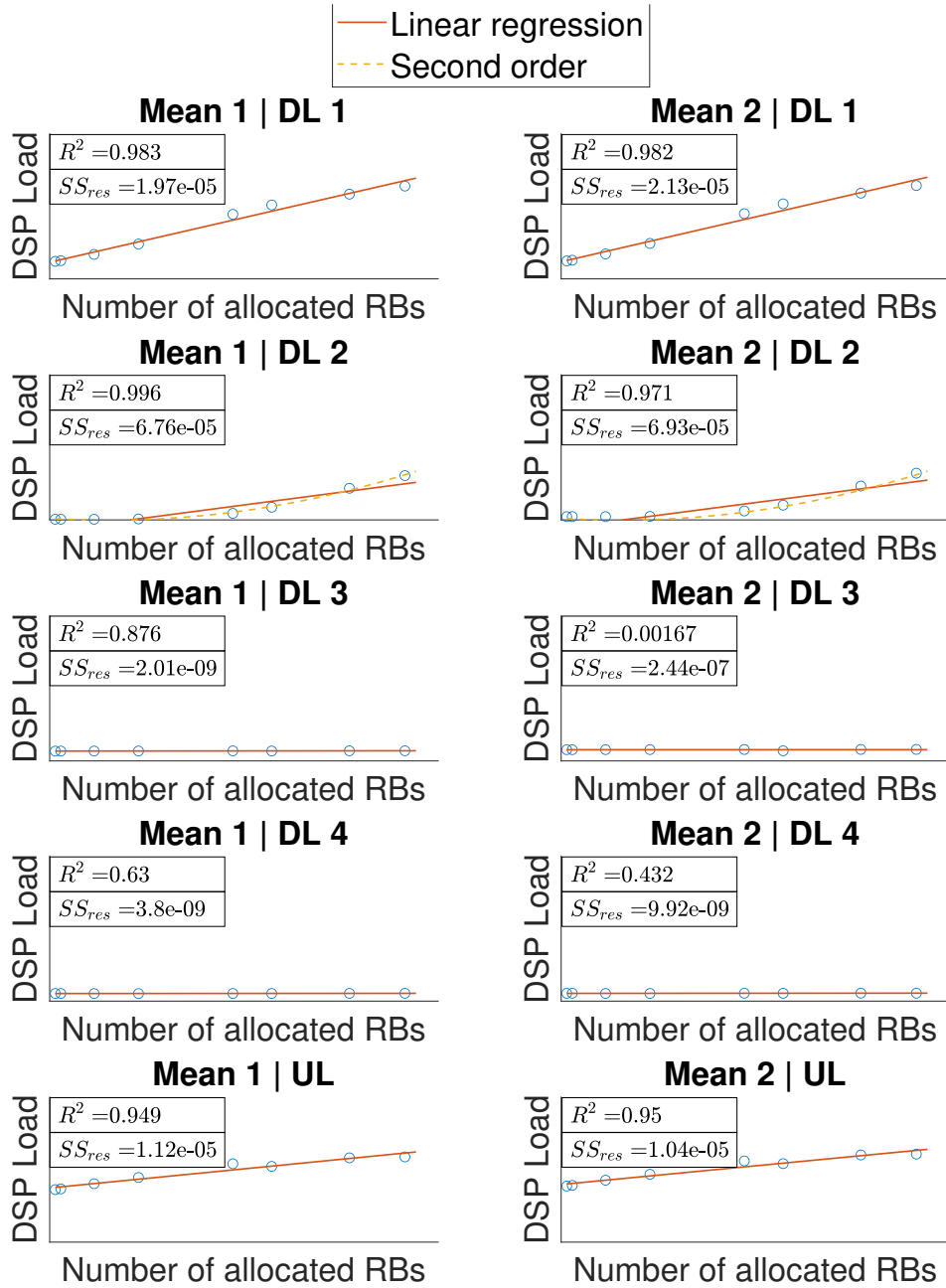spaced but in increasing order

Figure 6.11: Measured mean data and fitted curves for the mean
parameter over the 4:1 TDD pattern. In the Figure all the
subplots share the same DSP load axis.
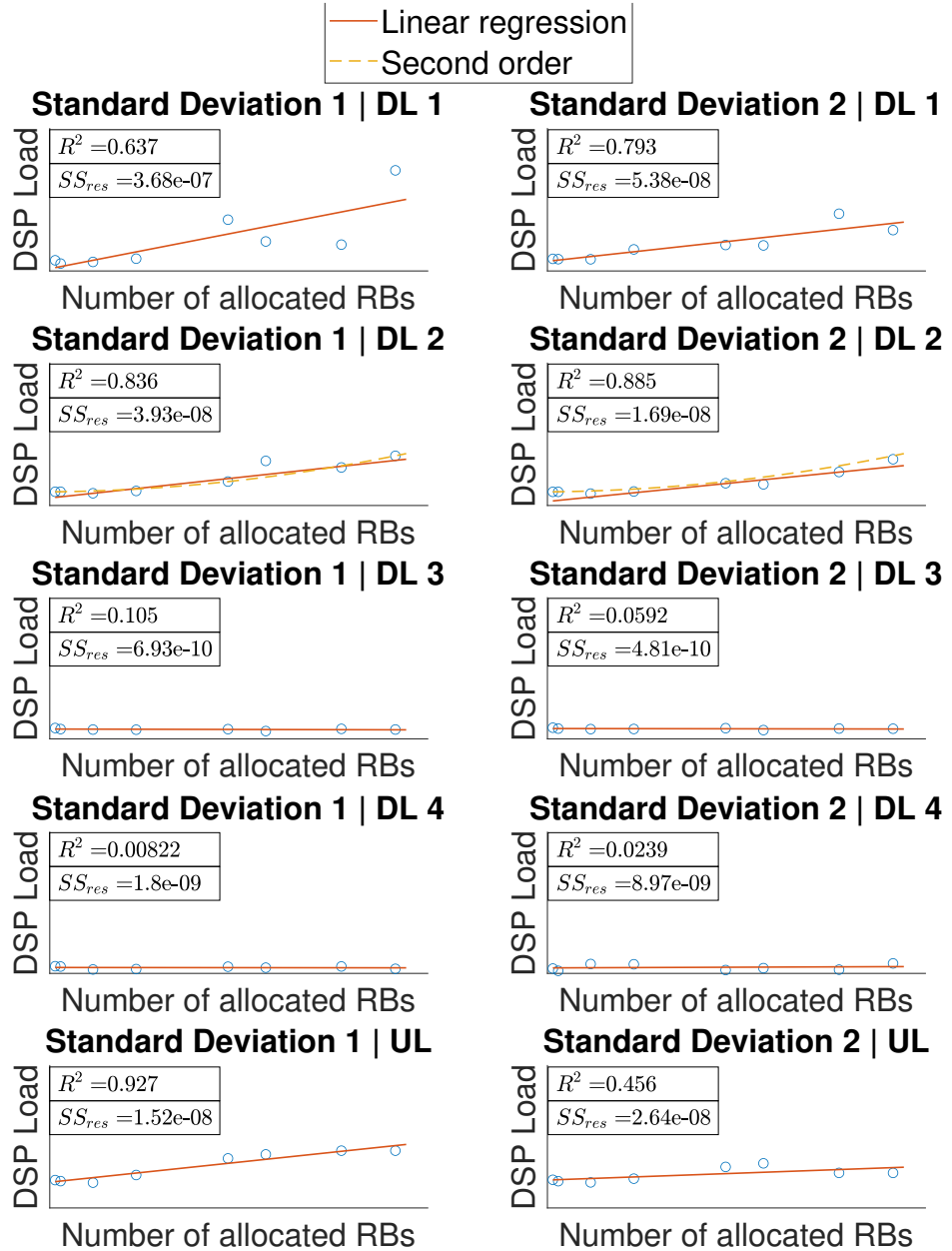
Figure 6.12: Measured mean data and fitted curves for the standard
deviation parameter over the 4:1 TDD pattern. In the Figure
all the subplots share the same DSP load axis.
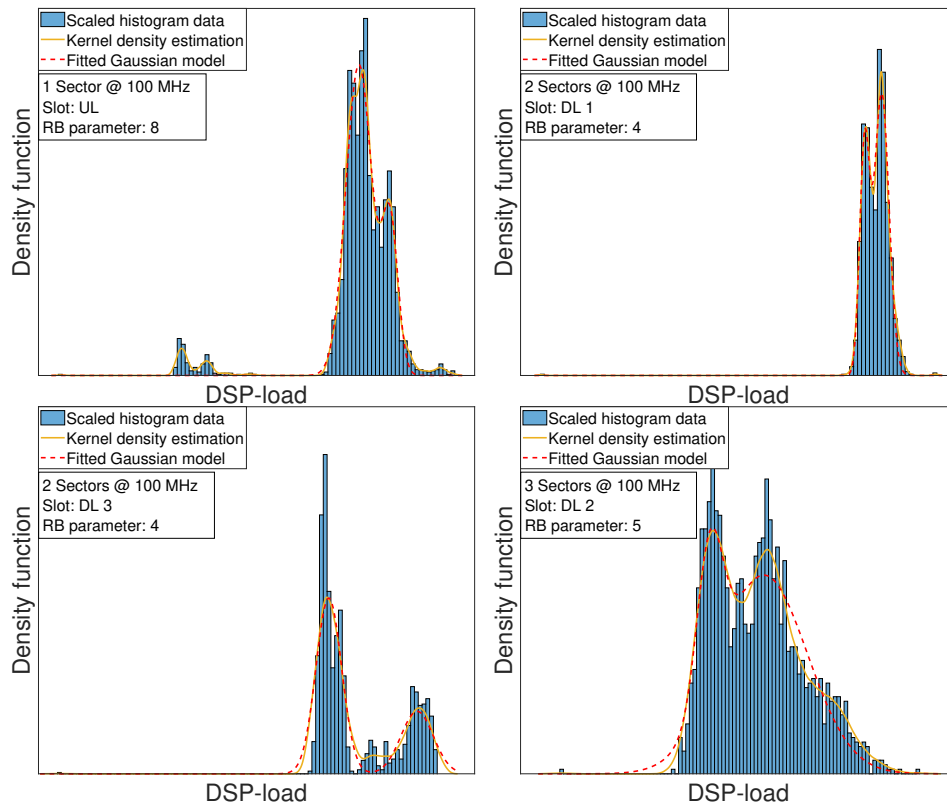
Figure 6.13: Four generated distributions, showcasing the kernel density estimation as well as our model fitting for 100MHz

6.6   Comparing Model Generated and Real Data

Running the simulator with the parameters given in Table 5.1 and processing the data in the same way as in Figures 6.6 and 6.7 enabled comparisons between the simulator and the real data. The resulting comparison between simulated and validation load distribution for a high number of RBs, 100 MHz and a mid number of RBs, 50 MHz can be seen in Figures 6.14 and 6.15. The observed simulated data performs as expected, with general distributions matching relative placement as in the validation data, with slight offsets in both shape and mean value. The outliers also disappears as expected, since the model is incapable of picking up smaller peaks. The same behaviour holds for both plots.
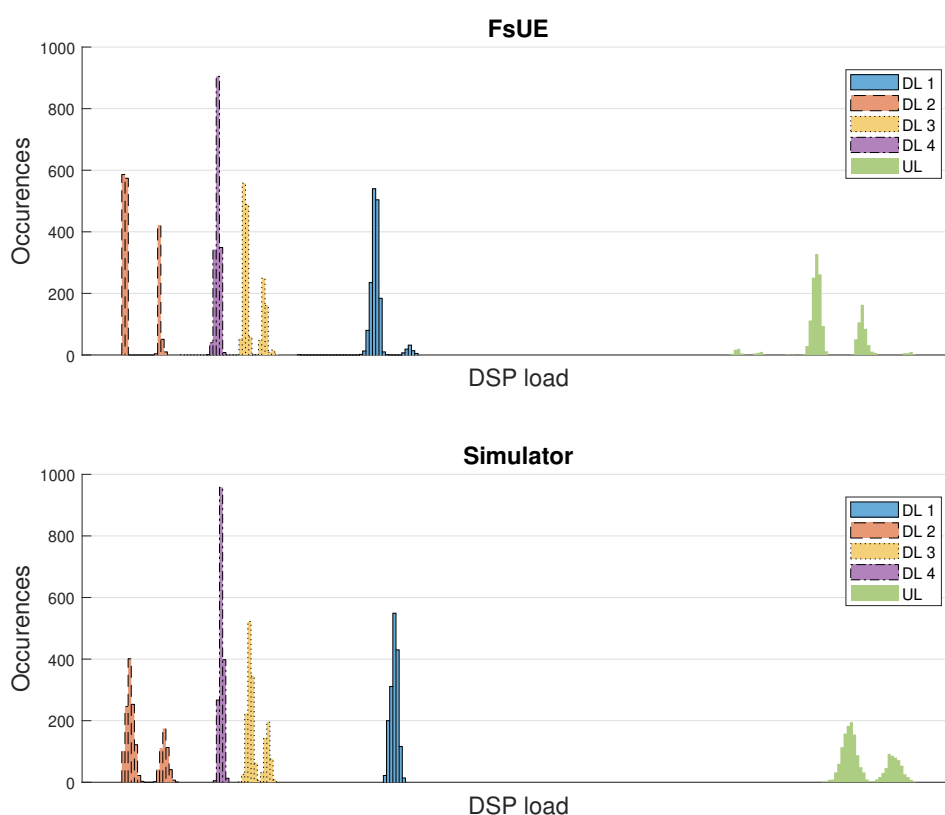
Figure 6.14: Histograms for different slot types over the 4:1 TDD pattern of DSP load from the simulator and FSUE. Plotted for a 50 MHz cell bandwidth and mid RB configuration. The DSP load axis are the same for both plots.

Since the third and fourth DL slots are almost constant throughout changing the number of allocated RBs, further analysis of behaviour when increasing RBs was performed on slots DL 1, 2 and UL. Picking three verification data samples showcasing the behaviour, the results are shown in Figures 6.16, 6.17 and 6.18. Generally good approximation of the actual load distributions were observed as
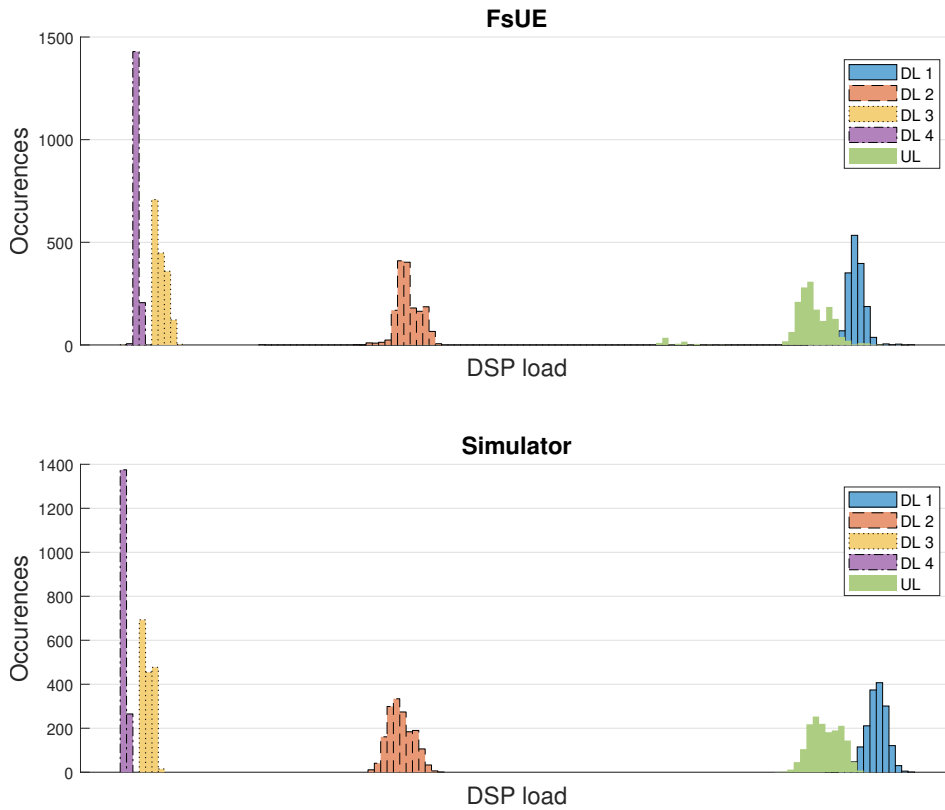
Figure 6.15: Histograms for different slot types over the 4:1 TDD
pattern of DSP load from the simulator and FSUE. Plotted for
a 100 MHz cell bandwidth and high number of RBs. The DSP
load axis are the same for both plots.

the the RBs were increased. Figure 6.16 specifically showcases the model's ability
to go from a mixture of two Gaussian distributions, to what almost looks like
unimodal, single Gaussian distribution.

There are however simulation results indicating that the assumption of linearity
in distribution means is not entirely true. Looking at Figure 6.18, it can clearly be
seen how the simulated loads fail to encapsulate the actual validation loads in terms
of distribution placement, although the shape of the distribution seems accurate.
This is caused by the very small standard deviations of these distributions, meaning
that even slight deviations in mean value might result in our model completely
falling outside of the validation distribution.

6.6.1   Measuring Validity of the Model

Having analyzed the previous results, it should be noted that they only represent
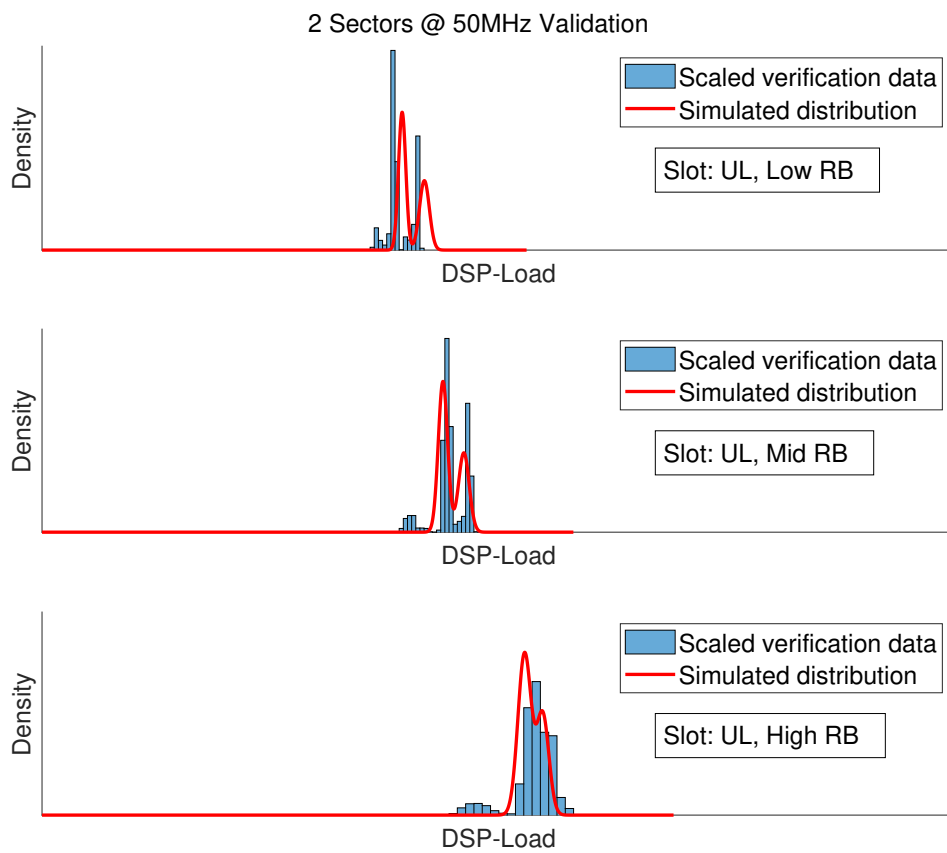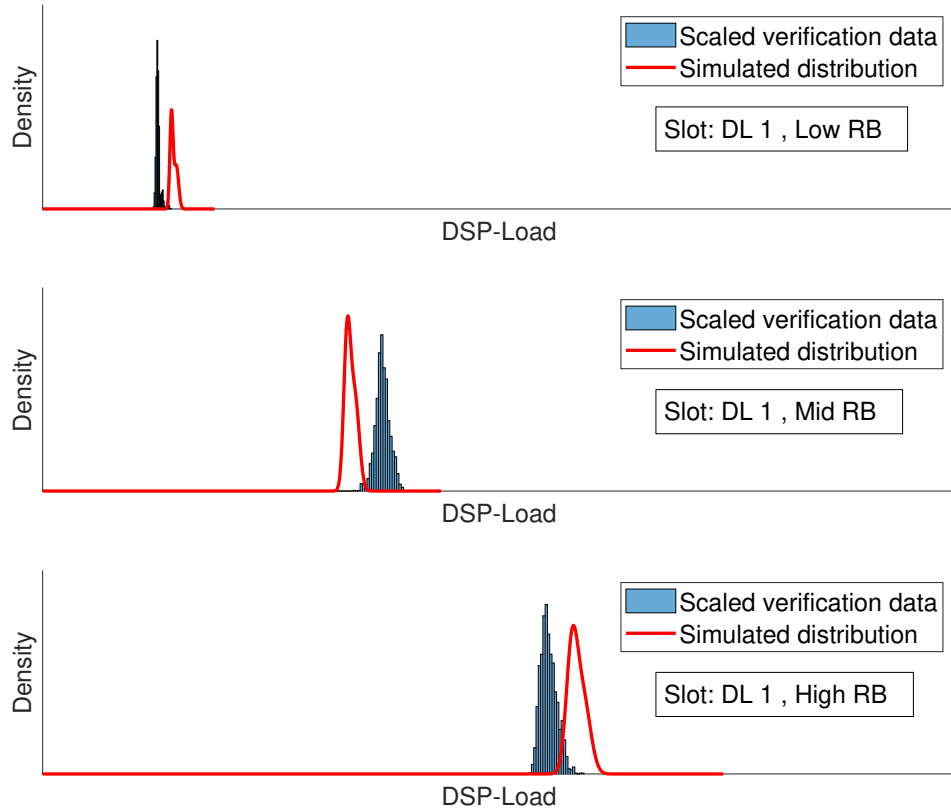a small subset of all the validation measurements taken. In order to make a
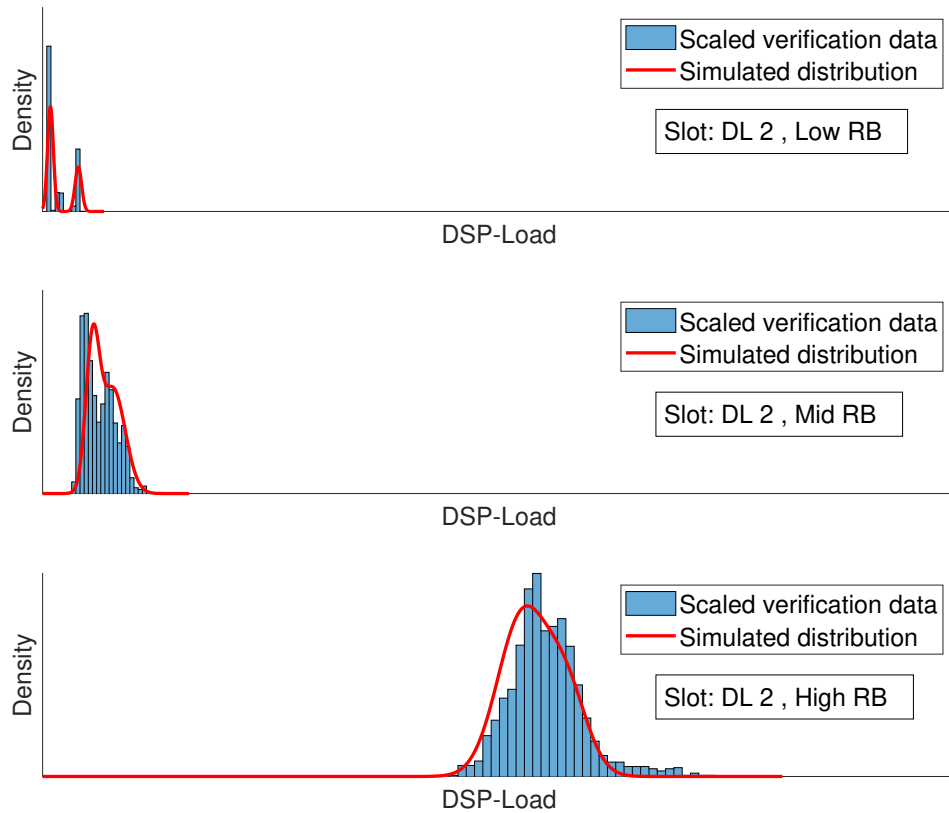
Figure 6.16: Distribution of the validation and simulated data for the different RB configurations taken on a 100 MHz cell for slot DL 2.

bigger comparison of all the gathered data, the divergence described in Section 3.3 was calculated for all the distributions obtained in the validation data and corresponding simulated data. The results can be seen in Tables 6.1 and 6.2.

For the first DL slot, the KLD values seem to change arbitrarily for different sector and RB configurations. For a 50 MHz cell and one sector, the approximation fits worst for the mid and best for the low number of RBs. Whereas for two sectors, the worst fit is for the low and the best is for the high number of RBs. The highest KLD values probably occur when the simulated distribution does not align well with the verification data as in Figure 6.18.

For both the 50 and 100 MHz cells the third and fourth DL slot seem to score most of the low KLD values. This is expected since both the mean and standard deviation for those slots are more or less constant making them much easier to model. Generally, a pattern where the simulated distributions of interest (DL 1, DL 2 and DL 3) score lower for the lower RBs was observed. This could be due to the load distributions having much lower standard deviations for low number

Figure 6.17: Distribution of the validation and simulated data for
the different RB configurations taken on a 50 MHz cell, for slot
UL.

of RBs measurements, as seen in Figure 6.12, thus making it more unlikely for the
model to overlap and causing larger divergence. This is further indication that the
hypothesis on linearity for the model parameters is incorrect.

Since the discrete distributions are approximated by binning samples when calcu-
lating the Kullback-Liebler, some of the distribution characteristics could be lost,
and affect the resulting values.

Figure 6.18: Distribution of the validation and simulated data for the different RB configurations taken on a 100 MHz cell, for slot DL 1.

Slot	1 Sector			2 Sectors			3 Sectors		
	RBs			RBs			RBs		
	Low	Mid	High	Low	Mid	High	Low	Mid	High
DL 1	0.869	18.392	1.181	7.998	6.975	0.288	2.735	4.633	1.798
DL 2	5.929	7.852	2.180	8.590	3.885	0.545	6.085	1.536	0.235
DL 3	4.098	7.296	16.917	2.795	1.281	1.640	1.213	10.163	1.019
DL 4	0.467	0.198	8.840	7.348	0.267	2.361	0.748	0.035	5.655
UL	5.496	9.420	0.677	6.776	2.287	0.451	2.696	0.543	0.212

Table 6.1: Kullback-Leibler divergence values for cells with 50 MHz bandwidth and different sector and RB configurations.

| Slot | 1 Sector | | | 2 Sectors | | | 3 Sectors | | |
| | RBs | | | RBs | | | RBs | | |
	Low	Mid	High	Low	Mid	High	Low	Mid	High
DL 1	17.521	12.733	3.452	4.461	4.242	4.843	17.488	16.013	8.031
DL 2	6.935	14.496	0.156	10.888	1.301	0.182	9.690	0.189	0.078
DL 3	0.397	0.775	1.897	0.673	0.512	1.257	3.584	0.610	1.448
DL 4	0.028	0.052	0.136	0.048	0.026	0.020	0.010	0.063	0.112
UL	16.490	7.160	0.814	8.050	9.048	1.254	2.033	1.416	1.127

Table 6.2: Kullback-Leibler divergence values for cells with 100 MHz bandwidth and different sector and RB configurations.

6.7   Improvements and Future Work

In order to avoid hitting the limit for processing load, the results show that for
the 4:1 TDD pattern, the UL generated load is generated in the UL and the two
following DL slots. In an ideal scenario, the events that take place in one TTI
should take only one TTI or less to process to avoid a processing debt into the
next TTIs. This "spillover" effect of processing continuing into other TTIs that
has been observed could pose a problem in the future for another pattern with
several consecutive UL slots after each other.

There also needs to be considerations of how the DL load behaves, which has not
been considered in this work. It is desirable to smooth out the load over all the
TTIs and put most of the DL processing in slots where the UL processing is low.
An obvious drawback to this approach is that the latency would suffer, especially
in a case where there are a lot of connected UEs requesting resources. Otherwise,
housekeeping task should be performed in slots with low amount of prioritized
processing which is the third and fourth DL slot for the 4:1 TDD pattern and only
considering UL load.

For future work, the load for different TDD pattern combinations could be studied
for both DL and UL data. It would also be interesting to verify if the results from
the simulator with a different TDD pattern would match data from a real node.
All the UEs in this work has also been of the full buffer type. To be able to more
safely draw conclusions and explore how many CUs there could be with a lower
number of RBs available, studies of how the load would behave if the amount of
UEs is increased but the total load over all UEs is kept the same as for one full
buffer UE should be carried out, similar to what [1] has done.

It is also not certain that for a higher number of RBs, possible in a 200 or 400
MHz cell, that the load or processing will extend over to the third or fourth DL
slot. Because of the software design, where tasks are performed in parallel, the
load could still be generated over three slots but at a higher utilization.

Conclusions

In this thesis the behaviour of how the DSP load scales with the number of RBs assigned per cell has been investigated for UL data, both in a single and multisector configuration. The findings indicate that it is possible to support a larger number of cells, and thus more CUs, by assigning a lower number of allocated RBs per cell. Further analyzing the load showed that it should be considered on a TDD pattern basis, where it could then be approximated as a linear or quadratic polynomial function of the number of allocated RBs. Analyzing the load on a TDD pattern basis revealed that the distribution of the load was most often uni- or bimodal in character.

From the collected data a model was then developed and the parameters implemented into an internal simulator. Verification measurements from the simulator shows that the generated distribution parameters are too stochastic to be modeled as strictly linear or quadratic polynomial function. As can be seen in Figure 6.9, the different plots contains many outliers that cannot be explained without a more thorough understanding of the Ericsson software execution. The results can however be seen as a well-founded estimation and could be used in the internal Ericsson simulator to explore other scenarios, or investigate potential improvements.

To get a full understanding of the DSP load behaviour, DL needs to be investigated as well. If it was also implemented in the simulator, it would enable more conclusions on the design choices that need to be made in future versions of the software.

References

[1] Somreeta Pramanik, Adlen Ksentini, and Carla-Fabiana Chiasserini. "Characterizing the Computational and Memory Requirements of Virtual RANs". In: *CoRR* abs/2201.00673 (2022). arXiv: `2201.00673`. URL: `https://arxiv.org/abs/2201.00673`.

[2] Tuyen X. Tran, Ayman Younis, and Dario Pompili. "Understanding the Computational Requirements of Virtualized Baseband Units Using a Programmable Cloud Radio Access Network Testbed". In: *2017 IEEE International Conference on Autonomic Computing (ICAC)*. 2017, pp. 221–226. DOI: `10.1109/ICAC.2017.42`.

[3] 3GPP. *NR; Base Station (BS) radio transmission and reception.* Technical Specification (TS) 38.104. Version 17.5.0. 3rd Generation Partnership Project (3GPP), Apr. 2022. URL: `https://portal.3gpp.or` `g/desktopmodules/Specifications/SpecificationDetails.aspx` `?specificationId=3202`.

[4] Erik Dahlman, Stefan Parkvall, and Johan Sköld. "Chapter 3 - OFDM Transmission". In: *4G LTE/LTE-Advanced for Mobile Broadband*. Ed. by Erik Dahlman, Stefan Parkvall, and Johan Sköld. Oxford: Academic Press, 2011, pp. 27–44. ISBN: 978-0-12-385489-6. DOI: `https:` `//doi.org/10.1016/B978-0-12-385489-6.00003-5`. URL: `https:` `//www.sciencedirect.com/science/article/pii/B978012385489` `6000035`.

[5] 3GPP. *NR; Physical layer procedures for data.* Technical Specification (TS) 38.214. Version 17.0.0. 3rd Generation Partnership Project (3GPP), Jan. 2022. URL: `https://portal.3gpp.org/desktopmodul` `es/Specifications/SpecificationDetails.aspx?specification` `Id=3216`.

[6]   3GPP. *NR; Physical Channels and modulations.* Technical Specification (TS) 38.211. Version 17.0.0. 3rd Generation Partnership Project (3GPP), Jan. 2022. URL: `https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3213`.

[7]   3GPP. *Study on new radio access technology Physical layer aspects.* Technical Report (TR) 38.802. Version 16.8.0. 3rd Generation Partnership Project (3GPP), Sept. 2017. URL: `https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3066`.

[8]   3GPP. *Study on new radio access technology Radio interface protocol aspects.* Technical Report (TR) 38.804. Version 16.8.0. 3rd Generation Partnership Project (3GPP), Mar. 2017. URL: `https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3070`.

[9]   3GPP. *NR; User Equipment (UE) radio transmission and reception; Part 2: Range 2 Standalone.* Technical Specification (TS) 38.2101-2. Version 17.4.0. 3rd Generation Partnership Project (3GPP), Jan. 2022. URL: `https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3284`.

[10]   Erik Dahlman, Stefan Parkvall, and Johan Sköld. "Chapter 7 - Overall Transmisison Structure". In: *5G NR: the Next Generation Wireless Access Technology.* Ed. by Erik Dahlman, Stefan Parkvall, and Johan Sköld. Academic Press, 2018, pp. 103–131. ISBN: 978-0-12-814323-0. DOI: `https://doi.org/10.1016/B978-0-12-814323-0.00006-5`. URL: `https://www.sciencedirect.com/science/article/pii/B9780128143230000065`.

[11]   Erik Dahlman, Stefan Parkvall, and Johan Sköld. "Chapter 6 - Radio-Interface Architecture". In: *5G NR: the Next Generation Wireless Access Technology.* Ed. by Erik Dahlman, Stefan Parkvall, and Johan Sköld. Academic Press, 2018, pp. 73–102. ISBN: 978-0-12-814323-0. DOI: `https://doi.org/10.1016/B978-0-12-814323-0.00006-5`. URL: `https://www.sciencedirect.com/science/article/pii/B9780128143230000065`.

[12]   3GPP. *NR; NR and NG-RAN Overall description; Stage-2.* Technical Specification (TS) 38.300. Version 16.8.0. 3rd Generation Partnership Project (3GPP), Dec. 2021. URL: `https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3191`.

[13] Erik Dahlman, Stefan Parkvall, and Johan Sköld. "Chapter 16 - Initial Access". In: *5G NR: the Next Generation Wireless Access Technology*. Ed. by Erik Dahlman, Stefan Parkvall, and Johan Sköld. Academic Press, 2018, pp. 311–334. ISBN: 978-0-12-814323-0. DOI: `https://doi.org/10.1016/B978-0-12-814323-0.00016-8`. URL: `https://www.sciencedirect.com/science/article/pii/B9780128143230000168`.

[14] N.R. Draper and H. Smith. *Applied Regression Analysis*. Wiley Series in Probability and Statistics. Wiley, 1998, 135–139. ISBN: 978-0-47-117082-2.

[15] A.R. Conn, N.I.M. Gould, and P.L. Toint. *Trust Region Methods*. MPS-SIAM Series on Optimization. Society for Industrial and Applied Mathematics, 2000. ISBN: 9780898714609.

[16] Jorge J. Moré and D. C. Sorensen. "Computing a Trust Region Step". In: *SIAM Journal on Scientific and Statistical Computing* 4.3 (1983), pp. 553–572. DOI: `10.1137/0904038`. eprint: `https://doi.org/10.1137/0904038`. URL: `https://doi.org/10.1137/0904038`.

[17] Wikipedia contributors. *Hessian matrix — Wikipedia, The Free Encyclopedia*. `https://en.wikipedia.org/w/index.php?title=Hessian_matrix&oldid=1086536756`. [Online; accessed 31-May-2022]. 2022.

[18] B.W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis, 1986. ISBN: 9780412246203.

[19] S. Kullback and R. A. Leibler. "On Information and Sufficiency". In: *The Annals of Mathematical Statistics* 22.1 (1951), pp. 79–86. DOI: `10.1214/aoms/1177729694`.

[20] D.J.C. MacKay et al. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 2003. ISBN: 9780521642989.

Additional figures

A.1 Linearity figures for additional sector and cell BW configurations
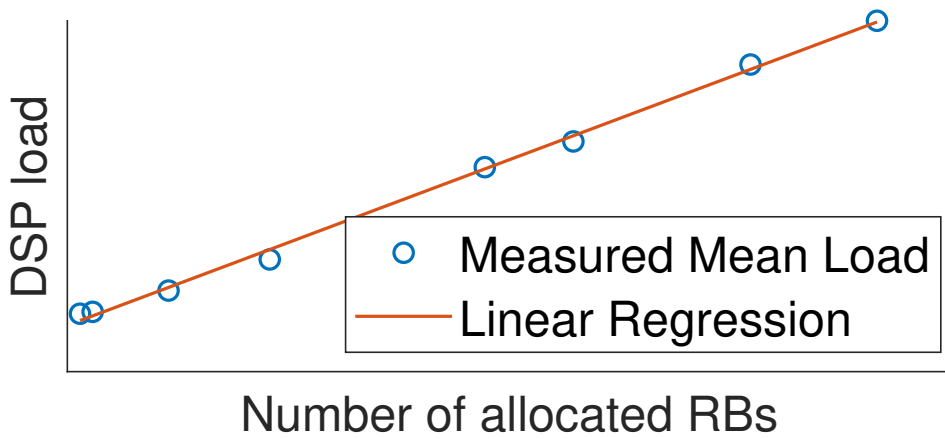
Figure A.1: Measured mean load and linear regression versus RBs for 100 MHz 2 sectors.
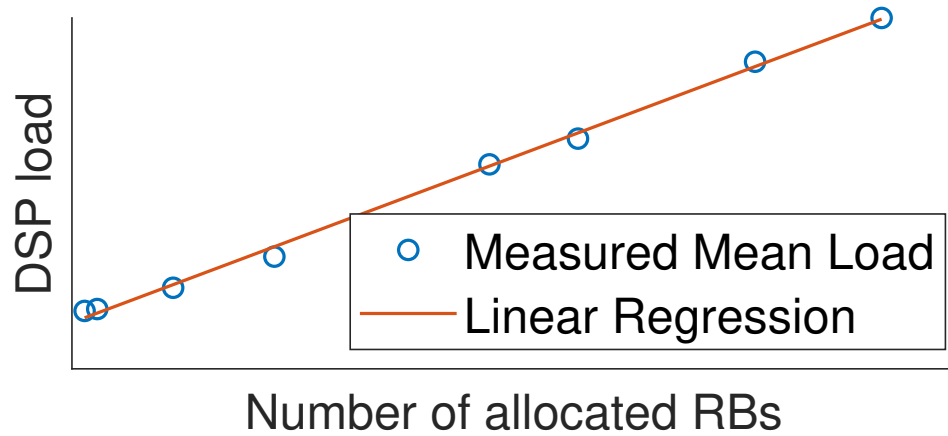
Figure A.2: Measured mean load and linear regression versus RBs for 100 MHz 3 sectors.
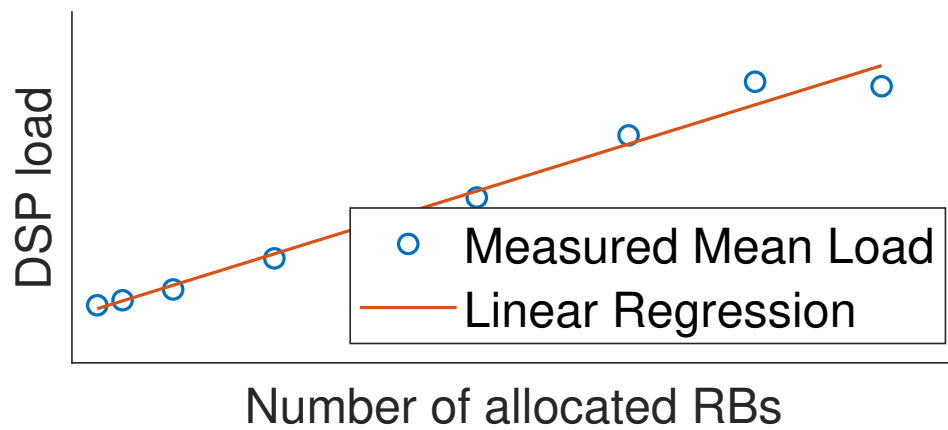
Figure A.3: Measured mean load and linear regression versus RBs for 50 MHz 1 sector.
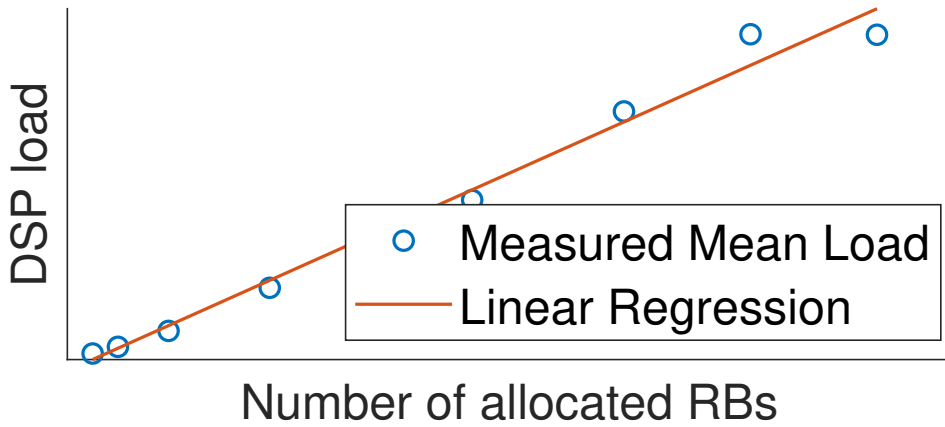
Figure A.4: Measured mean load and linear regression versus RBs for 50 MHz 2 sectors.
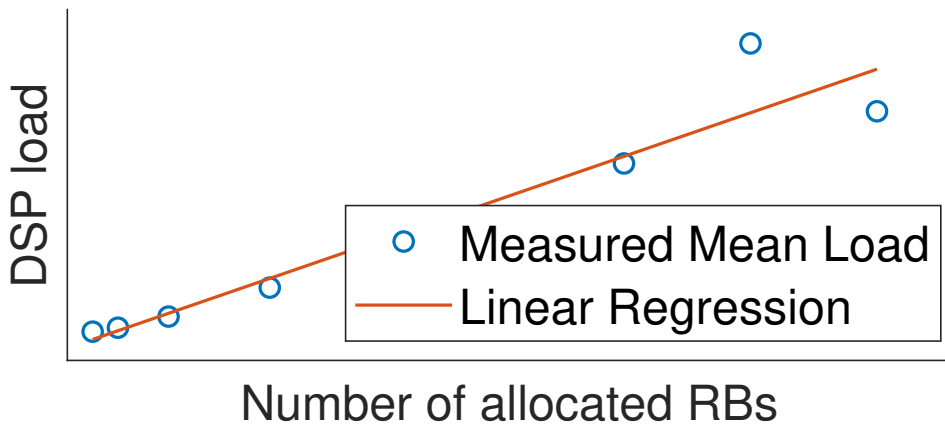
Figure A.5: Measured mean load and linear regression versus RBs for 50 MHz 3 sectors.

A.2 Histogram figures for additional sector and cell BW configurations
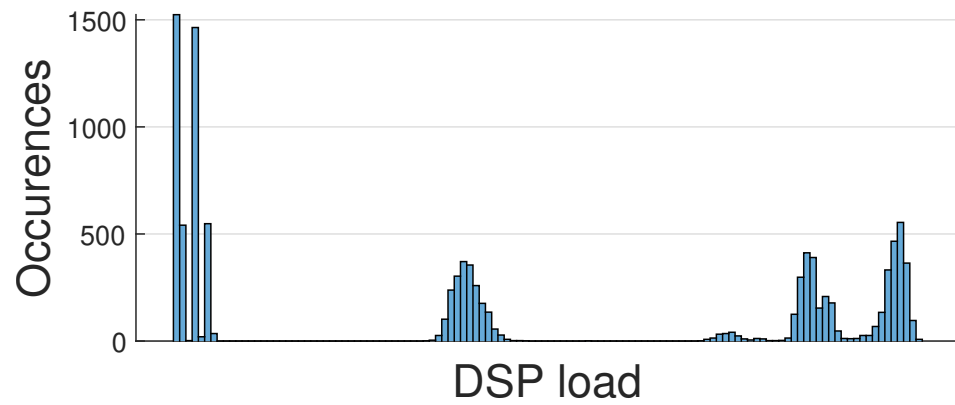
Figure A.6: Histogram for DSP load measurements. Two sectors and a 100 MHz cell.
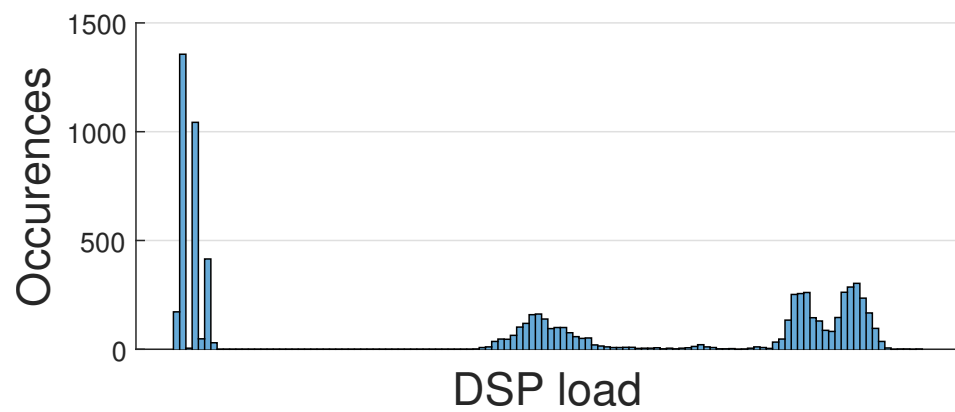
Figure A.7: Histogram for DSP load measurements. Three sectors and a 100 MHz cell.
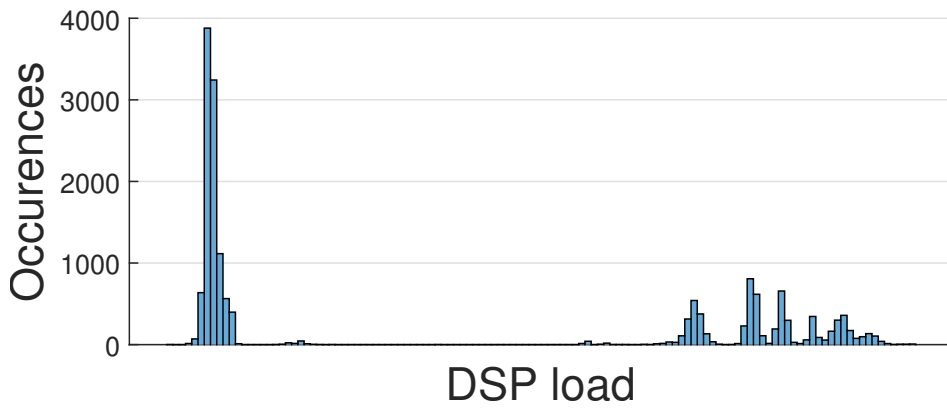
Figure A.8: Histogram for DSP load measurements. One sector and
a 50 MHz cell.
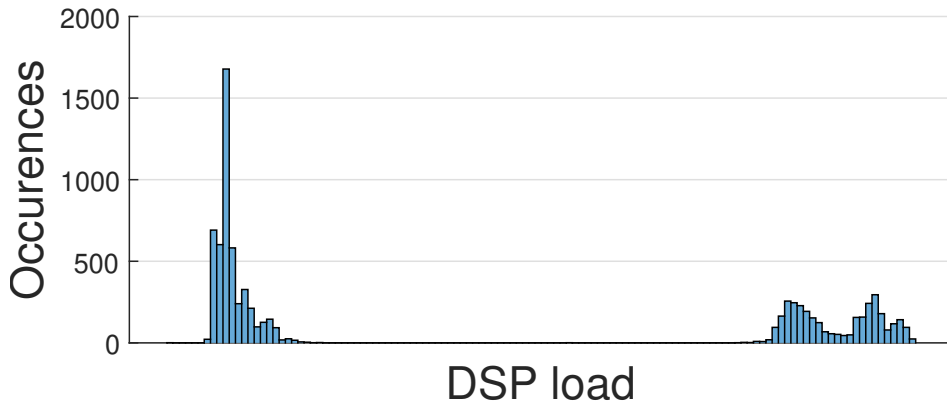
Figure A.9: Histogram for DSP load measurements. Two sectors
and a 50 MHz cell.

Figure A.10: Histogram for DSP load measurements. Three sectors
and a 50 MHz cell.

A.3 Time series figures for additional sector and cell BW configurations
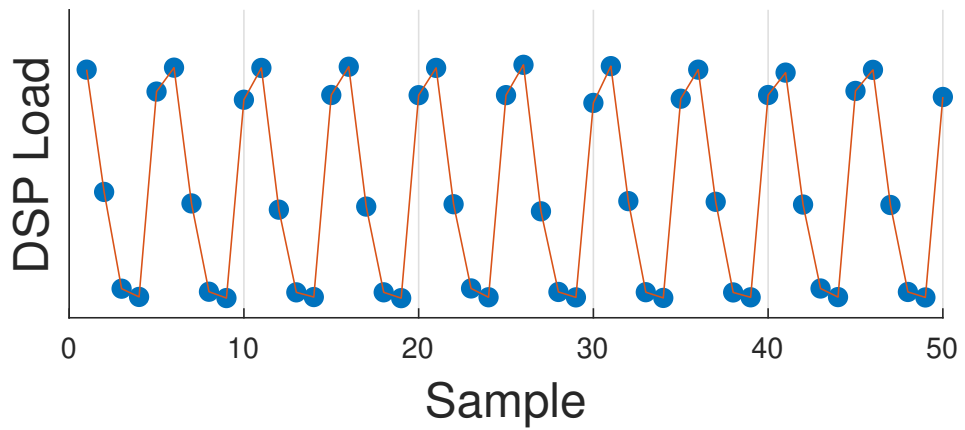
Figure A.11: DSP load in the time domain. Two sectors and a 100 MHz cell.
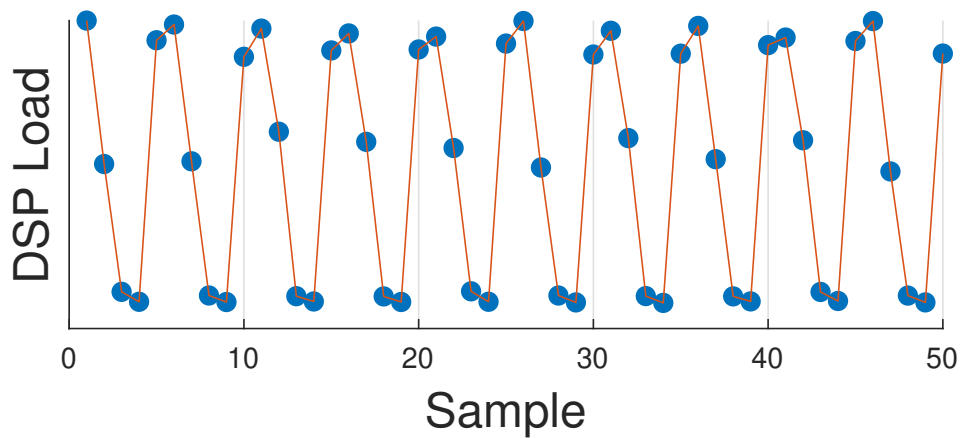
Figure A.12: DSP load in the time domain. Three sectors and a 100 MHz cell.
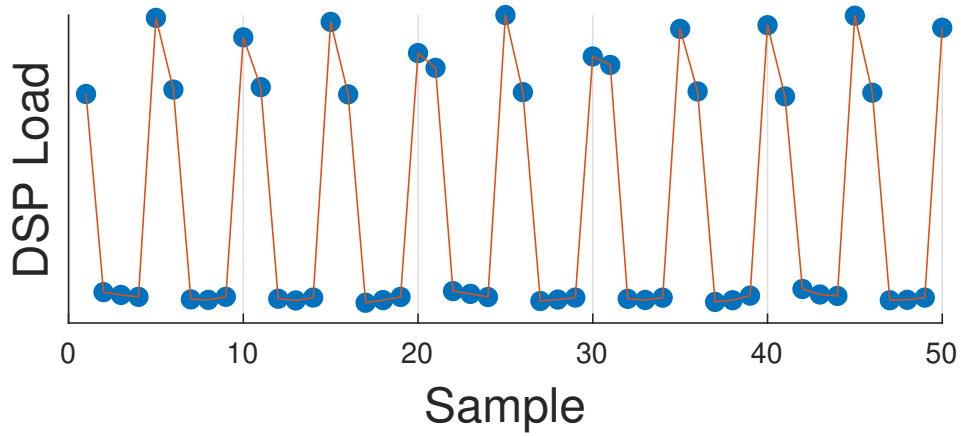
Figure A.13: DSP load in the time domain. One sector and a 50 MHz cell.
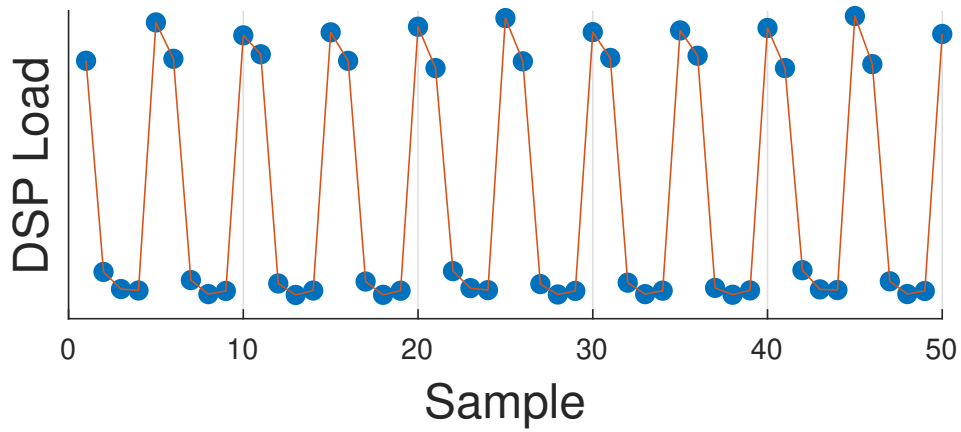
Figure A.14: DSP load in the time domain. Two sectors and a 50 MHz cell.
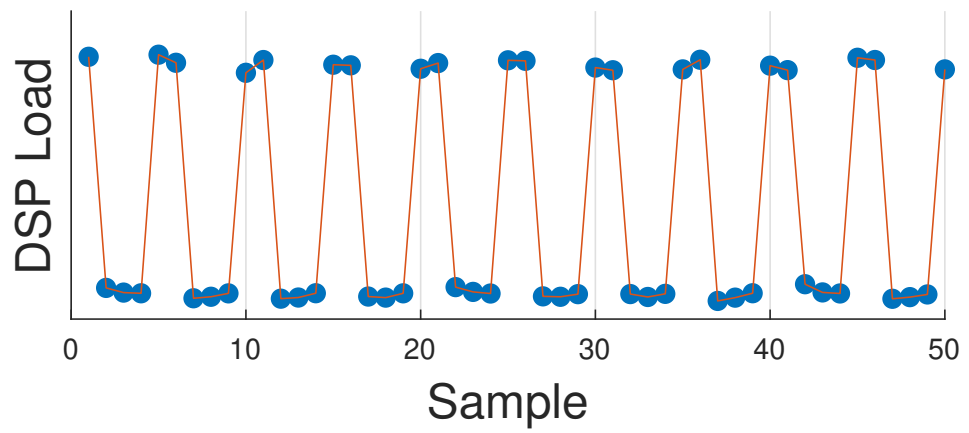
Figure A.15: DSP load in the time domain. Three sectors and a 50 MHz cell.

A.4   TDD pattern divided figures for additional sector and cell BW configurations
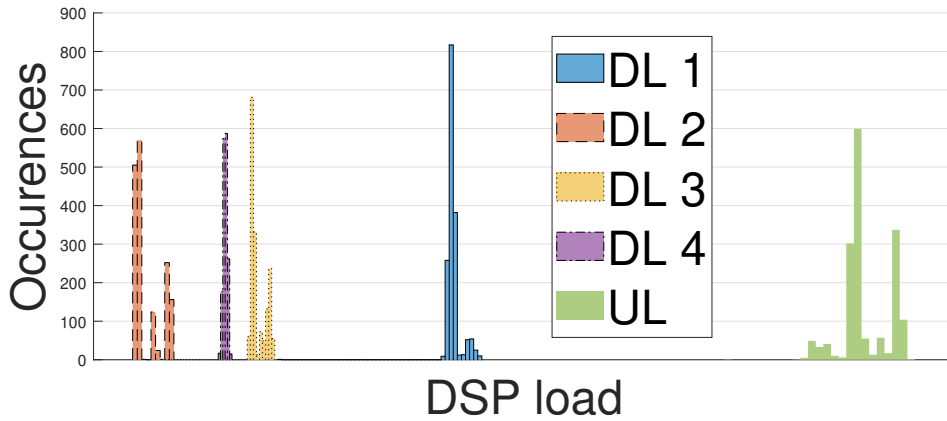
Figure A.16: Histograms for different slot types over the 4:1 TDD pattern of DSP load for low RB, two sectors and a 100 MHz cell.
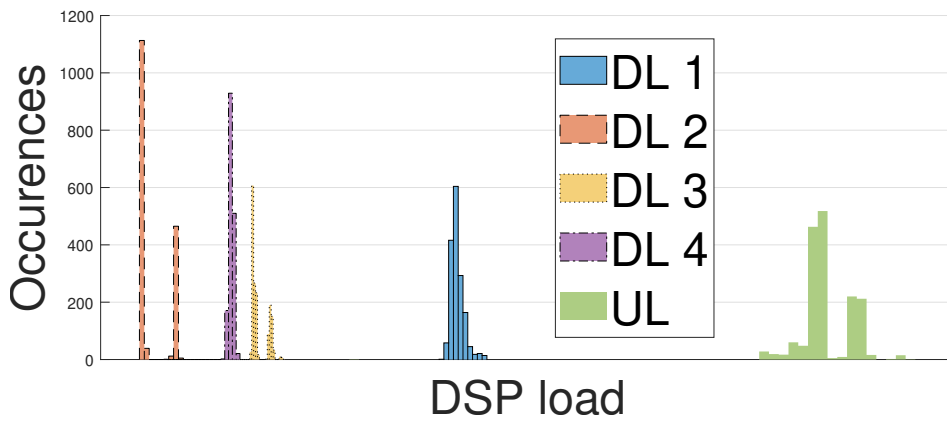
Figure A.17: Histograms for different slot types over the 4:1 TDD pattern of DSP load for low RB, three sectors and a 100 MHz cell.
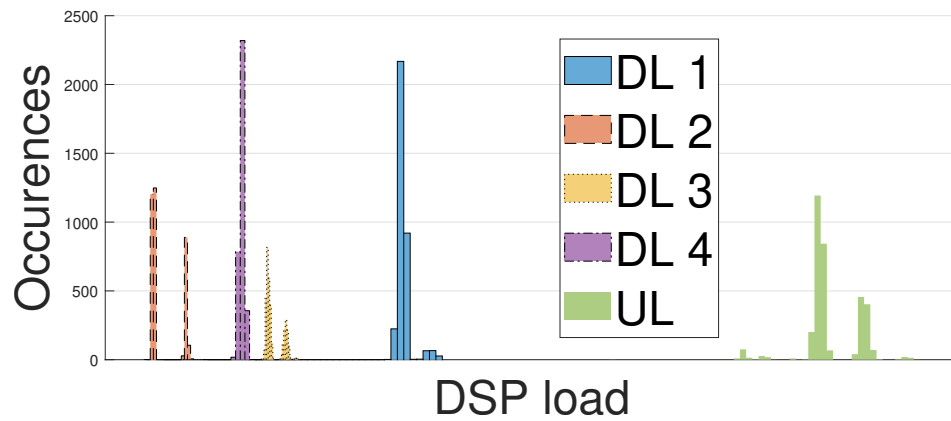
Figure A.18: Histograms for different slot types over the 4:1 TDD pattern of DSP load for low RB, one sector and a 50 MHz cell.
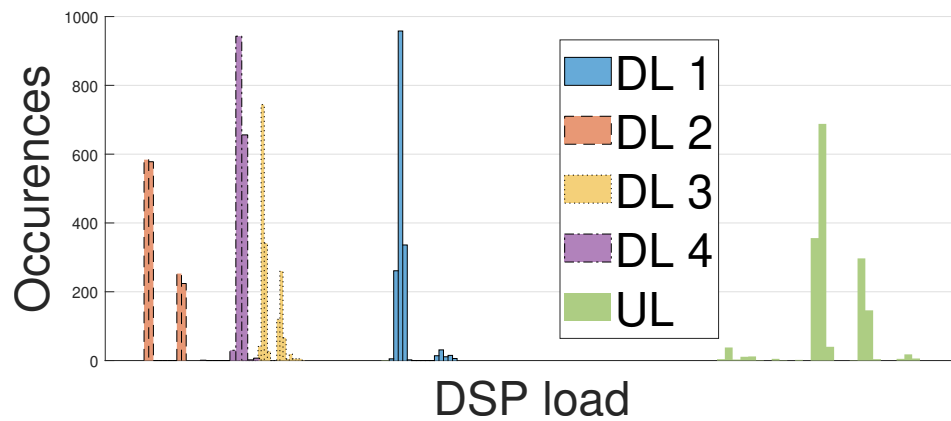
Figure A.19: Histograms for different slot types over the 4:1 TDD pattern of DSP load for low RB, two sectors and a 50 MHz cell.
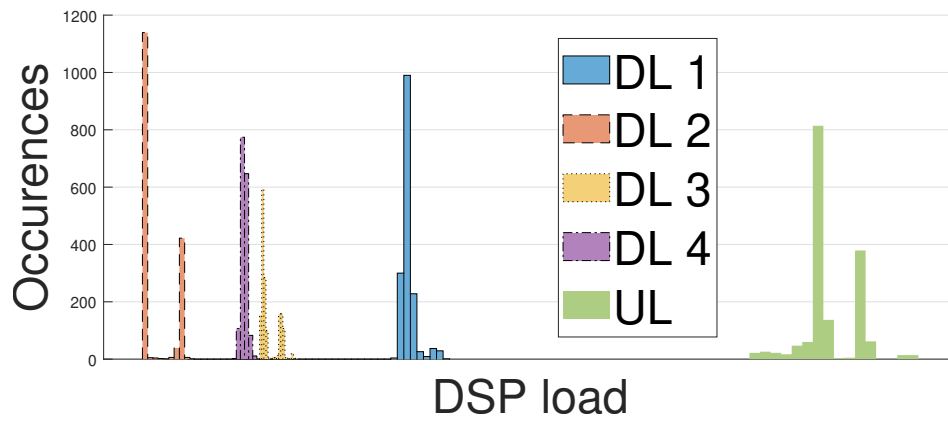
Figure A.20: Histograms for different slot types over the 4:1 TDD pattern of DSP load for low RB, three sectors and a 50 MHz cell.

A.5   Model parameters for additional sector and cell BW configurations
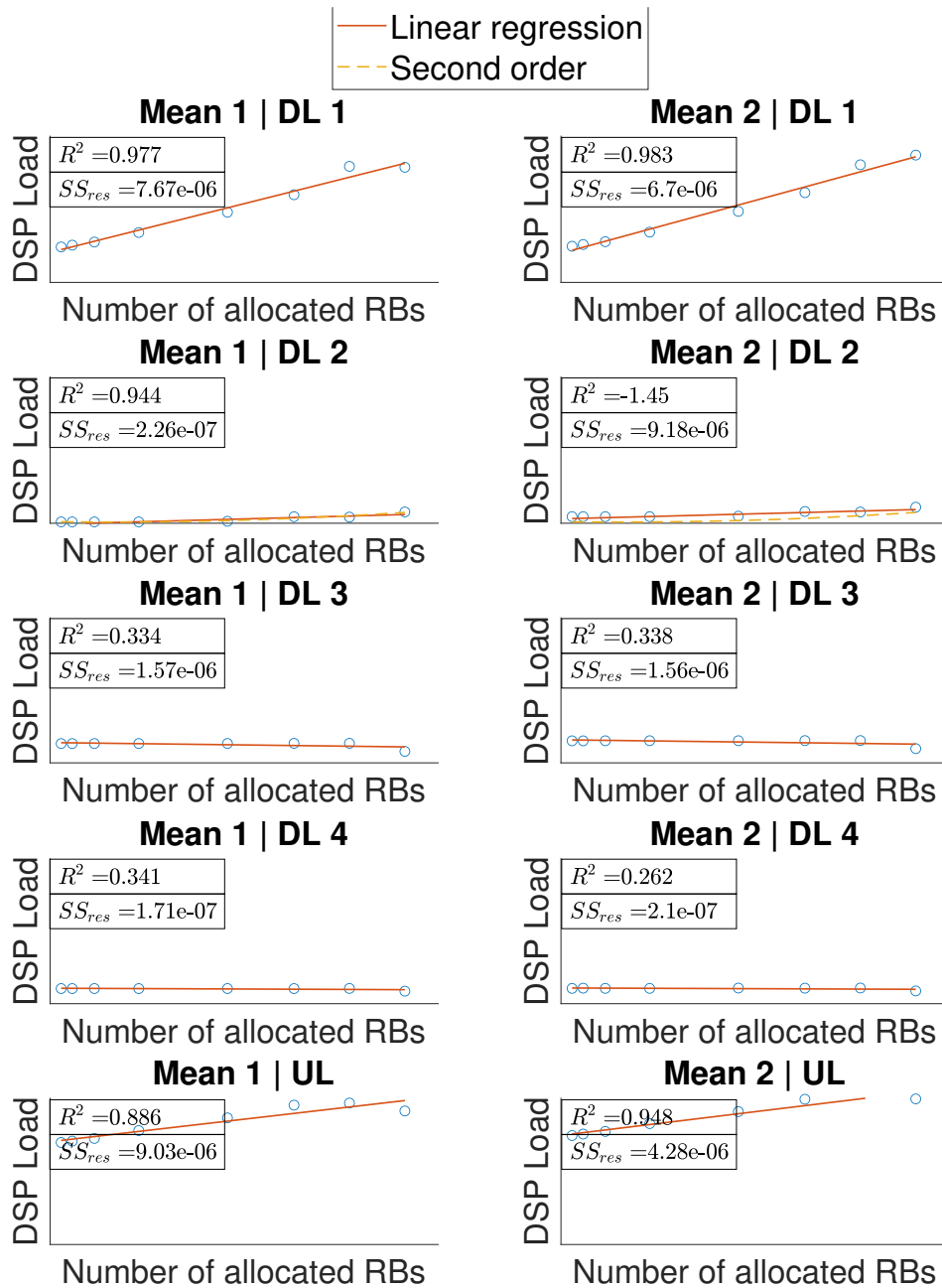
Figure A.21: Interpolated model for mean parameter for the distributions. Configuration: 50 MHz, 1 Sector
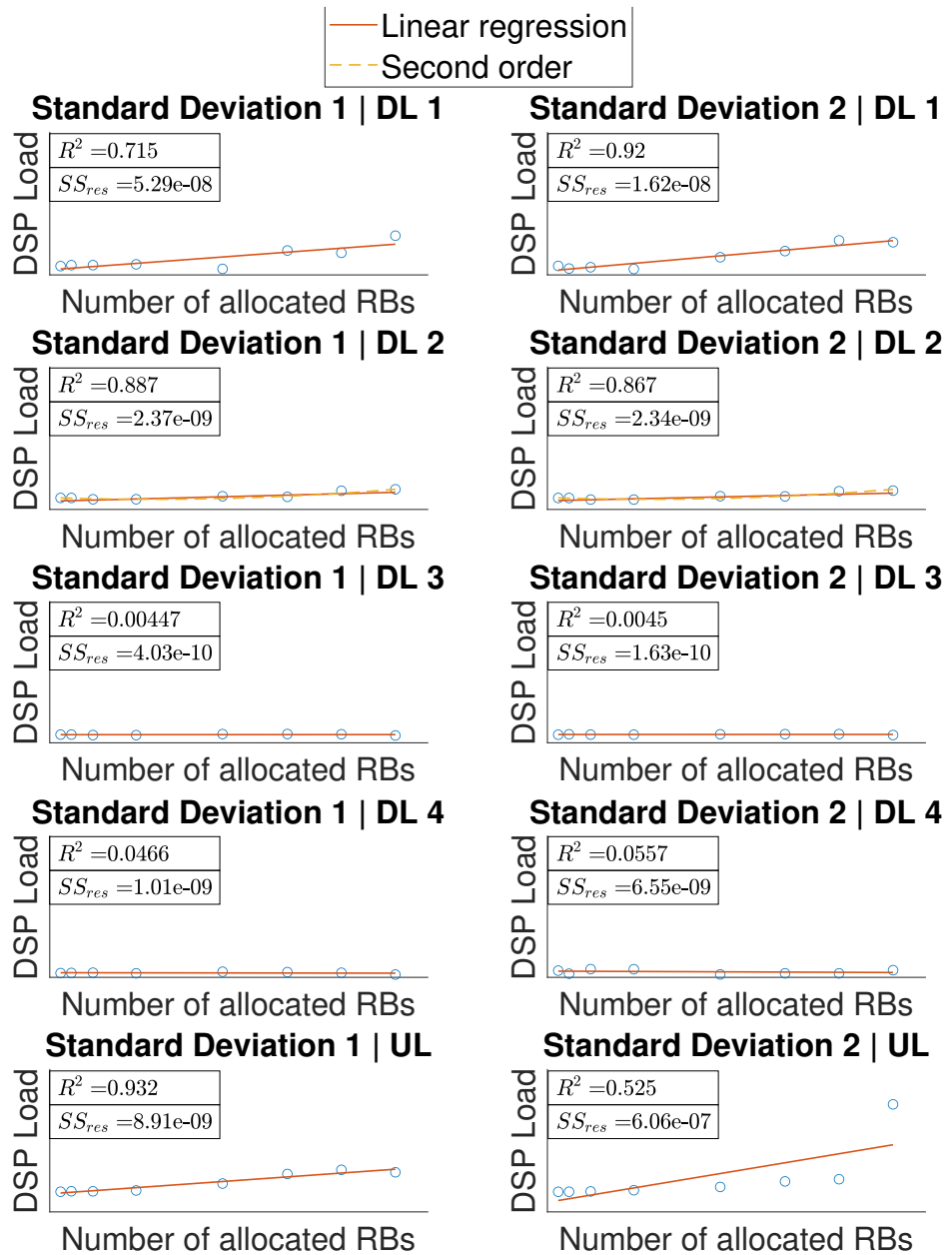
Figure A.22: Interpolated model for standard deviation of the distributions. Configuration: 50 MHz, 1 Sector
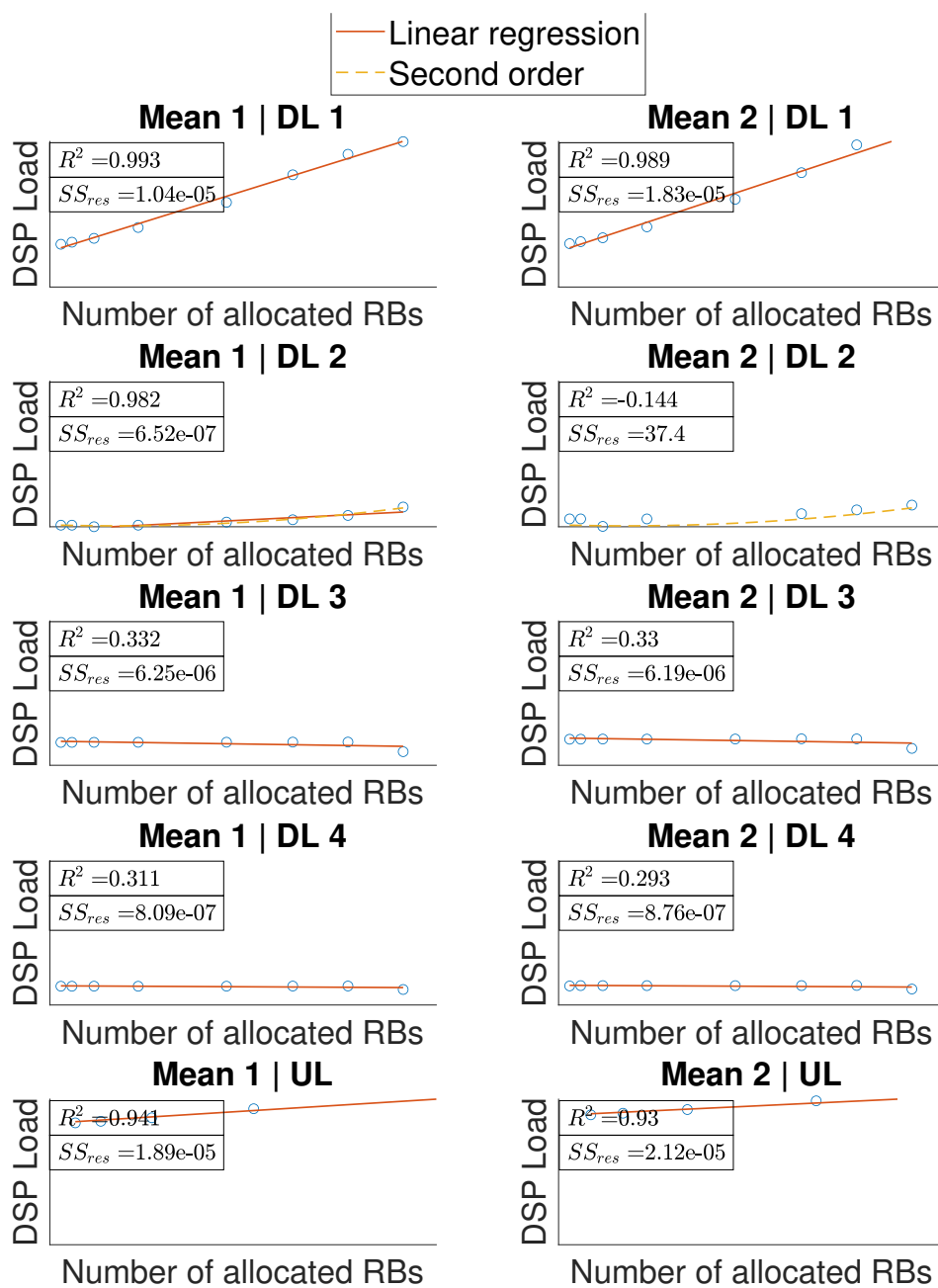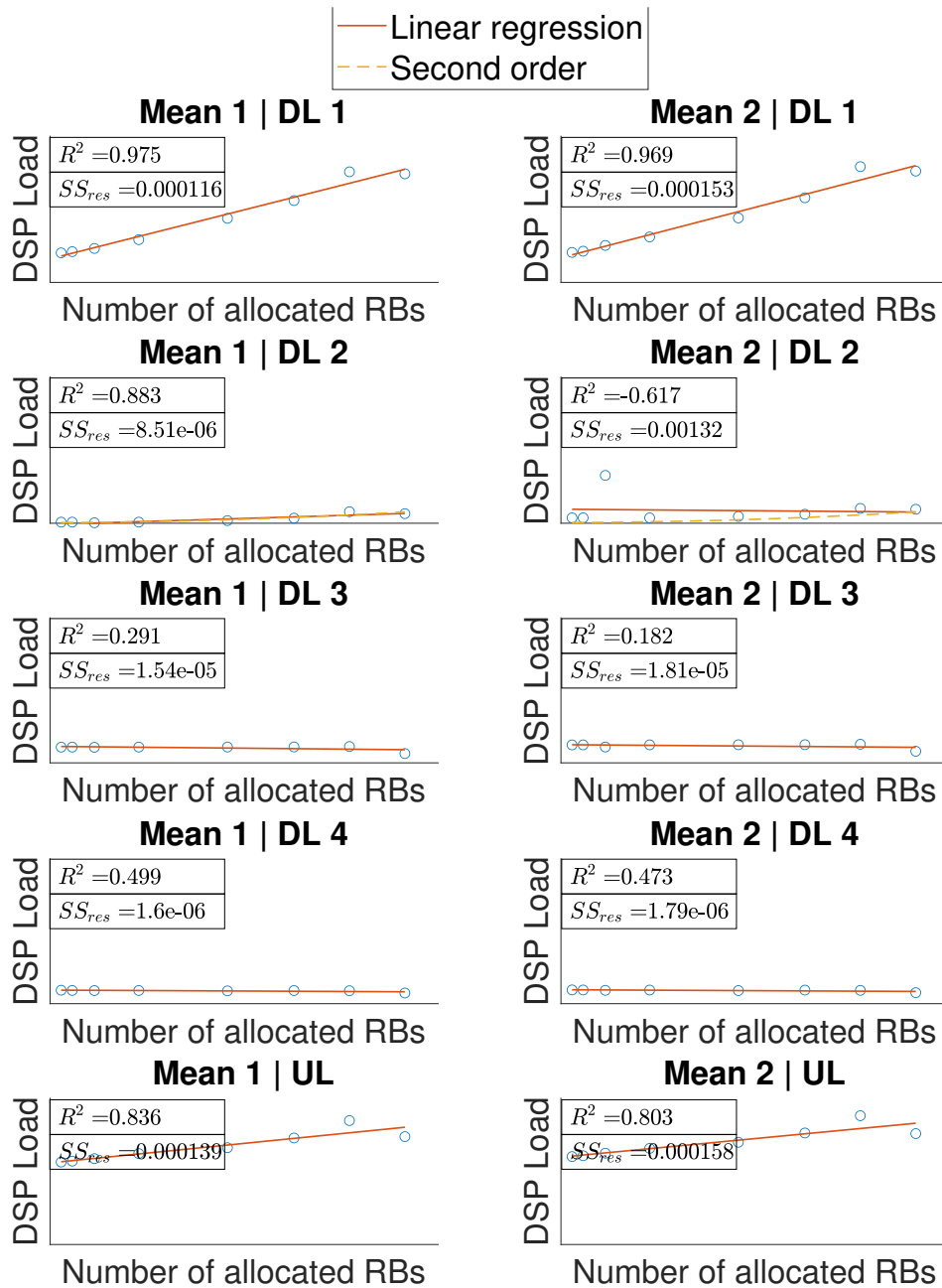
Figure A.23: Interpolated model for mean parameter for the distributions. Configuration: 50 MHz, 2 Sectors
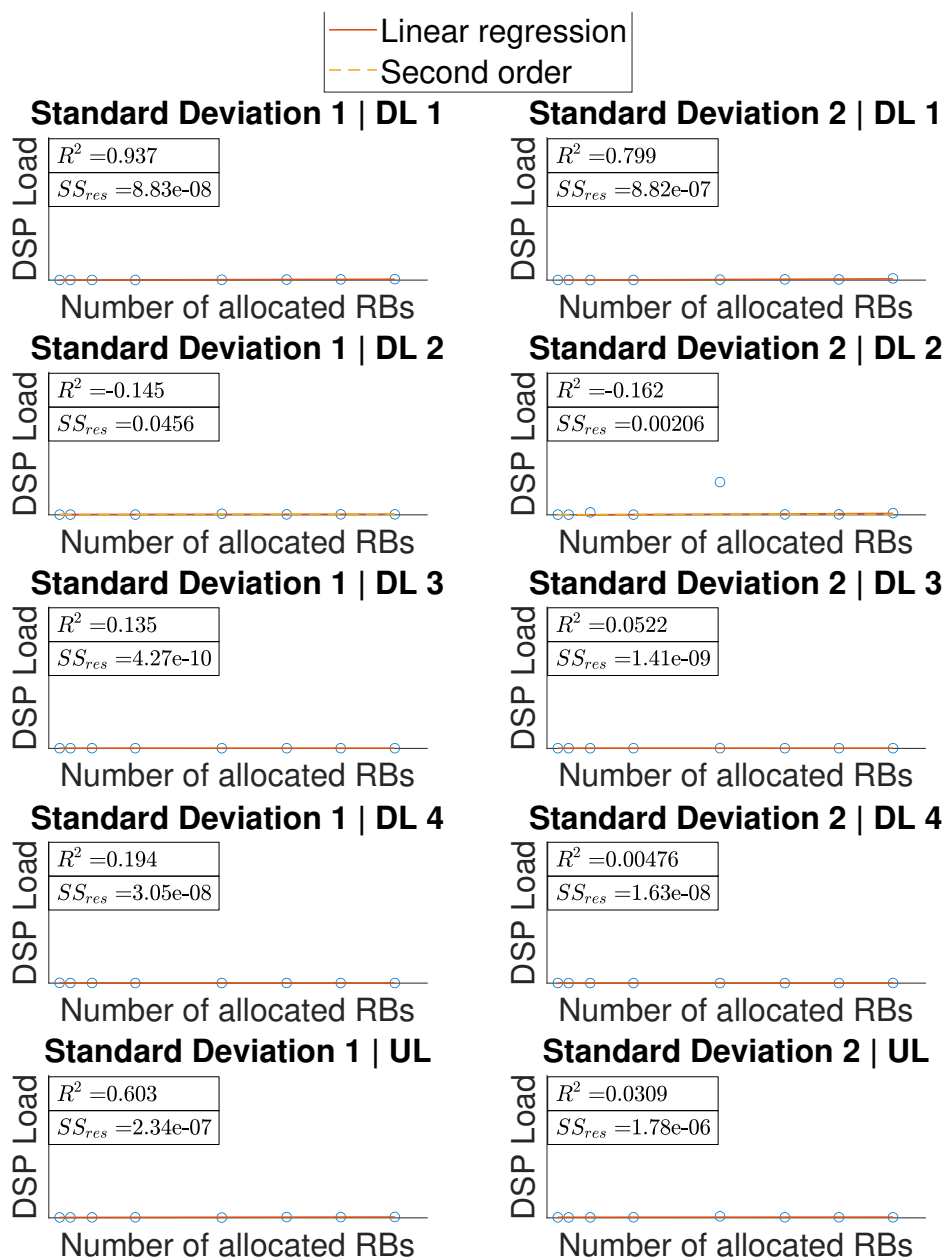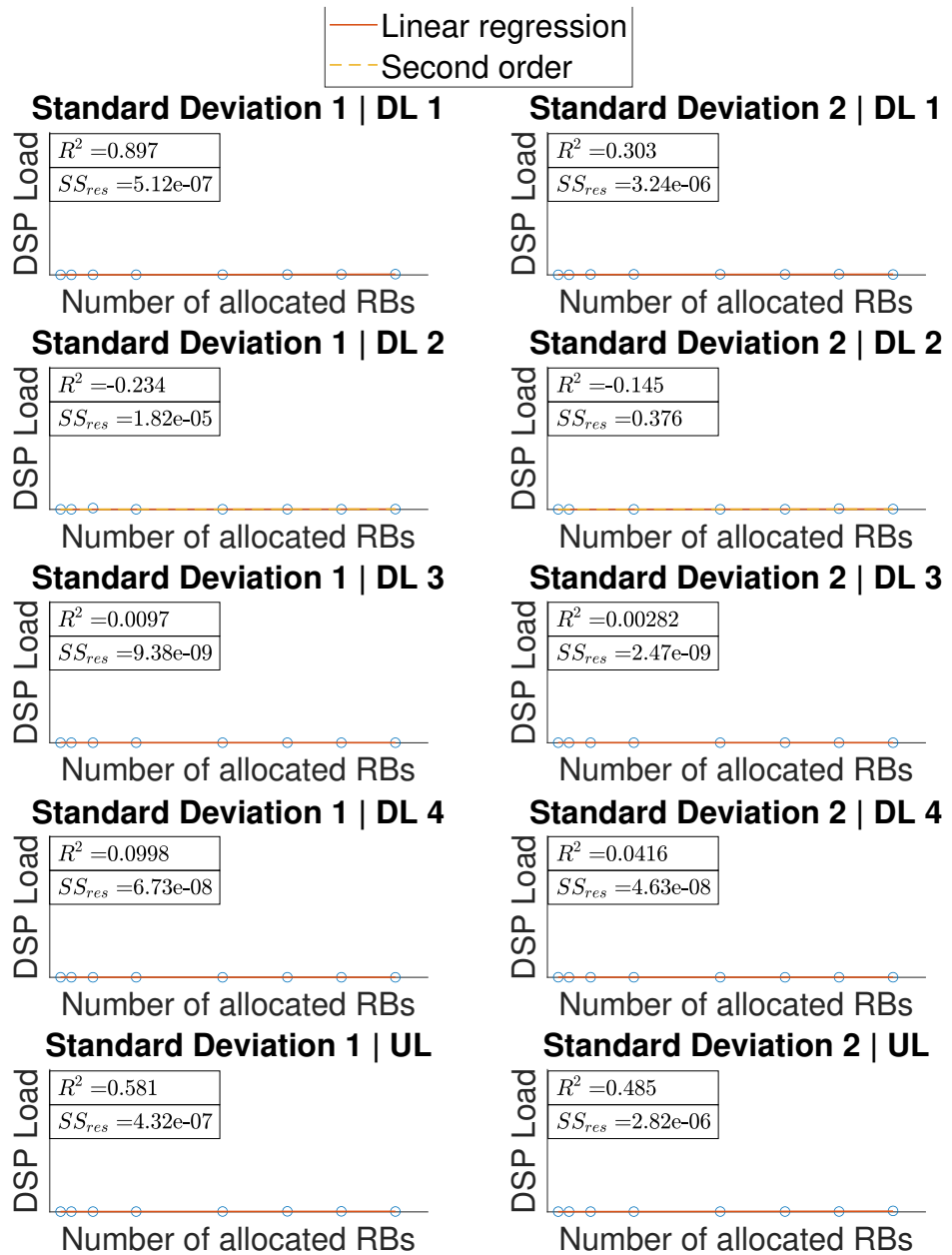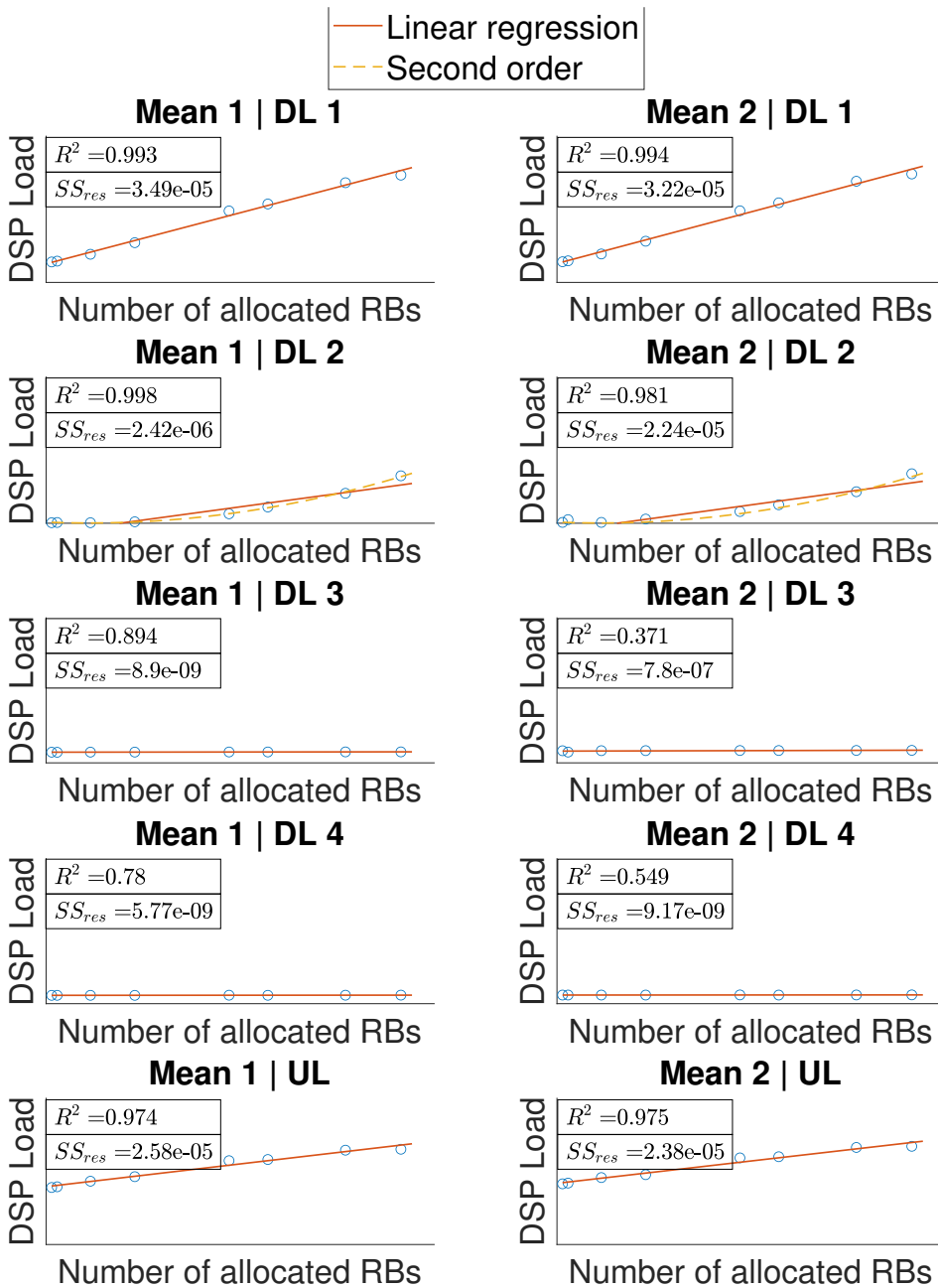
Figure A.24: Interpolated model for standard deviation of the distributions. Configuration: 50 MHz, 2 Sectors

Figure A.25: Interpolated model for mean parameter for the distributions. Configuration: 50 MHz, 3 Sectors
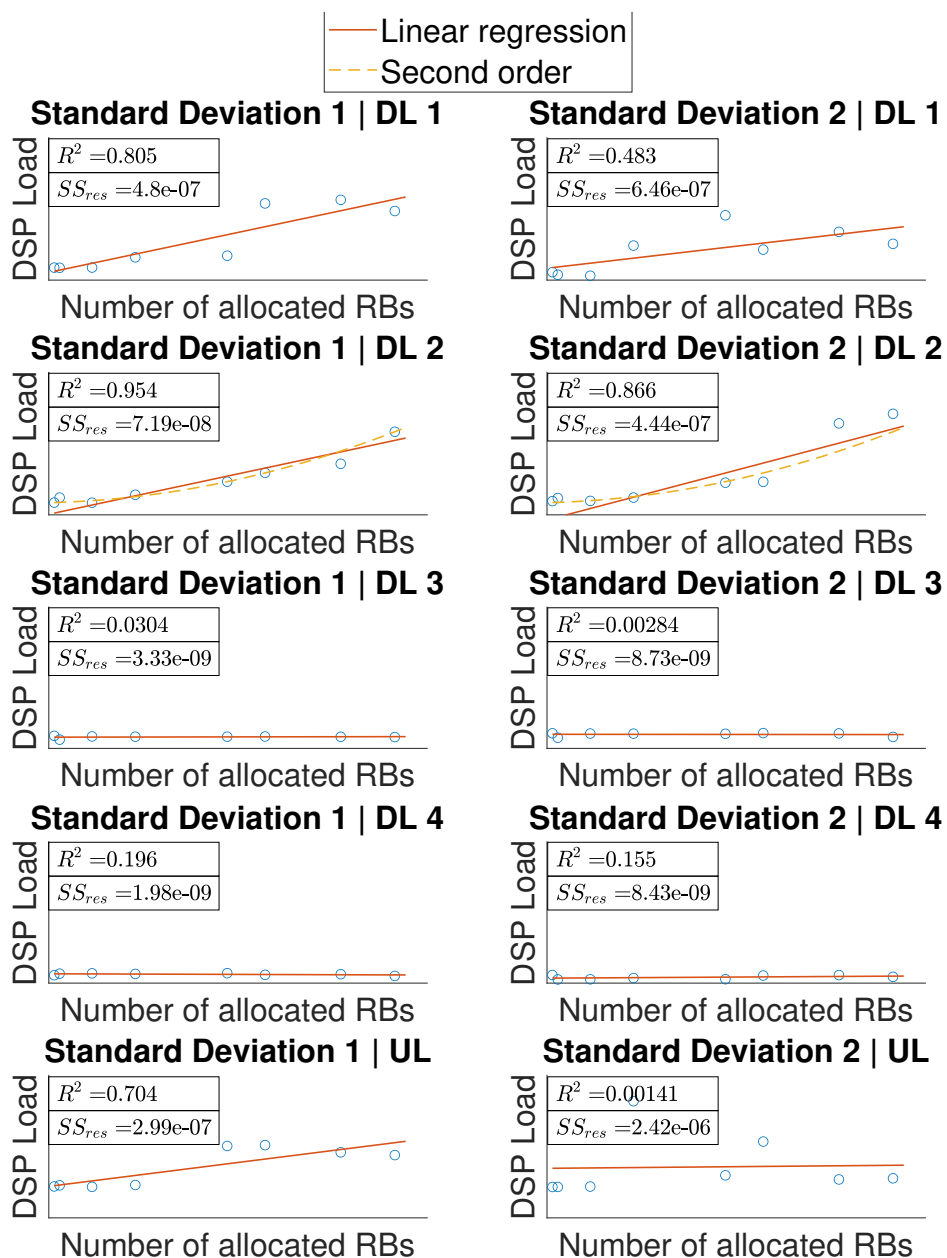
Figure A.26: Interpolated model for standard deviation of the distributions. Configuration: 50 MHz, 3 Sectors
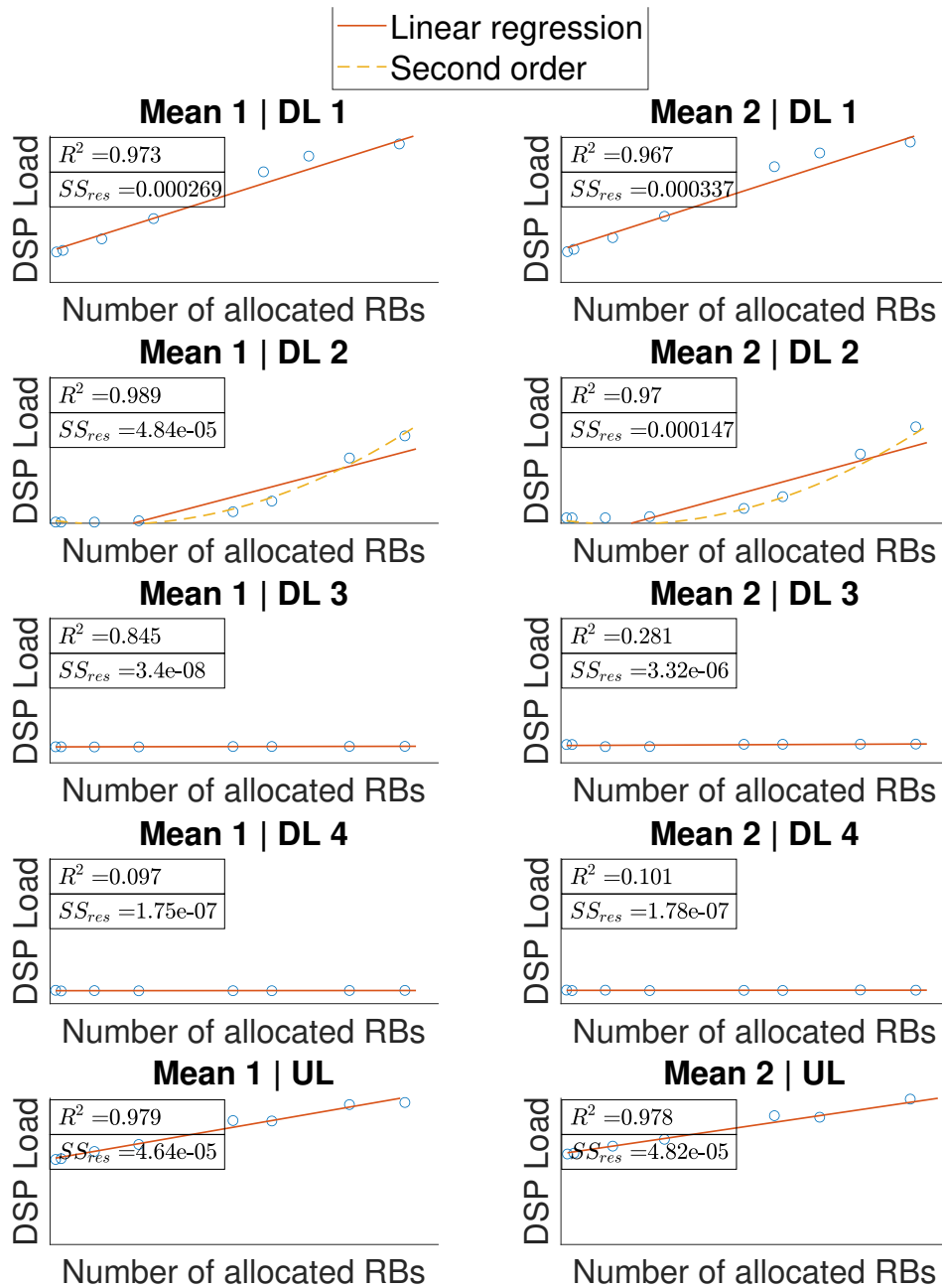
Figure A.27: Interpolated model for mean parameter for the distributions. Configuration: 100 MHz, 2 Sectors

Figure A.28: Interpolated model for standard deviation of the distributions. Configuration: 100 MHz, 2 Sectors

Figure A.29: Interpolated model for mean parameter for the distributions. Configuration: 100 MHz, 2 Sectors
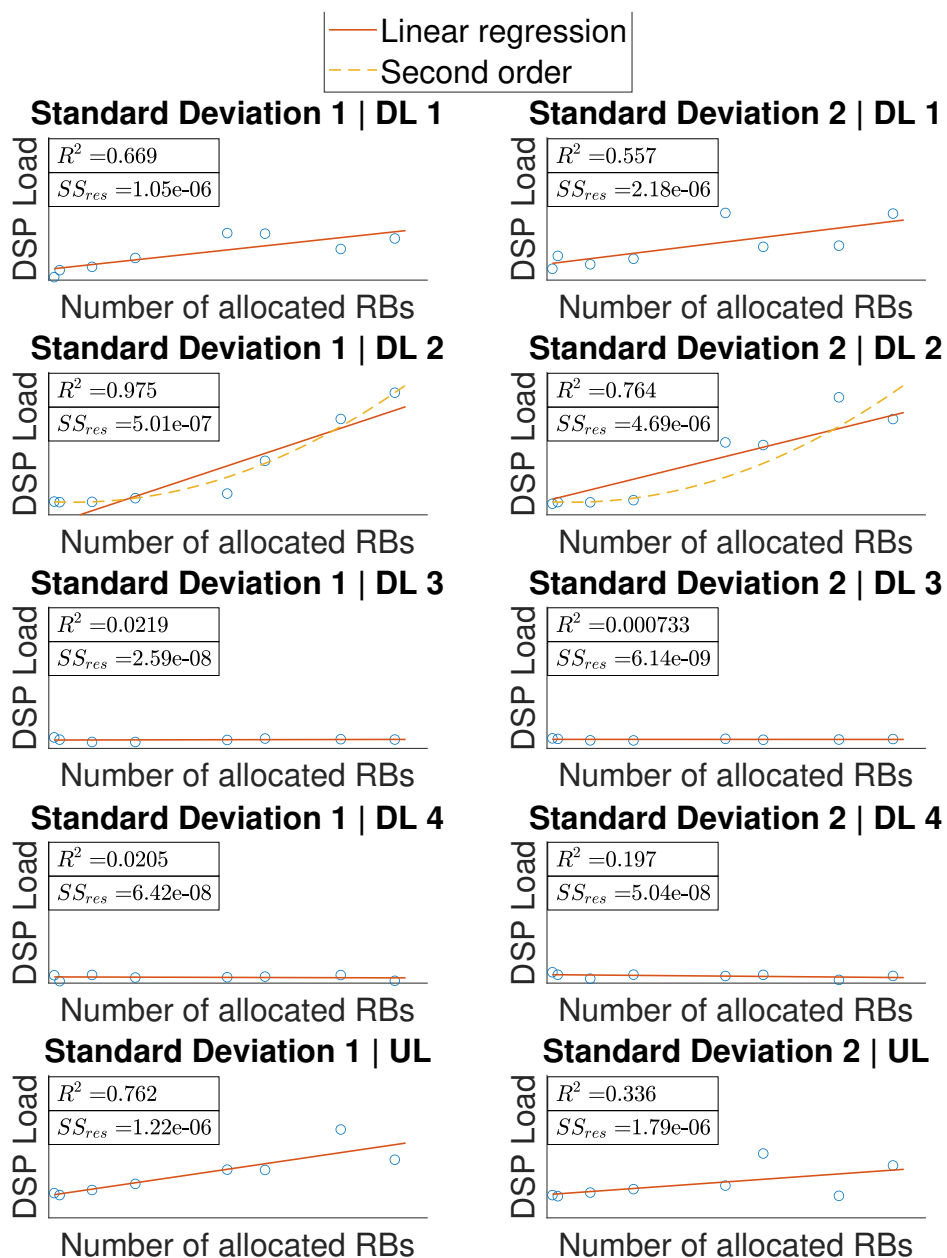
Figure A.30: Interpolated model for standard deviation of the distributions. Configuration: 100 MHz, 3 Sectors

LUND
UNIVERSITY