



Hardware Emulations

using 5G Toolkit and SDRs: Hands-on

GIGAYASA

For academia only



Contributors:

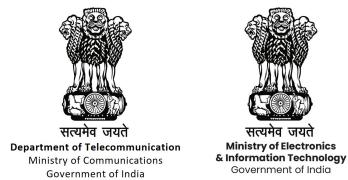
Editor: Simulator Design Team, GIGAYASA

Contributing Authors: Aman Mishra, Vijaya Mareedu and Vikram Singh, GIGAYASA

Grants and Funding

Gigayasa is supported by:

- Indian Institute of Technology, Madras Incubation Center (IITM-IC).
- Startup India.
- Department of Telecommunication (DoT), India.
- Center of Excellence in Wireless Technology (CEWiT), IITM.
- Ministry of Electronics and Information Technology (MEITY), India.



© 2023-2024 Gigayasa Wireless Private Limited. All rights reserved.

Gigayasa/GigaYasa (name and logo), 5G Toolkit/5G-Toolkit/Toolkit-5G/6G Toolkit/6G-Toolkit/Toolkit-6G, and related trade dress used in this publication are the trademark or registered trademarks of GigaYasa and its affiliates in India and other countries and may not be used without express written consent of GigaYasa Wireless. All other trademarks used in this publication are property of their respective owners and are not associated with any of GigaYasa Wireless's products or services. Rather than put a trademark or registered trademark symbol after every occurrence of a trademark name, names are used in an editorial fashion only, and to the benefit of the trademark owner, with no intention of infringement of the trademark. The content in this document is for education and research purposes and not to be shared to persons or organized without license or consent of Gigayasa.

Contents

1	Introduction to ADALM-Pluto SDR and its application programming interface	1
1.1	What are Software Defined Radios (SDRs)?	1
1.2	Why do we use SDRs?	3
1.3	Specifications of some famous SDRs	3
1.4	Introducing some important Pluto SDR APIs	3
1.5	Useful Resources	7
2	Analysis of Wireless Channel Spectrum using ADALM-Pluto SDR	9
2.1	What is a Wireless Spectrum Analysis?	9
2.2	How to Analyze the Wireless Spectrum?	9
2.3	Spectrum Analysis Techniques	9
2.4	Fast Fourier Transform (FFT) based Spectrum Analysis	11
2.5	Results	11
2.6	Useful Links	11
3	Coarse Downlink Time Synchronization in 5G-Wireless Networks	14
3.1	Why Time Synchronization?	14
3.2	How Time Synchronization is Performed?	14
3.3	Coarse Downlink Synchronization in 5G	15
3.4	Results	17
3.5	Useful Resources	18
4	Implementation of OFDM in 5G networks	19
4.1	Introduction to Multiplexing Schemes	19
4.2	Classification of OFDM	22
4.3	Why Orthogonal Frequency Division Multiplexing in 5G Networks	24
4.4	Design of OFDM in 5G	24
4.5	OFDM Implementation in 5G Toolkit	27
4.6	Results	27
4.7	Useful Resources	29
5	Fine Time Synchronization in 5G Networks using PSS and SSS	30
5.1	Why is Fractional/Fine Time Synchronization important?	30
5.2	Fine Time Synchronization Techniques	30
5.3	Fine Time Synchronization in 5G	31
5.4	Results	31
5.5	Useful links	31
6	Carrier Frequency Offset Estimation and Correction in 5G Networks	32
6.1	What is Carrier Frequency Offset?	32
6.2	CFO Estimation Techniques	32
6.3	Results	36
6.4	Useful links	38
6.5	Exercise	38
7	Channel Estimation and Equalization for SSB/PBCH using PBCH-DMRS	40
7.1	Why Estimate Wireless Channel?	40
7.2	Channel Estimation Techniques	40
7.3	Channel Interpolation Techniques	41
7.4	What is Symbol Equalization?	42
7.5	Results	43
7.6	Appendix	45
7.7	Further Reading	48

8 Implementation of PBCH Chain in 5G Networks	49
8.1 What is PBCH Chain?	49
8.2 Design of PBCH Chain	49
8.3 Results	51
9 Uplink Synchronization using physical Random Channel (PRACH) in 5G Networks	53
9.1 Why Uplink Sychronization?	53
9.2 What is PRACH?	53
9.3 Design of PRACH Chain	53
9.4 Results	55
10 PDCCH Implementation and Blind Decoding in 5G Networks	56
10.1 What is the purpose of PDCCH?	56
10.2 PDCCH Tx Chain	56
10.3 References	61
11 Data Communication using PDSCH in 5G Networks	62
11.1 What is PDSCH?	62
11.2 Design of the PDSCH Chain: Transmitter	63
11.3 PDSCH Resource Allocation and Resource Mapping	68
11.4 Design of the PDSCH Chain: Receiver	70
11.5 Results	71
11.6 Appendix	74
12 Performance Analysis of 2x2 Downlink MIMO in 5G Networks	77
12.1 What is MIMO?	77
12.2 MIMO Aspects in 5G	78
12.3 Results	80
12.4 Useful Links	81
13 Important System Parameters and their effect on the 5G Networks Performance	83
13.1 What is Flexible numerology?	83
13.2 What is Frame structure?	83
13.3 Impact of parameters on network performance	84
13.4 Results	86
13.5 Exercise	89
14 References	91

1 | Introduction to ADALM-Pluto SDR and its application programming interface

Welcome to the hand-on laboratory course on **5G wireless systems**. In this course, we will introduce the audience to some practical aspects related to 5G systems using the 5G toolkit and software defined radios (SDRs). We will be using ADALM-Pluto SDR primarily due to the availability, cost effectiveness and ease of use. However, the course can experienced using any other SDR available with you as long as it supports the Python APIs. These APIs will be used for passing the I/Q samples of 5G signals and information generated using 5G -Toolkit to radio unit of the SDRs for analog processing followed by transmission of processed signal over the wireless channel using antennas.

1.1 | What are Software Defined Radios (SDRs)?

Software Defined radios (SDRs) are radio communication systems which allows controlling one or more aspects of the radio unit's functionality using software [3]. The initial communication systems were hard coded systems hence building SDRs seemed lucrative and aspirations. However, with the advent of the technology

1.1.1 | Architecture of the SDRs

The following diagrams shows the a simplified architecture of the SDR which consists primarily for 3 parts.

- Baseband Unit
- Radio Unit
- Antennas

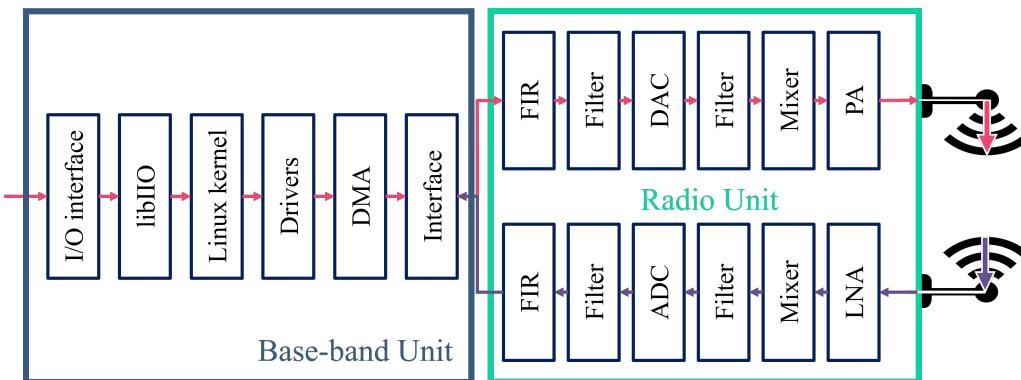


Figure 1.1: Architecture of a simple SDR (ADALM Pluto)

1.1.2 | Components of the SDRs

Software-Defined Radio (SDR) systems consist of various components and modules that work together to enable the flexibility and programmability characteristic of SDR. Here are some important components typically found inside an SDR:

- **RF Front-End:**
 - **Antenna:** The antenna receives radio frequency (RF) signals from the air.
 - **Low Noise Amplifier (LNA):** Amplifies weak incoming signals from the antenna.
- **Analog-to-Digital Converter (ADC):**
 - Converts the analog RF signal into digital form for processing by the digital components.
- **Digital Signal Processor (DSP):**

- Converts the analog RF signal into digital form for processing by the digital components.

- **Field-Programmable Gate Array (FPGA):**

- A programmable logic device that provides hardware acceleration for certain processing tasks, enhancing the overall performance of the SDR.

- **Processor and General-Purpose Computer:**

- The main computing unit responsible for executing higher-level software applications and managing the overall SDR system.

- **Software Interface:**

- Allows users to interact with the SDR, configure settings, and control its operation. This can include graphical user interfaces (GUIs) or command-line interfaces.

- **Operating System:**

- Provides a platform for running software applications and managing hardware resources.

- **Digital-to-Analog Converter (DAC):**

- Converts digital signals back to analog form before transmission through the RF front-end.

Digital-to-Analog Converter (DAC):

- **Power Supply:**

- Provides the necessary power to different components within the SDR.

- **Communication Interfaces:**

- Interfaces for connecting the SDR to external devices or networks. This can include USB, Ethernet, or wireless interfaces.

- **Clocking System:**

- Provides accurate timing for the various components to ensure proper synchronization.

- **Memory and Buffers:**

- Stores digital samples, configuration settings, and other data. This can include both volatile (RAM) and non-volatile (storage) memory.

- **Software Stack:**

- The software stack includes the SDR's firmware and higher-level software applications that define the SDR's functionality. It enables the reprogramming and customization of the radio's behavior.

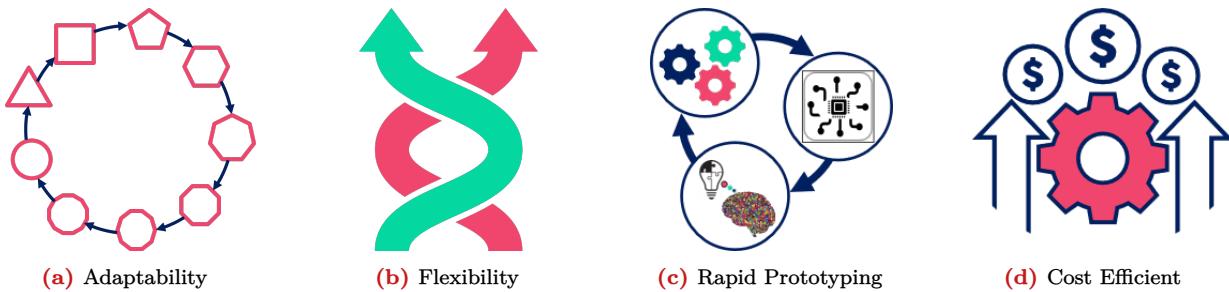
- **Digital Upconverter (DUC) and Digital Downconverter (DDC):**

- The DUC converts baseband digital signals to the intermediate frequency (IF) or RF for transmission, while the DDC converts incoming RF signals to baseband for further processing.

- **Filtering and Amplification Stages:**

- Digital and analog filters, as well as amplifiers, may be included in the signal path to enhance or modify the signal.

These components work together to create a flexible and programmable radio system, allowing users to implement a wide range of communication protocols and adapt to various operating conditions. The combination of hardware and software elements enables the versatility that distinguishes SDR from traditional hardware-based radio systems.

**Figure 1.2:** Utility of SDRs

1.2 | Why do we use SDRs?

The primary advantage of SDR lies in its flexibility and adaptability, as it allows for the reconfiguration and reprogramming of the radio's functionality through software updates. Here are some utilities of SDR and how they differ from traditional hardware-based radios:

1.2.1 | Adaptability

SDRs can operate on multiple frequency bands and support various communication standards concurrently. This is advantageous for applications that require interoperability across different frequency bands or standards. Upgrades can be implemented through software updates, allowing for improvements in performance, security, or additional features without requiring hardware modifications.

1.2.2 | Flexibility and Reconfigurability

With SDR, the radio's functionality can be modified through software updates, enabling rapid adaptation to changing communication standards. This flexibility is particularly beneficial for applications like military communication, emergency services, and research.

1.2.3 | Rapid Prototyping and Validation

SDRs are popular in research and development environments because they provide a platform for experimenting with new communication techniques and protocols without the need to build new hardware for each experiment.

1.2.4 | Cost Efficiency

While the initial cost of developing SDR may be high, the ability to adapt and upgrade through software can result in long-term cost savings, especially in applications with evolving communication standards.

1.3 | Specifications of some famous SDRs

The popularity and relevance of SDRs generally change over time. There are many good SDRs available in the market which are being used for variety of the purposes. However, as per our expertise the SDRs listed in table 1.1 are most suitable SDRs for learning, teaching and academic research on 5G. These SDRs supports Python APIs and are within the budget for academic institutes and Wireless SDR amateurs. There are other SDRs available in the market such as RTL-SDR, Xilinx RFSoCs, NI USRPs but these SDRs are either reception only, or too expensive or doesn't support Python APIs.

1.4 | Introducing some important Pluto SDR APIs

The software functionalities of the SDRs are controlled using the APIs which are provided by the SDR vendor. These APIs are segregated into properties (variables) and methods (functions). The properties set the important parameters which are crucial to the operation of the radio units and its components. On the other hand, the methods are crucial for reading from and writing to the SDRs (I/O interfacing). We have segregated the APIs into three categories which are the SDR setup API, transmitter APIs and receiver APIs. Lets discuss these APIs in detail.

Table 1.1: Specifications of the some famous SDRs

Function	Pluto-SDR	LimeSDR mini 2.0	AntSDR E200	USRP B210/200/205mini
Chipset	AD9363	LMS7002M	AD9361/9363	AD9361/9363/9363
Frequency Range	325 MHz - 3.8 GHz	10 MHz - 3.5 GHz	70/325 MHz - 6/3.8 GHz	70 MHz - 6 GHz
Interface	USB 2.0	USB 3.0	Gb Ethernet	USB 3.0
Embedded	Yes	No	Yes	No
RF Bandwidth	20 MHz	40 MHz	56/20 MHz	61.44 MHz
Sample Depth			12 bits	
Sample Rate	61.44 MSPS	30.72 MSPS	61.44 MSPS	61.44 MSPS
Tx Channels	2	1	2	2/1/1
Rx Channels	2	1	2	2/1/1
Programmable Logic Gates	28k	44k	85k	100k/75k/150k
Memory			512 MB	
Duplex			Full	
Open Source		Full		Schematic & firmware
Oscillator Precision	± 20 ppm	± 1 ppm initial, ± 4 ppm stable		± 2 ppm
Transmit Power*	≤ 6 dBm		Up to 10 dBm	≥ 10 dBm
Cost†	\$250	\$400	\$600/\$350	\$2300/\$1300/\$1600
Note-1: Lime Microsystems is currently shipping LimeSDR mini 2.0 only.				
Note-2: AntSDR E200 comes in two versions which differs in the RF-chipsets:				
<ul style="list-style-type: none"> ■ AntSDR E200 with AD9361 ■ AntSDR E200 with AD9363 				
* The maximum transmit power of a RF chipset depends on the carrier frequency.				
† The prices may change over time and doesn't include the import/excise duty.				

1.4.1 | APIs for SDR Setup

Before transmitting or receiving the data using SDRs, we have to configure some basic setup parameters crucial to SDR operation. These APIs connects the specific SDR over USB, set the sample rate for SDR operation etc.

Table 1.2: Some important application interfaces (APIs) used in Analog Devices interface (ADI)

Type	API	Function
method	Pluto	Create object of SDR setup object.
property	sample_rate	Sets the sample rate for the ADC/DAC of the SDR.

adi.Pluto(uri=ip)

This method creates an SDR object which can access the device with *uri* address of *ip*. Following code shows a few ways to do this.

```

1 sdr = adi.Pluto(uri="ip:192.168.2.1")
2 # or
3 sdr = adi.Pluto("ip:192.168.2.1")
4 # or
5 sdr = adi.ad9361(uri="ip:192.168.2.1")
6 # or
7 sdr = adi.ad9361("ip:192.168.2.1")

```

sdr.sample_rate

This property is used to set the sample rate (in SPS) used by SDR for analog to digital conversion (ADC) and digital to analog conversion (DAC) at the receiver and transmitter path respectively.

```

1 sdr = adi.Pluto(uri="ip:192.168.2.1") # Setup object connected to SDR with given IP.
2 Nfft          = 1024                  # FFT size
3 subcarrierSpacing = 30*(10**3)        # subcarrier spacing
4 sdr.sample_rate = int(sample_rate)    # set the sample rate

```

1.4.2 | Transmitter APIs

The APIs allows the users to control carrier frequency, signal amplification, bandwidth of the transmitted signal, transmit signal, configure the repeated transmission of the buffered I/Q samples and clear the transmit buffer.

Table 1.3: Some important APIs related to transmitter used in ADI

Type	API	Function
method	tx	Transmits the samples input to the method.
	tx_destroy_buffer()	Clear the buffer and stop the transmission of data.
property	tx_rf_bandwidth	Set the bandwidth of the transmit filter.
	tx_lo	Sets the transmitter local oscillator frequency.
	tx_hardwaregain_chan0	Sets the gain of the transmitter power amplifier.
	tx_cyclic_buffer	Enable/disable cyclic transmission of samples in buffer.
	tx_enabled_channels	Integer indicating the indices of the tx-channels.

■ **sdr.tx(samples)**

This method is used to *transmit* the *samples* input to it. It is important to note that the input samples is expected to stay within the interval -2^{14} to $2^{14} - 1$. Otherwise, the input samples will be a victim of saturation while digital to analog conversion. Its always good to scale the input to these values to relax the need of amplification using power amplifier which is a non linear system. The scale and range of inputs is different for different SDRs.

■ **sdr.tx_destroy_buffer()**

This property is a boolean flag to clear/destroy the transmission buffer and stop the transmission of samples from the tx buffer. Default value is False.

■ **sdr.tx_rf_bandwidth**

This is a property to set the bandwidth (in Hz) of the transmission filter. The variable should comply with the specifications of the SDR. For Pluto SDR, the transmission bandwidth should be between 200 kHz and 56 MHz.

■ **sdr.tx_lo**

This property sets the carrier frequency (in Hz) used by local oscillator to generate the carrier/sinusoid waveform. The carrier is used to modulate/up-convert the analog information signal. For Pluto SDR, this variable can take a value between 325 MHz to 3.8 GHz.

■ **sdr.tx_hardwaregain_chan0**

This property sets the amplification gain (dB) provided by the power amplifier to the modulated signal. The higher the value stronger the output signal and longer a signal can propagate over the wireless medium. For Pluto SDR, this variable can take values between -90 to 0.

■ **sdr.tx_cyclic_buffer**

This property is a boolean flag to enable/disable the cyclic transmission of samples stored in the transmit buffer. This property is very useful for non realtime simulations/emulations. Default value is False.

■ `sdr.tx_enabled_channels`

This is Python list type property used to configure the number of tx channels and their channel indices. The number of entries or length of the property defines the number of tx channel. For Pluto SDR, this property can either have length 1 or 2 (if set up for dual channel transmission).

```

1 # Import the library
2 import adi
3
4 # Set carrier frequency
5 carrierFrequency      = 10**9
6
7 # Generate 1000 samples
8 x_time                = np.random.rand(1000)
9
10 # Config Tx
11 sdr.tx_rf_bandwidth    = int(60*10**6) # filter cutoff, just set it to the same as
12             sample rate
12 sdr.tx_lo                  = int(carrierFrequency)
13 sdr.tx_hardwaregain_chan0 = -10 # Increase to increase tx power, valid range is -90 to 0
14             dB
14
15 # Initiate transmission
16 sdr.tx_cyclic_buffer     = True # Enable cyclic buffers
17 sdr.tx(x_time)           # start transmitting

```

1.4.3 | Receiver APIs

The receiver APIs provides the users with methods and properties to sample the wireless channel and store the received samples, control the reception filter bandwidth, strength the received signal and, set the local oscillator frequency. The APIs are discussed briefly in table 1.4 and in detail in this sections.

Table 1.4: Some important APIs related to receiver used in ADI

Type	API	Function
method	<code>rx</code>	Sample the wireless channel and load them into buffer.
property	<code>rx_rf_bandwidth</code>	Set the bandwidth of the receive filter.
	<code>rx_lo</code>	Sets the receiver local oscillator frequency.
	<code>rx_hardwaregain_chan0</code>	Sets the gain of the receive low noise amplifier.
	<code>gain_control_mode_chan0</code>	Defines the mode of receiver AGC.
	<code>rx_buffer_size</code>	Number of samples to read and load into SDR buffer.
	<code>rx_enabled_channels</code>	Integer indicating the indices of the rx-channels.

■ `sdr.rx()`

This method samples the wireless channel using the RF chipset, stores the samples in the receive buffer, and return the content of the receive buffer. The number of samples stored in the buffer is configured using `sdr.rx_buffer_size`.

■ `sdr.rx_rf_bandwidth`

This is a property to set the bandwidth (in Hz) of the reception filter. The variable should comply with the specifications of the SDR. For Pluto SDR, the transmission bandwidth should be between 200 kHz and 56 MHz.

■ `sdr.rx_lo`

This property sets the carrier frequency (in Hz) used by local oscillator to generate the carrier/sinusoid waveform. The carrier is used to demodulate/down-convert the analog information signal. For Pluto SDR, this variable can take a value between 325 MHz to 3.8 GHz.

■ `sdr.rx_hardwaregain_chan0`

This property sets the amplification gain (dB) provided by the low noise amplifier to the received signal. The higher the value stronger the output signal. The variable is relevant if and only if the `sdr.gain_control_mode_chan0` is set to ***manual***. For Pluto SDR, this variable can take values between 0 to 74.5 dB. When setting the gain manually, one must ensure that output of low noise amplifier doesn't saturate the ADC.

■ `sdr.gain_control_mode_chan0`

This property is sets the gain control mode for the low noise amplifier (LNA). Pluto SDR supports 3 modes of gain control

- `manual`
- `slow_attack`
- `fast_attack`

For the manual gain control, the gain is set using the variable `sdr.rx_hardwaregain_chan0`. The gain can be configured automatically using either "`slow_attack`" or "`fast_attack`" mode. The AGC ensure that the ADC doesn't saturate and loose waveform information. The "`fast_attack`" mode estimates the optimal gain aggressive initially and may results in sharp transients before stabilizing.

■ `sdr.rx_buffer_size`

This property set the receiver buffer size. When the `sdr.rx` is invoked, the SDR starts to sniff the wireless channel and stores the received samples in the receiver buffer. The upper limit on the size of this buffer is defined by the DRAM size.

■ `sdr.rx_enabled_channels`

This is Python list type property used to configure the number of rx channels and their channel indices. The number of entries or length of the property defines the number of rx channel. For Pluto SDR, this property can either have length 1 or 2 (if set up for dual channel reception).

```

1 # Import the library
2 import adi
3
4 # Config Rx
5 sdr.gain_control_mode_chan0 = 'manual'
6 sdr.rx_hardwaregain_chan0 = 40.0      # dB
7 # The receive gain on the Pluto has a range from 0 to 74.5 dB.
8
9 # or
10
11 sdr.gain_control_mode_chan0 = 'slow_attack'
12 # # AGC modes:
13 #     # 1. "manual"
14 #     # 2. "slow_attack"
15 #     # 3. "fast_attack"
16
17 sdr.rx_lo          = int(carrierFrequency)
18 sdr.rx_rf_bandwidth = int(60*10**6) # filter width, just set it to the same as sample
19 # rate for now
20 sdr.rx_buffer_size = int(4*buffer_size)
21
22 # Clear buffer just to be safe
23 for i in range (0, 10):
24     raw_data = sdr.rx()
25
26 # Receive samples
27 rx_samples = sdr.rx()
28
29 # # Stop transmitting
30 sdr.tx_destroy_buffer()
```

1.5 | Useful Resources

We request all the readers to please explore the following resources as well. These blogs, YouTube channels, and web-pages will be useful for firmware/libraries installation, and utilities of the SDRs beyond wireless communication.

- ADI installation.
 - Installation for Windows
 - Installation for Linux
- Jon Kraft Youtube Channel.
 - Upgrading Pluto to 2×2 MIMO SDR.
 - Simplest way to Upgrade Pluto Firmware.
- Mark Litchman's PySDR.
 - Setup and Getting Started with Pluto SDR
 - Setup and Getting Started with NI USRP B2x0
- Pluto-ADALM SDR official resources.
- Gigayasa's resources: SDR integration with 5G-Toolkit.

2 | Analysis of Wireless Channel Spectrum using ADALM-Pluto SDR

Wireless spectrum refers to the range of electromagnetic radio frequencies used for wireless communication. It encompasses all the frequencies used for transmitting data wirelessly, including radio waves, microwaves, and infrared radiation. The wireless spectrum is a critical resource for various communication technologies, including radio and television broadcasting, mobile communication, Wi-Fi, satellite communication, and many other wireless applications.

2.1 | What is a Wireless Spectrum Analysis?

Wireless channel spectrum analysis involves the examination and study of the frequency spectrum characteristics of a wireless communication channel. The goal is to understand the frequency distribution, channel conditions, and potential sources of interference in a given frequency band. The wireless channel spectrum is analyzed for following purposes,

- **Frequency Spectrum Overview:** It involves identifying the frequency bands available for wireless communication and understanding the allocation of frequency ranges for various services.
- **Channel Characteristics:** Analyzing the characteristics of individual communication channels within the spectrum. This includes studying factors such as bandwidth, signal strength, noise levels, and other parameters that affect the quality of communication.
- **Interference Detection:** Identifying and analyzing any interference present in the spectrum. Interference can come from other wireless devices, electronic equipment, or environmental factors. Detecting interference is crucial for optimizing the performance of wireless networks.
- **Signal Propagation:** Understanding how signals propagate in the frequency domain, including factors like signal attenuation, fading, and multipath effects. This knowledge helps in designing communication systems that can cope with different propagation conditions.
- **Frequency Selectivity:** Analyzing how different frequencies are affected differently by the channel. Some frequencies may experience higher attenuation or interference, and spectrum analysis helps in making informed decisions about channel selection.
- **Occupancy Analysis:** Examining how frequently different frequency bands are occupied or used by existing wireless services. This information is essential for selecting clear frequency bands to avoid interference.
- **Dynamic Spectrum Access:** In some cases, wireless channel spectrum analysis is used for dynamic spectrum access, allowing systems to adapt their operating frequency based on real-time analysis of the spectrum.

2.2 | How to Analyze the Wireless Spectrum?

This analysis is essential for designing and optimizing wireless communication systems, ensuring efficient use of available frequencies and minimizing interference. The wireless spectrum is analyzed and visualized using following tools tools,

- Spectrum analyzers,
- Software-defined radios.

In this chapter, we will use software defined radio for real time and quasi-real time wireless spectrum analysis.

2.3 | Spectrum Analysis Techniques

Spectrum analysis techniques involve various methods for examining and understanding the frequency components of a signal within the electromagnetic spectrum. These techniques are crucial for tasks such as identifying signal characteristics, detecting interference, and optimizing the use of available frequency bands. Here are some common spectrum analysis techniques:

1. Spectral Density Analysis:

- **Description:** Analyzing the power distribution of a signal across different frequency components.

- **Application:** Understanding how signal power is distributed across the spectrum.

2. Fast Fourier Transform (FFT):

- **Description:** Transforming a time-domain signal into its frequency-domain representation.

- **Application:** Identifying specific frequencies within a signal and understanding their amplitudes.

3. Swept-Tuned Spectrum Analysis:

- **Description:** Sweeping a narrowband filter across the frequency range to measure the signal power at different frequencies.

- **Application:** Identifying the frequency components and detecting narrowband signals.

4. Real-Time Spectrum Analysis:

- **Description:** Continuously monitoring and analyzing the spectrum in real-time.

- **Application:** Detecting and responding to dynamic changes in the spectrum, such as identifying intermittent interference.

5. Vector Signal Analysis (VSA):

- **Description:** Analyzing both amplitude and phase information of a signal in the frequency domain.

- **Application:** Comprehensive analysis of modulation and encoding characteristics in communication signals.

6. Pulse-Analysis Techniques:

- **Description:** Focusing on analyzing signals with pulses or time-varying characteristics.

- **Application:** Characterizing radar signals, pulsed communication systems, or other signals with specific time-domain features.

7. Coherence Analysis:

- **Description:** Examining the degree of correlation between two signals in the frequency domain.

- **Application:** Understanding the relationship between different signals and identifying coherent components.

8. Waterfall Display:

- **Description:** Displaying the spectral content of a signal over time, providing a visual representation of signal changes.

- **Application:** Monitoring how the frequency content of a signal evolves over time.

9. Power Spectral Density (PSD) Analysis:

- **Description:** Estimating the power distribution of a signal per unit frequency.

- **Application:** Quantifying the signal power at different frequencies, helping to identify frequency bands with high or low power.

10. Intermodulation Analysis:

- **Description:** Detecting and analyzing unwanted signals created by the interaction of multiple signals.

- **Application:** Identifying and mitigating intermodulation interference in communication systems.

In this Chapter, we will implement some of these techniques on the SDR to analyze and visualize the wireless channel spectrum.

2.4 | Fast Fourier Transform (FFT) based Spectrum Analysis

Among various techniques, Fourier transform based spectrum analysis is considered the most computationally efficient. The received signal $y(t)$ can be transformed to frequency domain using continuous time Frequency transform (CTFT) as follows,

$$Y(f) = \frac{1}{2\pi} \int_{-\infty}^{\infty} s(t) * \exp^{-j2\pi ft} dt \quad (2.1)$$

The CTFT is non causal in nature and hence can not be implemented practically. However, the exact same processing can be performed if the sample the time domain signal and frequency spectrum satisfying the Nyquist criterion for sampling. The time domain samples can be organized in frames of size N_{FFT} . Sampling the time domain sample $s(t)$ at $t = n * T$ and frequency domain spectrum $Y(f)$ at $f = k * \Delta f$ results in,

$$Y[k * \Delta f] = \sum_{n=0}^{N_{FFT}-1} s[n * T] * \exp^{-j2\pi k * \Delta f * n * T} \quad (2.2)$$

if the $s(t)$ is band-limited to $N_{FFT} * \Delta f$. We can drop the frequency and time scaling factor of Δf and T , respectively, from equation 2.2 to simplify the representation to,

$$Y[k] = \sum_{n=0}^{N_{FFT}-1} s[n] * \exp^{-j2\pi k * n / N_{FFT}} \quad (2.3)$$

where the critical sampling or Nyquist sampling is performed such that $T = \frac{1}{N_{FFT} * \Delta f}$. It can be observed that the frequency domain version of the time domain can be simply computed by the taking the FFT of the time domain samples provided they are sampled at a rate higher than $N_{FFT} * \Delta f$ and bandwidth of the signal is not higher than $N_{FFT} * \Delta f$.

$$Y[k] = \text{FFT}_{N_{FFT}} \left(\left\{ s[n] \right\}_{n=l * N_{FFT}}^{(l+1) * N_{FFT} - 1} \right), \quad k = 0, 1, 2, \dots, N_{FFT} - 1. \quad (2.4)$$

2.5 | Results

We will discussed the performance and results of quasi real-time and real-time implementation of FFT based spectrum analysis.

Observation-1: The high sample-rate improves the time resolution and increases the bandwidth of the spectrum being sensed.

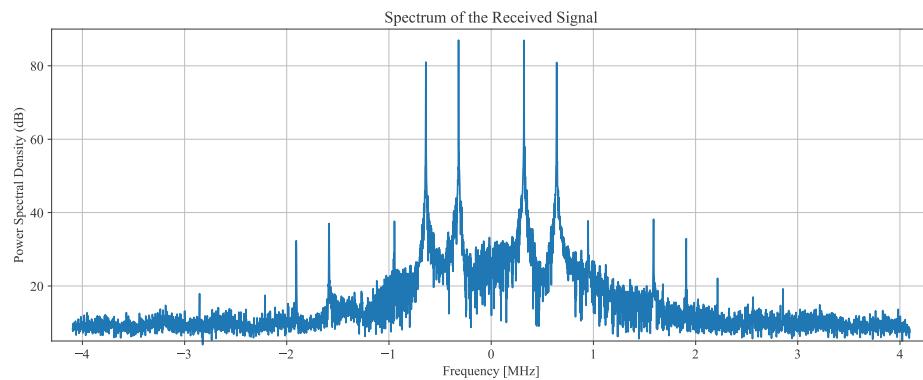
The higher sample rate also requires higher buffer size (DRAM size) to compute the spectrum of the received signal. Furthermore, the higher sample-rate reduces the sample duration and higher FFT size resulting in higher computational complexity. Both these factors results in higher cost and higher power consumption.

Observation-2: Reducing the subcarrier spacing improves the resolution of the signal spectrum but requires higher FFT-size to process the same bandwidth.

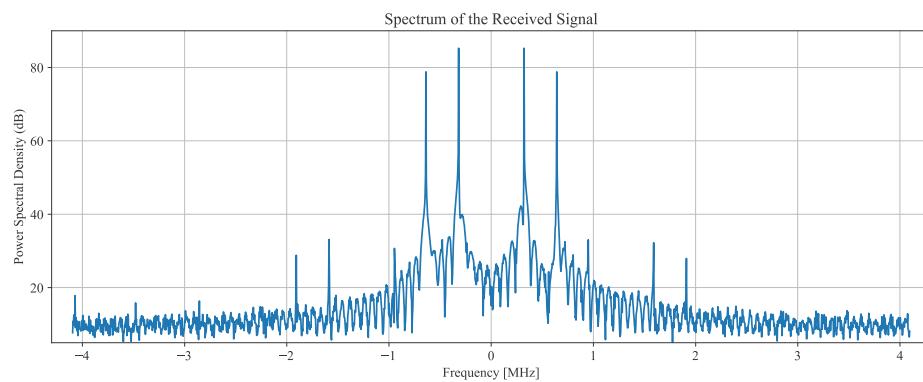
2.6 | Useful Links

The following links provides higher amount of details on computation of signal spectrum.

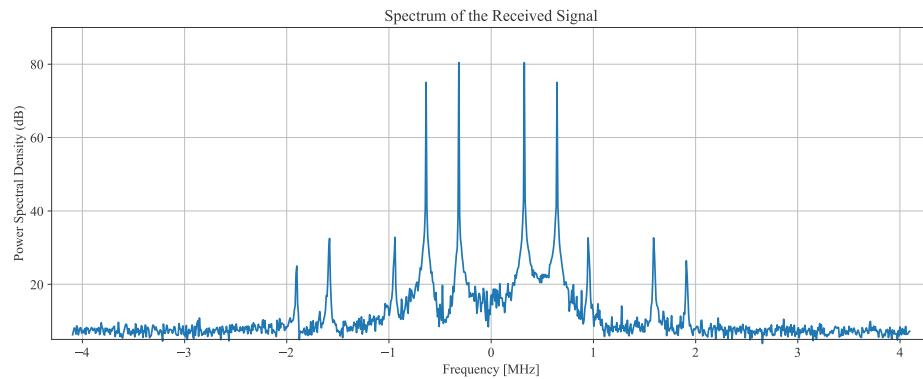
- Quasi real-time spectrum computation.
- Real-time spectrum computation.



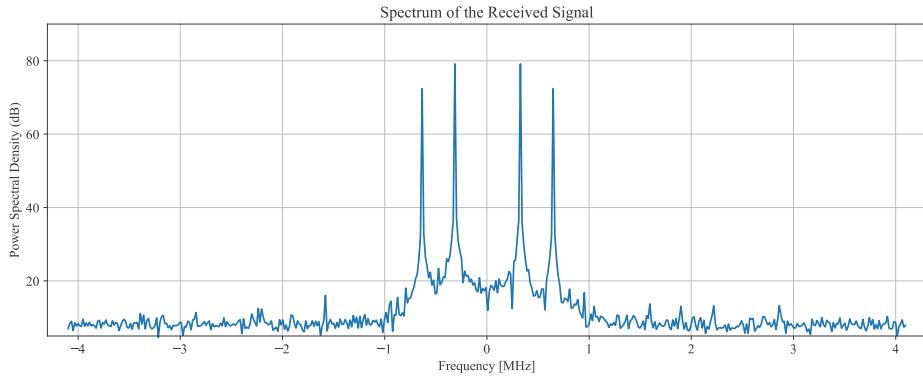
(a) Spectrum Computation with 1kHz subcarrier spacing and 4096 point FFT



(b) Spectrum Computation with 2kHz subcarrier spacing and 2048 point FFT

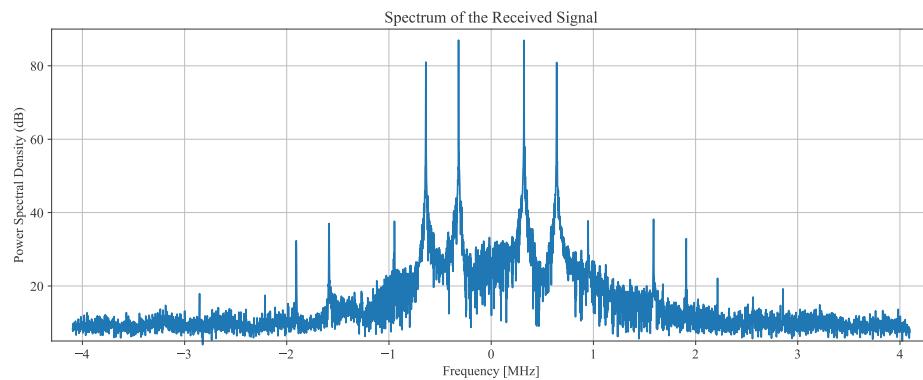


(c) Spectrum Computation with 4kHz subcarrier spacing and 1024 point FFT

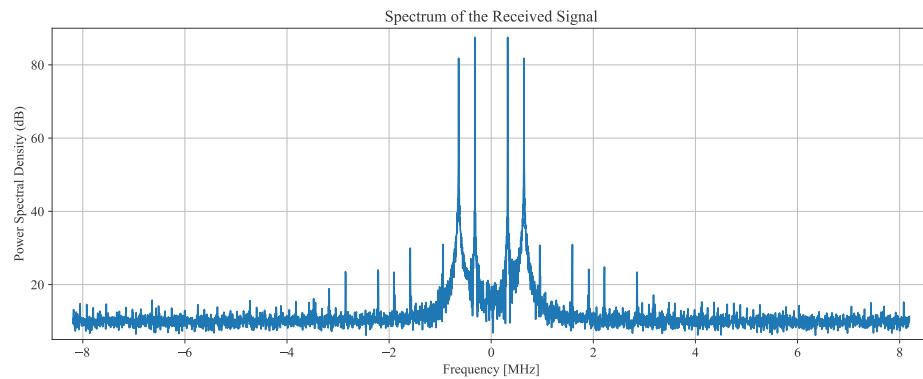


(d) Spectrum Computation with 8kHz subcarrier spacing and 512 point FFT

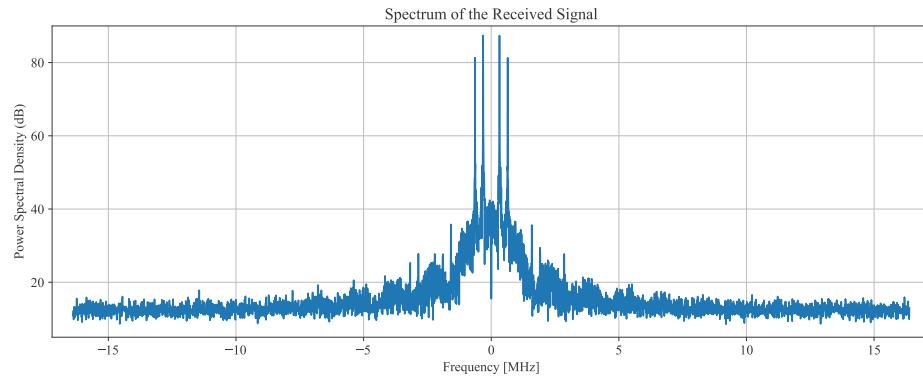
Figure 2.1: Variation in the resolution of spectrum for different subcarrier spacing



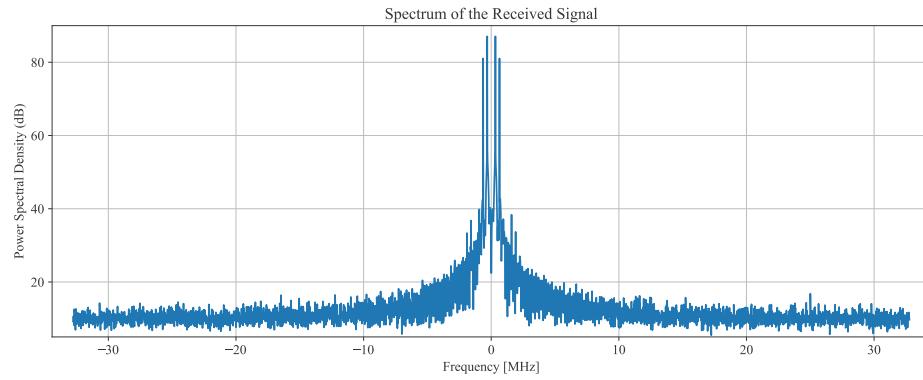
(a) Spectrum Computation with 1kHz subcarrier spacing and 4096 point FFT



(b) Spectrum Computation with 2kHz subcarrier spacing and 4096 point FFT



(c) Spectrum Computation with 4kHz subcarrier spacing and 4096 point FFT



(d) Spectrum Computation with 8kHz subcarrier spacing and 4096 point FFT

Figure 2.2: Variation in the resolution of spectrum for different sample-rate

3 | Coarse Downlink Time Synchronization in 5G-Wireless Networks

Time synchronization is the first and one of the most critical process in establishing the connection between the transmitter and receiver. The process of transmitting the data from the base-station to a user equipment is called **downlink communication**. Hence the process of finding the starting point or boundary of the frame transmitted by the base-station to UEs is called **downlink time synchronization**. In this experiment, we will introduce the audience to downlink time/frame synchronization in 5G networks.

3.1 | Why Time Synchronization?

The objective of communication, in general, is to reliably transmit data over the medium. This data, before transmission, is organized into frames, and each of these frames is communicated sequentially from the transmitter to the receiver. The sizes of these frames may vary depending on latency, data rate, and reliability requirements. At the physical layer, these frames are referred to as Orthogonal Frequency Division Multiplexing (OFDM) frames. The receiver accumulates frame samples zone by one, and once a frame's worth of samples is buffered, the complete frame or a group of frames is decoded together. Initially, when there is no secure link between the transmitter and receiver, the User Equipment (UE) doesn't know which sample marks the beginning of the frame or where the frame actually starts. Furthermore, attempting to decode the data blindly through brute force can overwhelm the system, resulting in high computational complexity. Hence, a robust, low-complexity method is required for frame synchronization in time.

3.2 | How Time Synchronization is Performed?

The time/frame synchronization is performed in 2 steps as stated below:

- Coarse Time Synchronization
- Fine Time Synchronization.

The coarse time synchronization estimates the integer sample offset (n_0) and the fine time synchronization estimates the fractional part of the sample offset (Δn_0) as shown in figure 3.1. The total time offset $n = n_0 + \Delta n_0$ gives the exact estimate of where the OFDM frame starts.

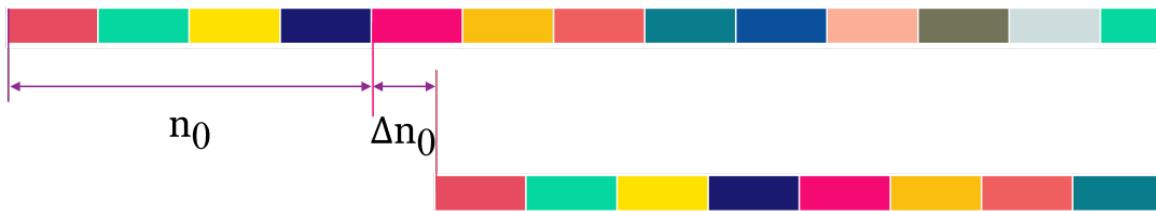


Figure 3.1: The sample offset estimation for time synchronization.

The fine synchronization is beyond the scope of this chapter and will be covered in the upcoming chapter. The coarse time sample offset is estimated in time domain based on running correlation. is typically performed in time domain based on time correlation method.

3.2.1 | Time Correlation based Coarse Time Synchronization

The received samples, denoted $y(n)$, are stored by the receiver in the buffer of size ($\geq N_{\text{FFT}} + L_{\text{CP}}$). The receiver is expected to know a $s(n)$ sequence of pilots which are broadcasted by the transmitter to facilitate the time synchronization. These pilots can be generated either in time domain or frequency domain. The time domain pilots are transmitted directly by RF chipset. On the other hand, frequency domain pilots loaded in OFDM resource grid are transformed in time domain and prefixed with a cyclic prefix before transmission as shown in fig. 3.2.

The receiver performs the running correlations with the possible pilot sequences,

$$r_i(n) = \sum_{l=0}^{N_{\text{FFT}}-1} y(l+n) * s_i^*(l) \quad (3.1)$$

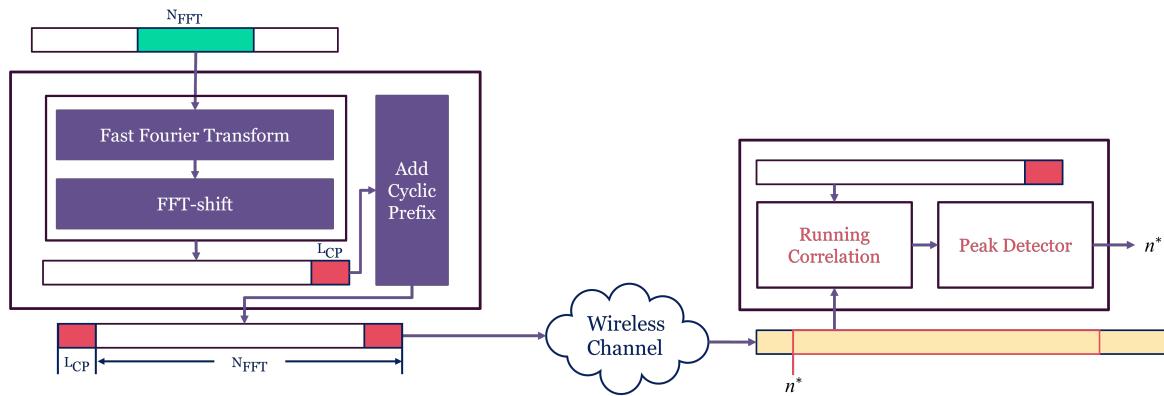


Figure 3.2: Frequency domain pilot generation and correlation based coarse time synchronization.

The output of running correlator is passed to peak detector which finds the sequence with a clear dominant spike and the index of the peak,

$$n^*, i^* = \operatorname{argmax}_{n,i} r_i(n) \quad (3.2)$$

3.3 | Coarse Downlink Synchronization in 5G

The 5G networks use synchronization signal block (SSB) for downlink (DL) synchronization. SSB plays a crucial role in initial access when there is no common understanding of time and frequency between the base station (BS) and user equipment (UE). The SSB is used to time synchronization, frequency synchronization, carrier frequency offset (CFO) estimation, and acquiring master information block (MIB). The details of SSB are provided in section.

3.3.1 | Structure of SSB

The SSB is 4 OFDM symbols duration and 20 resource blocks (240 sub-carrier) wide as shown in figure 3.3. It uses these time frequency resource to consist of 4 components crucial to meet the objective of downlink synchronization,

- Primary Synchronization Signal (PSS),
- Secondary Synchronization Signal (SSS),
- Physical Broadcast Channel (PBCH) symbols,
- Demodulation Reference Signal for PBCH (DMRS-PBCH)

SSB is transmitted in a burst called as SSB burst which carries 4, 8 or 64 synchronization signal blocks depending on the frequency band of operation. A burst spans a half frame transmitted by BS either in first half of the frame or second half of the frame. The SSB burst is transmitted periodically every 5ms/10ms/20ms/40ms/80ms/160ms based on network requirements. The shorter SSB burst period enables quick cell acquisition for UEs but higher network overhead. In this chapter, we will limit the scope of our discussion to PSS and its utility for downlink **time synchronization** and **cell camping**.

3.3.2 | Physical Cell Identities (PCID)

Physical Cell ID (PCID) in 5G refers to an identifier used to distinguish and identify different cells within a wireless network. It is specifically associated with the physical layer of the network and plays a crucial role in cell identification for both the user equipment (UE) and the network infrastructure. In 5G networks, PCID is a part of the cell identity information that helps UEs to differentiate between neighboring cells. The PCID is a numerical value assigned to each cell within a specific frequency band and represents a unique identifier for that cell. It is used by UEs to identify and communicate with the serving cell, especially in scenarios where multiple cells are operating on the same frequency. The PCID,

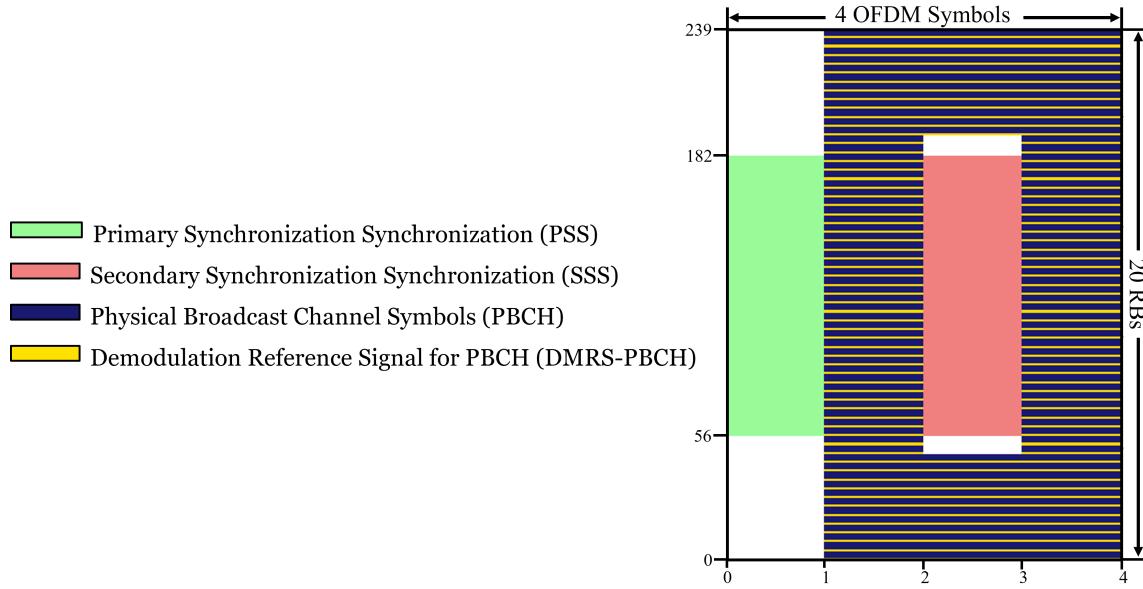


Figure 3.3: 5G Synchronization Signal Block (SSB).

along with other cell-specific parameters, aids in cell selection and reselection processes performed by UEs to maintain a stable and efficient connection as they move within the network coverage area.

The physical cell ID (N_{ID}^{cell}) is segregated into two parts cell ID-1 ($N_{ID}^{(1)}$) and cell ID-2 ($N_{ID}^{(2)}$) related via expression,

$$N_{ID}^{cell} = 3 * N_{ID}^{(1)} + N_{ID}^{(2)} \quad (3.3)$$

where $N_{ID}^{(2)}$ can take 3 values from the set $N_{ID}^{(2)} \in \{0, 1, 2\}$ and $N_{ID}^{(1)}$ can take 336 values from the set $N_{ID}^{(1)} \in \{0, 1, 2, \dots, 335\}$ resulting in a total of 1008 PCIDs from the set $N_{ID}^{cell} \in \{0, 1, 2, \dots, 1007\}$.

3.3.3 | PSS Generation

The generation of PSS is specified in section 7.4.2.2 of [1]. The generation uses $N_{ID}^{(2)}$ for seed computation. Its is m-sequence generated as follows,

$$d_{PSS}(n) = 1 - 2x(m) \quad (3.4)$$

where $m = (n + 43 * N_{ID}^{(2)})$ for $n = 0, 1, 2, \dots, 127$ provides the 3 possible cyclic shifts of 0, 43, 86 for $N_{ID}^{(2)} \in 0, 1, 2$ respectively to the base sequence $x(m)$. The base sequence is generated as follows,

$$x(i+7) = (x(i+4) + x(i)) \bmod 2 \quad (3.5)$$

where boundary/initial conditions defines first 7 samples of $x(n)$ as $[x(6) \ x(5) \ x(4) \ x(3) \ x(2) \ x(1) \ x(0)] = [1 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0]$.

Important: 5G PSS is choosen from a set of 3 possible sequence which are cyclic shifted version of a reference m-sequence. The UE when seeking to synchronize in time actually correlate the received sequence with 3 possible PSS sequence with an aim to estimate/detect following 3 parameters:

- cell ID-2 ($N_{ID}^{(2)}$),
- time sample offset (n_0),
- frequency offset (k_0).

3.4 | Results

In this section, we will demonstrate some important observations about the design of PSS which plays important role in the performance achievable in 5G networks.

Observation-1: The primary synchronization signals has excellent auto-correlation and cross-correlation properties for different cell ID-2.

This property results in accurate cell-ID 2 estimation in downlink. The figure 3.4 demonstrate the cross correlation property of time domain PSS corresponding to cell-ID ($N_{ID}^{(2)}$) 0, 1, and 2.

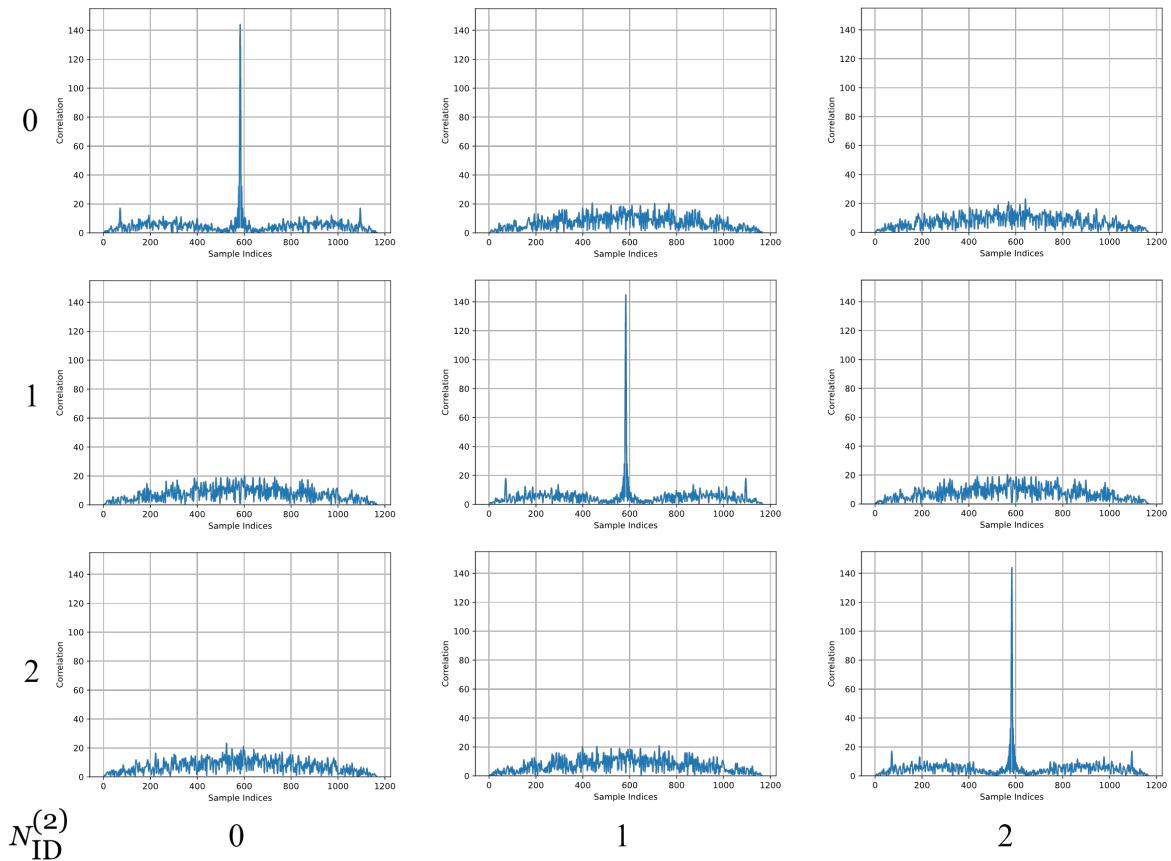


Figure 3.4: Correlation of PSS sequences for $N_{ID}^{(2)} = 0, 1, 2$.

Observation-2: The primary synchronization signals is designed to have good time-frequency correlation properties.

The PSS can be loaded anywhere on the frequency sync raster which eventually is utilized in time domain to estimate the sample index indicate the first index of the OFDM frame. The final correlation plot 3.5 demonstrates the magnitude of the correlation between the selected PSS and received signal.

Observation-3: The PSS is designed to be robust against noise and interference. The UE can decode cell ID-2 ($N_{ID}^{(2)}$) even low transmit power and receiver gains.

The $N_{ID}^{(2)}$ detection performance was analyzed for a a transmitter and receiver pair separated by a distance of 1 meter facing each other for transmitter and receiver gains stated in table 3.1.

The table 3.1 shows the detection of PSS at the UE SDR. The distance between the transmitter SDR and receiver SDR is kept at 1m. The results may vary when experiment is performed by students depending upon multi-path fading. It may be noted that for transmitter gain of -80 dB to -40 dB, PSS is not detected at receiver SDR. From -30 dB transmitter gain onwards, PSS is detectable for each receiver gain. These results may vary depending upon distances between two SDRs and objects therein creating multi-path.

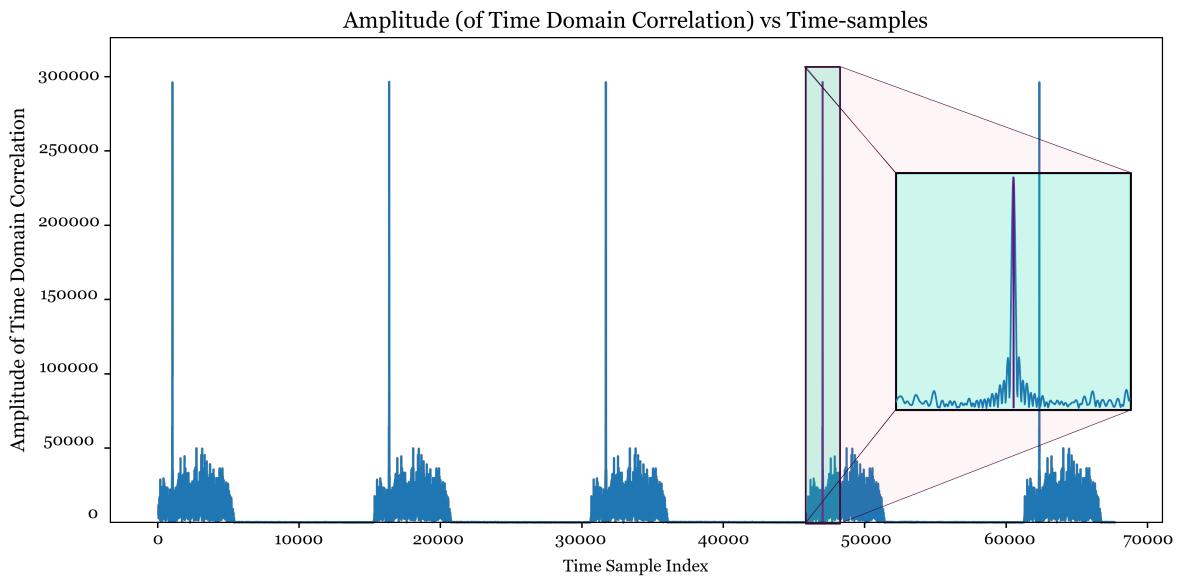


Figure 3.5: Coarse time synchronization using PSS in 5G networks.

Table 3.1: Detection of PSS (Not Detection: **ND** | Not Detection: **D**)

Rx Gain \ Tx Gain	0	10	20	30	40	50	60	70
-80	ND							
-70	ND							
-60	ND							
-50	ND							
-40	ND							
-30	D	D	D	D	D	D	D	D
-20	D	D	D	D	D	D	D	D
-10	D	D	D	D	D	D	D	D
0	D	D	D	D	D	D	D	D

3.5 | Useful Resources

The following are some good articles and papers that we believe are really good read to develop a deeper understanding from here onwards

- Time Synchronization code.
- P1 Procedure: Wide Beam management code [?? update the link].
- Arvind Chakrapani's paper on SSB/PBCH and PRACH design in 5G-NR.

4 | Implementation of OFDM in 5G networks

Orthogonal Frequency Division Multiplexing (OFDM) is a modulation technique used in wireless communication systems to transmit data over radio waves. It works by dividing the available spectrum into multiple orthogonal (non-overlapping) subcarriers, each carrying one constellation symbol. These subcarriers are spaced apart at precise intervals, ensuring they do not interfere with each other. OFDM is widely used in various wireless communication standards such as Wi-Fi, LTE (Long-Term Evolution), WiMAX, and 5G due to its ability to provide high-speed data transmission, robustness to interference, and efficient spectrum utilization. In this chapter, we will discuss the design of OFDM used in the air interface of the 5G networks.

4.1 | Introduction to Multiplexing Schemes

Every generation of cellular networks used a suitable waveform to support multiple access and manage time, frequency, and inter-node interference. The first (1st) and second (2nd) generations of cellular networks used **frequency division multiplexing** (FDD) to allocate resources to multiple user equipments (UEs) in the uplink (UL) and downlink (DL). However, FDD resulted in under-utilization of system bandwidth due to wastage resulting from the guard band. The guard band in FDD is necessary to counter the radiation leaking from the signal transmission in adjacent channels. These losses were reduced in the third (3rd) generation of cellular networks by using **time division duplexing** (TDD) and **code division multiple access** (CDMA) based telephony. The fourth (4th) generation of wireless networks was designed to support mobile phone-based internet and many new use-cases using 20 MHz of bandwidth. The high sample rate resulted in highly frequency-selective channels, demanding complex time-domain receivers. These receivers were power-hungry and required large silicon footprints. This challenge was addressed by OFDM technology, which was a low complexity implementation of multi-carrier modulation (MCM) schemes enabled by FFT algorithms, which can be implemented very efficiently on hardware accelerators.

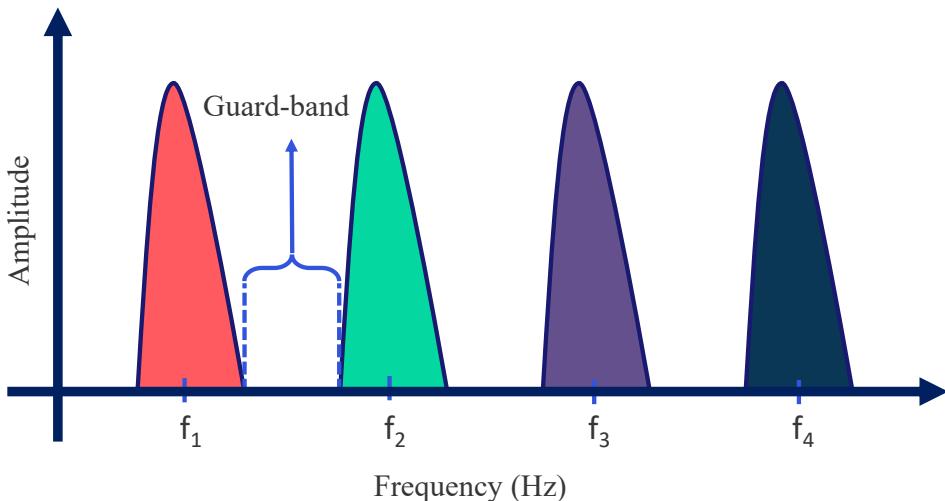


Figure 4.1: Frequency division multiplexing (FDM)

4.1.1 | Multi-Carrier Modulation (MCM)

The MCM transmits information symbols $s_k[n]$ using multiple carriers of frequency (f_k) defined by,

$$m_k(t) = s_k[n].\cos(2\pi f_k t) \quad (4.1)$$

where $f_k = \frac{k}{T}$ is the k^{th} harmonic for the fundamental frequency transmitted in the n^{th} symbol period $t \in [nT, (n+1)T]$. The N carriers can be used to transmit N constellation symbols in a period of T seconds.. The symbols are loaded onto each carrier and all the carriers are superimposed to form a

multi-carrier modulated (MCM) signal,

$$m(t) = \sum_{k=0}^{N-1} m_k(t) \quad (4.2)$$

$$= \sum_{k=0}^{N-1} s_k[n].\cos(2\pi f_k t) \quad (4.3)$$

This MCM signal is up-converted to the carrier frequency f_c before transmission,

$$x(t) = m(t).\cos(\omega_c t) \quad (4.4)$$

as shown in figure 4.2. This signal has spans a pass-band bandwidth of $\frac{N}{T}$ and transmits $\frac{Q_m \cdot N}{T}$ bits per second where Q_m defines the number of bits per constellation (QAM/PSK) symbols. The Q_m is also known as modulation order in 5G networks.

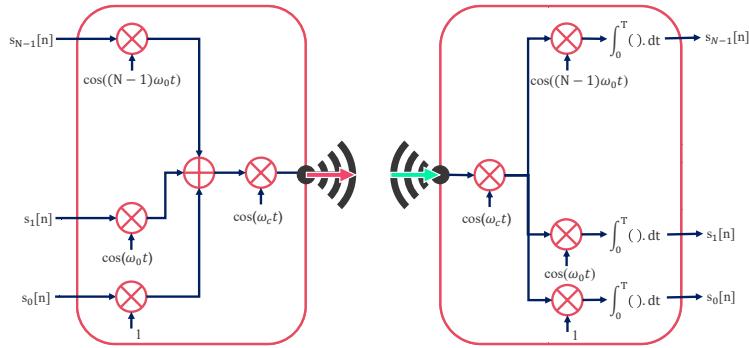


Figure 4.2: Transmitter and Receiver Design for multi-carrier modulations

Owing to the orthogonality property[4], of sinusoids defined by,

$$\int_{nT}^{(n+1)T} 2.\cos(2\pi f_j t).\cos(2\pi f_k t) dt = \begin{cases} 1, & j = k \\ 0, & j \neq k \end{cases} \quad (4.5)$$

the transmitted information symbols can be decoded easily at the receiver. The process of multiplying the carrier tone with the down-converted signal $r(t)$ and integrating it over a period T is called correlation. Each symbol is decoded using a correlator,

$$\hat{s}_k[n] = \int_{nT}^{(n+1)T} 2.r(t).\cos(2\pi f_k t) dt \quad (4.6)$$

The multi-carrier modulation schemes have the ability to divide the entire carrier bandwidth into multiple subcarriers, where each subcarrier can carry information that can be decoded independently at the receiver. The property that makes MCM particularly suitable for wideband communication is the simplicity in demodulation. The multi-path channel is modelled as,

$$h(t, \tau) = \sum_{l=0}^{L-1} \alpha_l.\delta(t - \tau_l) \quad (4.7)$$

This equation of wireless channel results in a system model defined by,

$$y(t) = x(t) \otimes h(t, \tau) \quad (4.8)$$

$$= x(t) \otimes \sum_{l=0}^{L-1} \alpha_l.\delta(t - \tau_l) \quad (4.9)$$

$$= \sum_{l=0}^{L-1} \alpha_l.x(t - \tau_l) \quad (4.10)$$

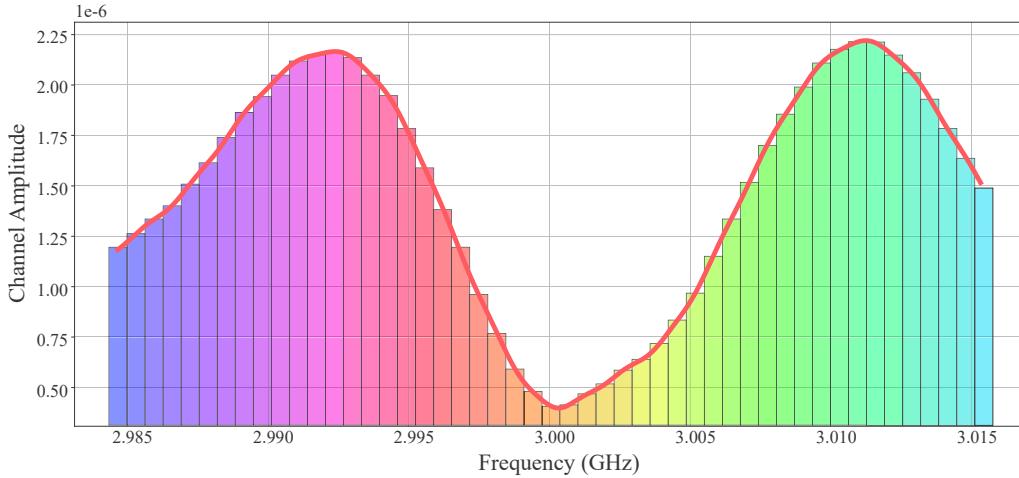


Figure 4.3: Frequency Selectivity in channel due to multipath propagation

where \otimes is the convolution operator and $x(t)$ is the pass-band transmitted signal. The base-band representation of this received signal is given by

$$r(t) = \sum_{l=0}^{L-1} \alpha_l \cdot m(t - \tau_l) \quad (4.11)$$

This signal when passed through the k^{th} correlator results in,

$$\begin{aligned} \hat{r}_k[n] &= \int_{nT}^{(n+1)T} 2.r(t).\cos(2\pi f_k t)dt \\ &= \int_{nT}^{(n+1)T} 2 \cdot \sum_{l=0}^{L-1} \alpha_l \cdot m(t - \tau_l) \cdot \cos(2\pi f_k t) dt \\ &= \sum_{l=0}^{L-1} 2.\alpha_l \cdot \int_{nT}^{(n+1)T} m(t - \tau_l) \cdot \cos(2\pi f_k t) dt \\ &= \sum_{l=0}^{L-1} 2.\alpha_l \cdot \int_{nT}^{(n+1)T} \sum_{k=0}^{N-1} s_k[n] \cdot \cos(2\pi f_k(t - \tau_l)) \cdot \cos(2\pi f_k t) dt \\ &= \sum_{l=0}^{L-1} 2.\alpha_l \cdot \sum_{k=0}^{N-1} s_k[n] \cdot \int_{nT}^{(n+1)T} \cos(2\pi f_k(t - \tau_l)) \cos(2\pi f_k t) dt \\ &= \sum_{l=0}^{L-1} 2.\alpha_l \cdot \sum_{k=0}^{N-1} s_k[n] \cdot \cos(2\pi f_k \tau_l) \cdot \delta(k - j) \\ &= \sum_{l=0}^{L-1} \alpha_l \cdot s_k[n] \cdot \cos(2\pi f_k \tau_l) \\ &= s_k[n] \cdot \sum_{l=0}^{L-1} \alpha_l \cdot \cos(2\pi f_k \tau_l) \quad (4.12) \\ &= s_k[n] \cdot h_k[n] \quad (4.13) \end{aligned}$$

The result in equation 4.13 very clearly shows that the multiplicative relation between the transmitted and received signal unlike the correlative relation in time domain. The correlation based relation uses very complex signal processing to cancel the inter-symbol interference to decode the signal accurately at the receiver. These algorithms consumes very high power and requires expensive circuitry in comparison to receivers with multiplicative relation which only requires a division to decode the information. However, its important to note that each correlator consists of one local oscillator and one integrator. Hence transmitting N symbols in time period T requires N oscillator at the transmitter and N correlators

at the receiver. This results in high complexity, power consumption and cost. Although each subcarrier experiences a flat fading, the wireless channel in frequency domain $H(f) = \sum_{l=0}^{L-1} \alpha_l e^{i \cdot 2\pi \cdot f \cdot \tau_l}$ offers different gain and non-linear phase to each of these sub-carriers as shown in figure 4.3. However, this challenge can be addressed using simple channel estimation and interpolation schemes which will be discussed in details in chapter 7.

4.1.2 | Efficient Implementations of MCM

The major limitations of the the MCM are its high computation complexity and high cost. Both these drawbacks are rooted in modulation and demodulation schemes given by,

$$\begin{aligned}
 \text{MCM Modulation} &= \sum_{k=0}^{N-1} s_k[n] \cdot \cos(2\pi k \cdot \Delta f t) \\
 &= \mathcal{R} \left\{ \sum_{k=0}^{N-1} s_k[n] \cdot e^{2\pi k \cdot \Delta f t} \right\} \\
 &= \mathcal{F}^{-1} \{ s_k \} \text{ where } s_k = [s_k[0], s_k[1], \dots, s_k[N-1]] \\
 \text{MCM Demodulation} &= \frac{2}{T} \int_{nT}^{(n+1)T} r(t) \cdot \cos(2\pi f_k t) dt \\
 &= \mathcal{R} \left\{ \frac{2}{T} \int_{nT}^{(n+1)T} r(t) \cdot e^{-2\pi k \cdot \Delta f t} dt \right\} \\
 &= \mathcal{F} \{ r(t) \}
 \end{aligned}$$

If the modulated and demodulated signal are samples at a rate of $f_s = N \cdot \Delta f$ where $\Delta f = \frac{1}{T}$, the Fourier transform/series can be replaced with Discrete Fourier Transform which can be implemented very efficiently using fast Fourier transform (FFT) algorithms. Furthermore, FFT has very low complexity of the order of $N * \log_2(N)$ in comparison MCM and DFT. Both these characteristics helps it overcome the limitations of MCM. However, this implementation suffers from inter-symbol interference due to multi-path propagation resulting in reception of multiple delayed and attenuated copies of the transmitted signal. This problem can be easily addressed by inserting the cyclic prefix to the transmitted signal. The cyclic prefix unlike the zero prefix not only mitigate the inter-symbol interference but also preserve the characteristics of the FFT. This superposition of multiple orthogonal tones, each loaded with information symbol, inserted with cyclic prefix is called orthogonal frequency division multiplexing (OFDM). The OFDM exists in multiple flavours which we will discuss in upcoming sections.

4.2 | Classification of OFDM

As discussed in previous sections, OFDM provides the low complexity receiver implementation of MCM for wideband frequency selective channels. The two most commonly used implementations are

- Cyclic Prefix based OFDM (CP-OFDM) and,
- Discrete Fourier Transform-spread-OFDM (DFT-s-OFDM)

The DFT-s-OFDM is also called as Single Carrier Frequency Division Multiplexing (SC-FDMA). We will discuss both these waveform in details in upcoming sections.

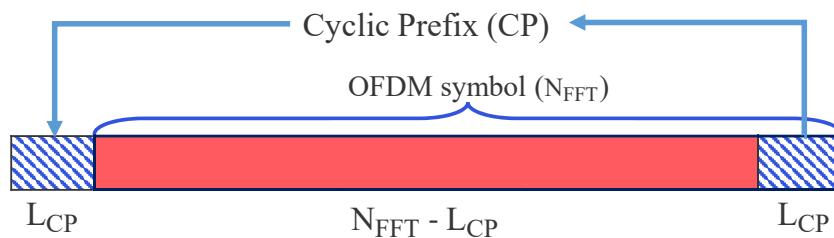


Figure 4.4: Visualization of cyclic prefix (CP)

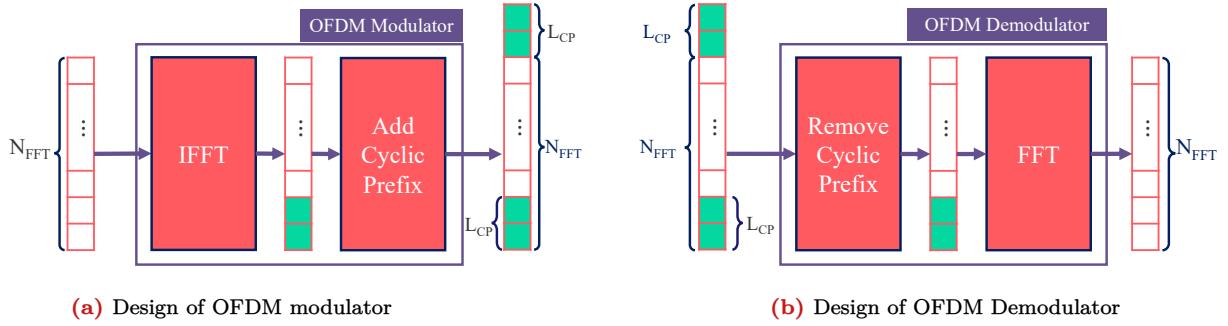


Figure 4.5: Implementation of OFDM

4.2.1 | Cyclic Prefix (CP) based OFDM

The CP-OFDM as discussed in previous section performs two operations:

- Fast Fourier Transform (FFT) to convert the frequency domain block at n -th time instant, $s_n = [s_n[0], s_n[1], \dots, s_n[N_{FFT} - 1]]$ to time domain samples aka time domain OFDM symbol,
- Attaches CP to time-domain OFDM symbol.

The detailed implementation of OFDM is shown in Fig 4.5a. The FFT helps in converting the convolution between the transmitted samples and the channel to the multiplication which reduces the complexity of decoding the data. The cyclic prefix helps meet two objectives,

- mitigate inter-symbol interference.
- preserve the orthogonality between subcarrier.

The receiver performs complementary operation to transmitter where first CP is removed, followed by the translating the time domain samples to frequency domain block using FFT as shown in Fig 4.5b. This implementation allows for much simple channel estimation and equalization discussed in details 7. The OFDM suffers from the problem of low peak to average power ratio (PAPR). The PAPR arises due to constructive interference in the tones in time domain causing the spikes in power for a few time domain samples while the average power stays the same. It limits the radio from transmitting at the maximum transmit power. The large PAPR forces the radio to transmit at much lower power compared to operating point to avoid the amplifier from operating in the non-linear region. The non-linear region causes signal saturation resulting in loss of information.

4.2.2 | DFT-s-OFDM

The cause of high PAPR in OFDM systems is the FFT block, which results in spikes in power due to constructive interference. The effect of PAPR can be reduced by using another DFT of lower order than N_{FFT} to spread the signal. CP-OFDM with DFT as a pre-processing block is called Discrete Fourier Transform-spread-OFDM (DFT-s-OFDM), as shown in Figure 4.6. Spreading by an M_{sc} -point DFT reduces the constructive interference resulting from the de-spreading caused by the FFT block by a factor of M_{sc} . In DFT-s-OFDM, the total number of information/constellation symbols is divided into sets (S), which are computed as follows,

$$\text{Numbers of sets } (S) = \frac{\text{Numbers of information symbols}}{M_{sc}}. \quad (4.14)$$

The number of sets (S) is equal to the number of OFDM symbols allocated to the respective channel, which is PUSCH or PUCCH for format 3 or format 4 in 5G NR. DFT-s-OFDM supports a single layer transmission and is employed only in the uplink to improve the coverage of the network. It is often used only with $\frac{\pi}{2}$ -BPSK modulation when the link-budget is constrained and channel conditions are unfavourable. The illustration of DFT-s-OFDM shown in Fig-4.6 assumes that there is no PTRS transmission along-side the information symbols. The M_{sc} for PUSCH is computed using the expression,

$$M_{sc}^{\text{PUSCH}} = M_{RB}^{\text{PUSCH}} \cdot N_{sc}^{\text{RB}}, \quad (4.15)$$

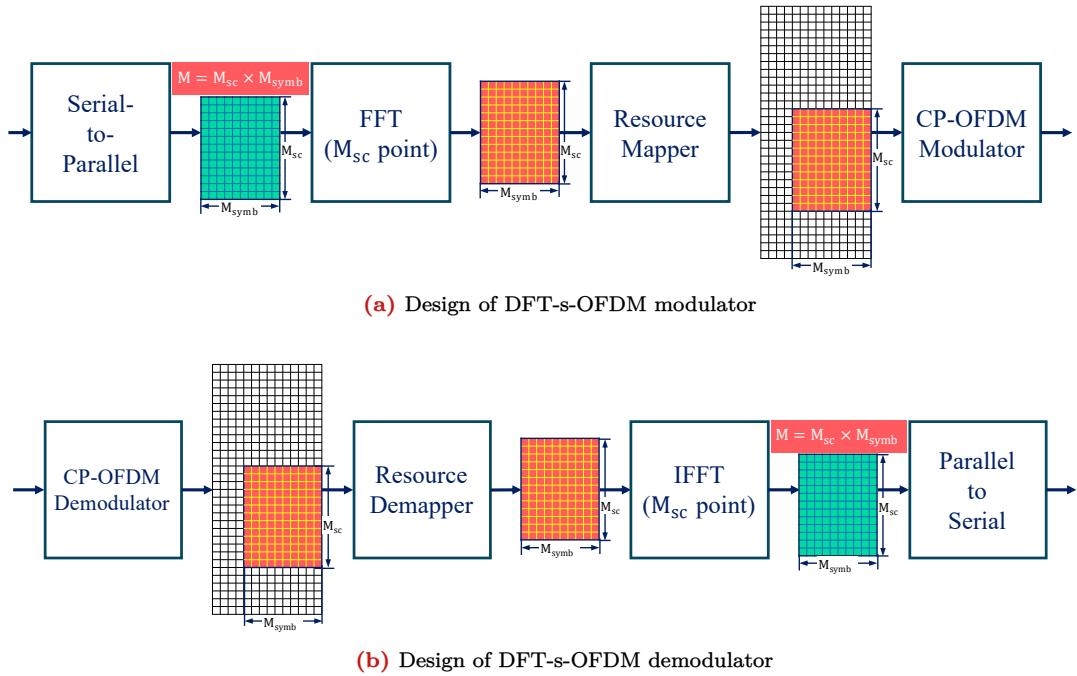


Figure 4.6: Implementation of DFT-s-OFDM

where $N_{sc}^{RB} = 12$ and M_{RB}^{PUSCH} denotes the RBs allocated to UE for data transmission. This variable in 5G networks is expected to satisfy the condition $M_{RB}^{PUSCH} = 2^{\alpha_2}.3^{\alpha_3}.5^{\alpha_5}$ where α_2, α_3 , and α_5 are expected to be non-negative integers.

4.3 | Why Orthogonal Frequency Division Multiplexing in 5G Networks

OFDM is adopted in 5G NR owing to its ability to cope with severe wideband frequency selective fading conditions and simple implementation. The reasons for the wide spread adoption of OFDM in 5G NR and other wideband communication technologies are listed below.

1. OFDM converts the frequency selective channel to flat fading channel at a subcarrier level, as depicted in Figure-4.3.
 - [a] Simplifies the receiver design by converting the wide-band frequency selective channel into multiple narrow-band flat channels for each subcarrier.
2. In OFDM, the orthogonality among sub-carriers is ensured, which improves bandwidth efficiency, spectral efficiency, and enables OFDMA.
3. OFDM can be seamlessly integrated with MIMO.
4. OFDM is based on Fast Fourier Transform (FFT) both at the transmitter and receiver. The FFT has lower complexity, can be easily scaled, and its implementation has a smaller silicon footprint while supporting hardware acceleration.

4.4 | Design of OFDM in 5G

The 5G uses three different implementations of CP-OFDM for PRACH, RIM-RS and rest of the channels. In the chapter, we will discuss the third one which is used by most of the physical channels. The implementation of CP-OFDM for PDSCH, PDCCH, PBCH, PUSCH, PUCCH and other chains in 5G is given by,

$$\bar{s}_l^{(p,\mu)}(t) = \sum_{k=0}^{N_{grid,x}^{\text{size},\mu} \cdot N_{sc}^{RB}-1} a_{k,l}^{(p,\mu)} e^{j2\pi(k+k_0^\mu - N_{grid,x}^{\text{size},\mu} N_{sc}^{RB}/2) \cdot \Delta f \cdot (t - N_{CP,l}^\mu T_c - t_{start,l}^\mu)} \quad (4.16)$$

where

- l : OFDM symbol index,
- p : Antenna port index,
- t : Time instant in time interval, $t_{\text{start},l}^{\mu} \leq t < t_{\text{start},l}^{\mu} + T_{\text{symb},l}^{\mu}$, of l -th OFDM symbol
- μ : Numerology,
- μ_0 : The largest μ value a UE is expected to support,
- k : Subcarrier index,
- k_0^{μ} : Subcarrier offset between the grids of numerology μ and μ_0 ,
- $T_{\text{symb},l}^{\mu}$: Duration of l -th OFDM symbol $= (N_u^{\mu} + N_{\text{CP},l}^{\mu})T_c$
- $N_{\text{CP},l}^{\mu}$: Cyclic prefix length for OFDM symbol- l with numerology (μ),
- N_u^{μ} : FFT-size for numerology (μ) $= 2048\kappa.2^{\mu}$,
- T_c : Basic time unit for NR computed as $\frac{1}{\Delta f_{\text{max}} \cdot N_f}$ where $\Delta f_{\text{max}} = 480 \times 10^3$ and $N_f = 4096$,
- $N_{\text{grid},x}^{\text{size},\mu}$: The size of resource grid. It represents the total bandwidth available at the BS.
- N_{sc}^{RB} : Number of subcarriers per resource block. It is fixed to 12 in 5G,
- $a_{k,l}^{(p,\mu)}$: Information loaded on subcarrier- k of antenna port (p) and numerology (μ) for OFDM symbol (l),
- $\bar{s}_l^{(p,\mu)}(t)$: Time continuous signal on antenna port (p) and numerology (μ) for OFDM symbol (l).

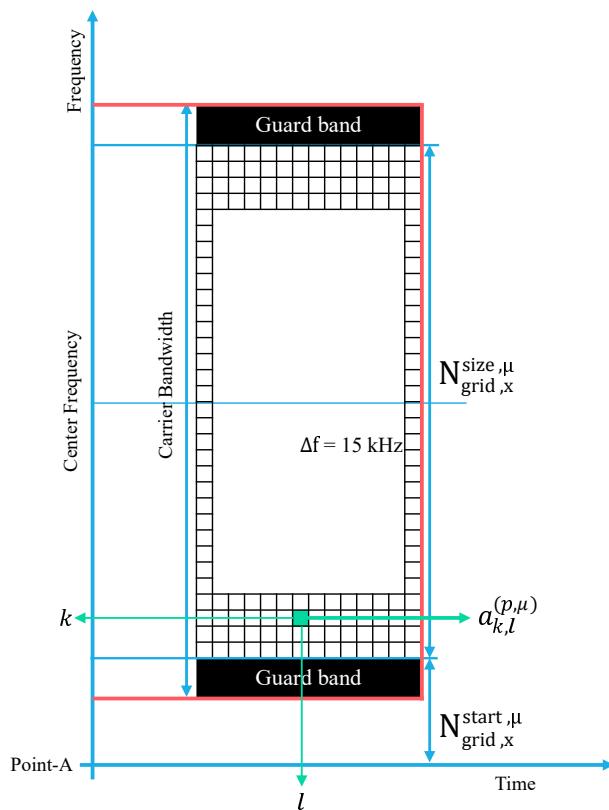


Figure 4.7: OFDM Resource Grid and Channel Bandwidth.

4.4.1 | Concept of Resource Grid

The 5G implementation of OFDM (Orthogonal Frequency Division Multiplexing) is based on the concept of the resource grid. A resource grid is a matrix or tensor that spans the complete bandwidth available at the base station across frequency and one slot (which typically consists of 14 OFDM symbols) across time,

as shown in Fig 4.7. The resource grid is organized into resource elements, which are Δf wide and span one OFDM symbol ($T_{\text{symb},l}^{\mu}$), capable of carrying one constellation symbol worth of information. The dimensions of the resource grid are $N_{\text{ports}} \times N_{\text{symb}} \times N_{\text{sc}}$, where the number of OFDM symbols and the number of sub-carriers in a resource grid are given by $N_{\text{symb}} = 142^{\mu}$ and $N_{\text{sc}} = 12N_{\text{grid},x}^{\text{size},\mu}$, respectively. However, the BS can maintain the resource grid in units of 14 symbols as well, with separate resource grids for each numerology. Resources for data transmission, CSI-RS for channel sounding, sending control information, and broadcasting network information are allocated in the resource grid only.

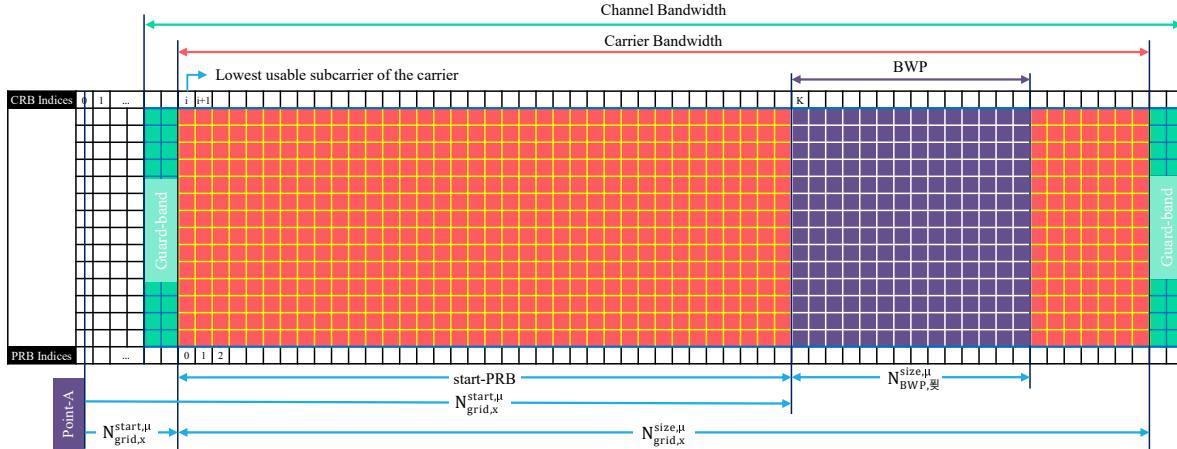


Figure 4.8: Channel Bandwidth, Carrier Bandwidth and BWP in 5G Networks.

4.4.2 | Numerologies in 5G Networks

The release 17 of 5G networks, unlike 4G networks, supports 7 numerologies/subcarrier spacing as shown in table 4.1. These numerologies allow the BS to find a judicious trade-off between receiver complexity, latency, cell coverage, robustness against CFO and Doppler.

Table 4.1: Numerologies in 5G

Numerology (μ)	Subcarrier spacing ($\Delta f = 15.2^{\mu}$) in kHz	Cyclic Prefix (CP) types
0	15	Normal
1	30	Normal
2	60	Normal and Extended
3	120	Normal
4	240	Normal
5	480	Normal
6	960	Normal

A subcarrier spacing of 240 kHz is reserved for the transmission of SSB, while other subcarrier spacings are employed for data transmission. Numerologies 0 and 1 (15/30 kHz) are used in FR1 (frequency range-sub 6 GHz). Numerology 2 (60 kHz) can be used in frequency ranges FR1 and FR2. Numerology 2 supports extended CP, which helps mitigate ISI under severe channel conditions. Numerology 3 and beyond are used in FR2, as FR2 is utilized in small cells with smaller propagation delay and delay spread, making them more suited for higher numerologies.

4.4.3 | Cyclic Prefix Length Selection in 5G Networks

The cyclic prefix length in 5G is not the same for all OFDM symbols. For 2 out of every $14 * 2^{\mu}$ symbols, a slightly longer cyclic prefix is used to ensure that the duration of a slot is $\frac{1}{2^{\mu}}$ milliseconds. Furthermore, with a 60 kHz subcarrier, the symbol duration reduces to 0.25 ms, and the cyclic prefix reduces proportionally, resulting in coverage of barely $180000/N_{\text{FFT}}$ meters, which is approximately 180

meters at a 200 MHz bandwidth. Hence, an extended cyclic prefix was introduced, as shown in equation 4.17. With the extended cyclic prefix, the coverage extends to 625 meters for a 60 kHz subcarrier spacing with a channel bandwidth of 200 MHz.

$$N_{\text{CP},l}^{\mu} = \begin{cases} 512\kappa.2^{-\mu} & \text{extended CP (only applicable for } \Delta f = 60 \text{ kHz)} \\ 144\kappa.2^{-\mu} + 16\kappa & \text{normal CP, } l = 0 \text{ or } l = 7.2^{\mu} \\ 144\kappa.2^{-\mu} & \text{normal CP, } l \neq 0 \text{ and } l \neq 7.2^{\mu} \end{cases} \quad (4.17)$$

Furthermore, this variation in the length of CP for different symbols results in irregular OFDM symbol boundaries for the time (t) parameter stated in equation 4.16. The starting time instant for OFDM symbol- l is given by,

$$t_{\text{start},l}^{\mu} = \begin{cases} 0 & l = 0 \\ t_{\text{start},l-1}^{\mu} + T_{\text{symb},l-1}^{\mu} & \text{otherwise} \end{cases} \quad (4.18)$$

4.5 | OFDM Implementation in 5G Toolkit

The implementation of OFDM consists of modulation at the transmitter and demodulation at the receiver. At the transmitter, the frequency resource grid is converted into time-domain OFDM samples before transmission across the air interface. Conversely, at the receiver, the time-domain samples are converted back into the frequency resource grid. In the subsection below, the detailed implementation of the OFDM transmitter and receiver is described. In 5G, the resources allocated for data transmission are within this resource grid.

4.5.1 | OFDM Implementation: Transmitter

The CP-OFDM modulation in the 5G Toolkit is implemented at the transmitter side using the class [OFDMModulator](#). This module requires the length of the CP (L_{CP}) as a parameter. The OFDM modulation of the input resource grid is performed, which is expected to be either a 1D or 2D matrix or a tensor. The modulation is carried out across the last dimension of the resource grid, and the size of the last dimension is considered as the FFT size. The frequency domain grid of shape 14 (number of OFDM symbols) $\times N_{\text{FFT}}$ is passed on to the object of the class, which generates the time-domain OFDM samples at the output of size $(N_{\text{FFT}} + L_{\text{CP}}) \times$ number of OFDM symbols.

Note: The OFDM modulator attaches the same length CP to all OFDM symbols. To conform to 3GPP standards, we pass the grid symbol by symbol to the OFDM modulator, where the length of the CP attached to each symbol is computed using the `lengthCP` variable returned by the [TimeFrequency5GParameters](#) class in the 5G Toolkit.

4.5.2 | OFDM Implementation: Receiver

The received time-domain samples are passed to the OFDM demodulator defined in the 5G Toolkit by [OFDMDemodulator](#). This module removes the L_{CP} samples, assuming them as the cyclic prefix, and reconstructs the OFDM resource grid. The module expects two parameters as inputs

- FFT-size (N_{FFT}) and,
- length of CP (L_{CP})

to reconstruct the resource grid of size $(\dots, N_{\text{symb}}, N_{\text{FFT}})$ from the received time-domain samples of shape $(\dots, N_{\text{symb}} \cdot (N_{\text{FFT}} + L_{\text{CP}}) + n)$, where n is a non-negative number.

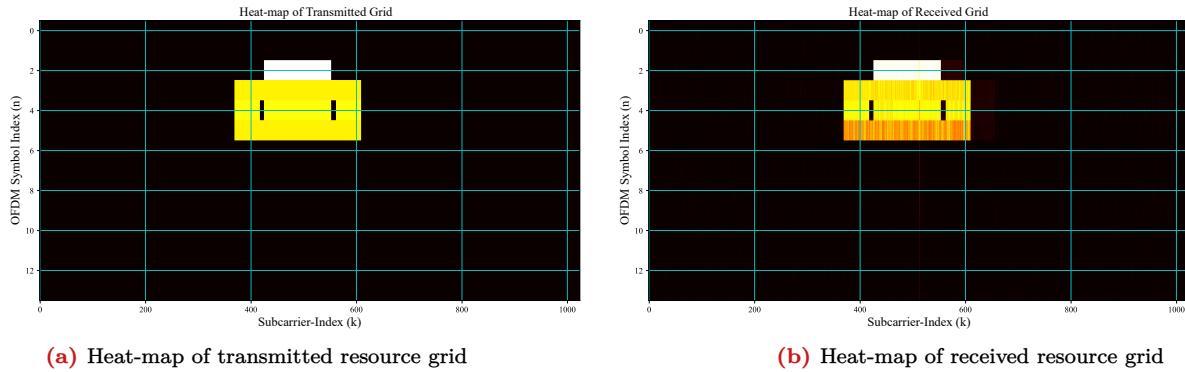
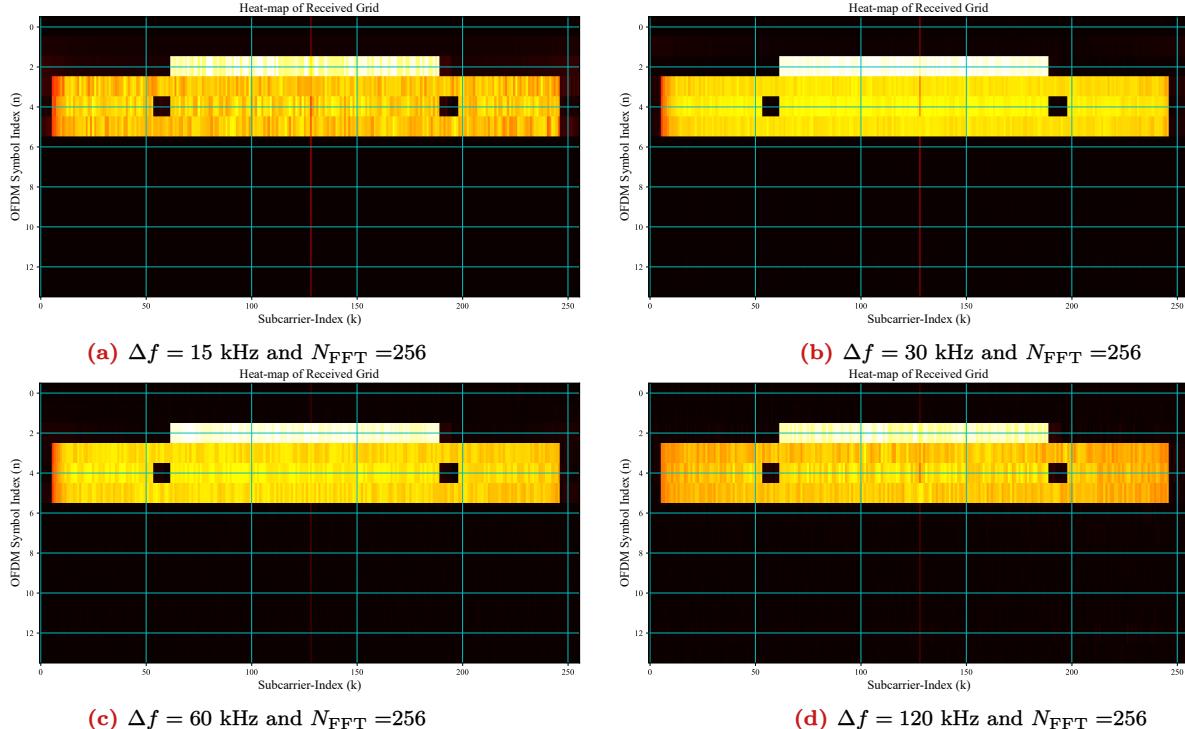
4.6 | Results

In this section, we will demonstrate the effect of the OFDM and system parameters on the behavior of the resource grid, frequency selectivity, and a few hardware impairments such as CFO. The observations will be made based on the parameters stated in table 4.2.

Table 4.2: Simulation Parameters for OFDM evaluations

Parameters	Value
Center frequency (f_c)	1000 MHz
Bandwidth (B)	5 MHz
FFT-size (N_{FFT})	2048/1024/512/256
Subcarrier spacing (Δf)	15/30/60/120 kHz
Transmitter-receiver separation	1 m

Observation-1: The time and frequency location of the information in transmitted and received OFDM resource grid are exactly the same if the systems are accurately synchronized.

**Figure 4.9:** Comparison between the tx and rx resource grid for $N_{FFT} = 1024$ and $\Delta f = 15$ KHz**Figure 4.10:** Analysis of quality of received grid for different subcarrier spacing for fixed FFT size.

The heat-map of power distribution in the resource grid transmitted and received by the SDR is shown in Fig 4.9. It can be seen that if the SDR receiver synchronizes in time accurately and mitigates the CFO and Doppler offset to a low enough level, the resource grids can be reconstructed pretty accurately owing

to the orthogonality between the subcarrier tones. The red lines in figure 4.9b shows distortions in the received grid due to DC offset, residual frequency offsets, and thermal noise.

Observation-2: The impact of residual frequency offsets is lower on higher subcarrier spacing for a fixed FFT-size.

The effect of the residual frequency offset is directly proportional to $\frac{f_{RFO}}{\Delta f}$. It can be observed from the equation that higher is the subcarrier spacing lower is the effect of the residual frequency offsets on the performance of OFDM. The same inferences can be drawn from the results shown in Fig-4.10 where SDR based emulations are performed for different subcarrier spacing Δf for a fixed FFT-size.

Observation-3: The impact of residual frequency offsets is lower on OFDM system using higher sample rates for a fixed subcarrier spacing.

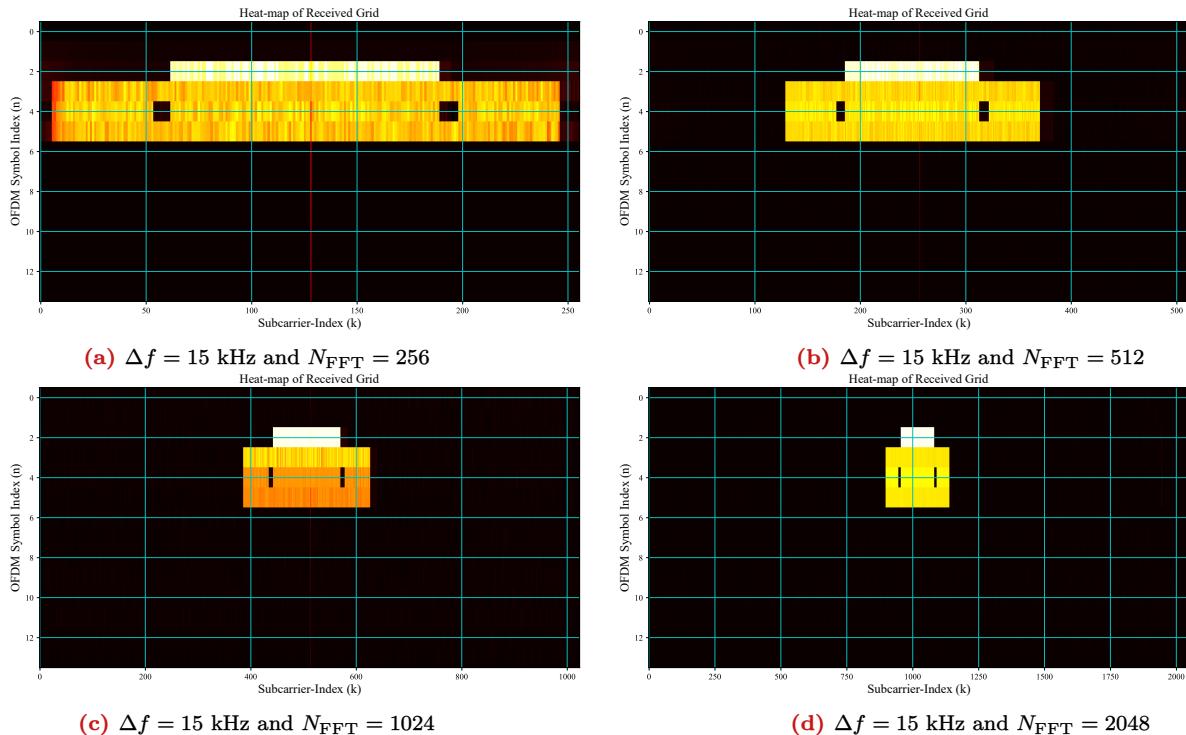


Figure 4.11: Analysis of quality of received resource grid for different FFT-size for a fixed $\Delta f = 15$ kHz.

It is clearly observed from the above figures that there is no significant effect of subcarrier spacing on reception of SSB. However, for subcarrier spacing of 15 KHz, the fading is a little higher.

4.7 | Useful Resources

The following are some good references to understand OFDM and OFDM in 5G Networks. You can find the link to the codes used in this tutorial.

- [OFDM Documentation in 5G Toolkit](#).
- [A Tutorial in 5G Networks](#).
- [Detailed Python Code of this Experiment](#).
- [A Video Tutorial for this Experiment](#).
- [Details of OFDM implementation in 5G networks \[1\]](#).

5 | Fine Time Synchronization in 5G Networks using PSS and SSS

In this chapter, we will cover the remaining details of time synchronization. As discussed in Chapter 3, time synchronization is performed in two steps. The first step establishes coarse time synchronization, which estimates the integer time offset. This step is typically performed using time correlation-based methods. These methods are very robust but not very accurate, as they cannot estimate the fractional sample offset. The fractional sample offset is often estimated in the second step of the time synchronization called fine time synchronization.

5.1 | Why is Fractional/Fine Time Synchronization important?

Fine time synchronization helps align the frame boundaries more accurately. This can be useful for many features in 5G. Some of these are listed below:

- **Coordination of Advanced Antenna Techniques:** 5G networks often utilize advanced antenna technologies such as beamforming and massive MIMO (Multiple Input Multiple Output). Fine time synchronization ensures precise alignment in the timing of transmissions and receptions, allowing these techniques to be effectively coordinated. This coordination optimizes the use of spatial resources, enhances coverage, and improves overall network performance.
- **Reducing Interference:** Fine time synchronization helps in minimizing interference between neighboring cells or base stations. By aligning the timing of transmissions, interference can be mitigated, leading to improved signal quality and network efficiency. This is particularly important in dense urban environments where multiple cells operate in close proximity.
- **Meeting Latency Requirements:** 5G networks aim to support applications with stringent latency requirements, such as ultra-reliable low-latency communication (URLLC) and industrial automation. Fine time synchronization contributes to meeting these low-latency demands by ensuring precise timing alignment between network elements.
- **Supporting Time-Sensitive Applications:** Various applications, including augmented reality (AR), virtual reality (VR), and tactile Internet, rely on accurate time synchronization for a seamless user experience. Fine time synchronization helps in achieving the timing precision needed for these time-sensitive applications.
- **Enhancing Positioning Accuracy:** For location-based services and applications that require accurate positioning information, fine time synchronization is crucial. It enables precise positioning techniques, contributing to improved accuracy in location-based services, emergency services, and vehicle-to-everything (V2X) communication.
- **Efficient Use of Frequency Resources:** Fine time synchronization contributes to the efficient use of the frequency spectrum. By aligning the timing of transmissions, it helps in optimizing spectral efficiency, reducing interference, and enhancing the overall capacity of the network.
- **Integration of AI/ML Technologies:** As 5G networks evolve, there is an increasing exploration of the integration of artificial intelligence (AI) and machine learning (ML) technologies. Fine time synchronization is vital for the successful implementation of AI/ML algorithms in optimizing network performance and resource allocation.

In summary, fine time synchronization in 5G networks plays a crucial role in enabling advanced technologies, reducing interference, meeting low-latency requirements, supporting time-sensitive applications, enhancing positioning accuracy, and optimizing the efficient use of frequency resources. These factors collectively contribute to the overall performance, reliability, and capabilities of 5G networks.

5.2 | Fine Time Synchronization Techniques

Fine synchronization aims to estimate the fraction sample offset. The following are the two typically used methods for it,

- Up-sampling the magnitude spectrum
- Frequency/OFDM domain delay estimation and correction

5.2.1 | Up-Sampling the Magnitude Spectrum

The coarse time synchronization computes the time correlation of all the possible pilots with the received signal. However, the time resolution of the magnitude plot for this method is generally poor. This resolution can be improved using two techniques:

- FFT based up-sampling
- Magnitude interpolation based up-sampling

Both these methods generally yield very good performance but requires higher memory for storing received signal and this memory consumption scales linearly with the up-sampling factor.

5.2.2 | Frequency/OFDM Domain Delay Estimation and Correction

This methods convert the time domain offset into frequency domain phase. IN frequency domain, the received grid is compared against the transmitted grid to estimate the phase. This phase is used to correct the received grids. The implementation of this method is detailed in figure [??]. The method has the lowest complexity but performance might degrade for highly frequency selective channels.

5.3 | Fine Time Synchronization in 5G

The fine time synchronization is also performed using primary synchronization signal (PSS). The only difference lie in following aspects.

5.4 | Results

In this experiment, we will discuss the performance of all the three methods as discussed above. Furthermore, we detail down some insights of all the three methods which results in superiority or inferiority of one over the other.

Observation-1: The performance of interpolation based up-sampling method relies heavily on the amount of information and details in magnitude of time correlation.

The performance of this method deteriorates significantly if the magnitude of the time correlation is very noisy.

Observation-2: FFT based up-sampling methods resulting in superior performance in comparison to method-1 but the computational complexity and memory required are significantly higher.

Observation-3: THe figure-[??] clearly shows that the performance and computational complexity of frequency domain time offset correction is significantly better than both the former methods for frequency flat channel but the performance degrades significantly for multi path channels.

5.5 | Useful links

The following are some useful reference to the codes used in the experiments, and some other advanced algorithms for further reading on fine time synchronization.

- Link to Code-1.
- Link to Code-2
- Link to Code-3.
- Method-1
- Method-2

6 | Carrier Frequency Offset Estimation and Correction in 5G Networks

The air interface of 5G NR is based on orthogonal frequency division multiplexing (OFDM), which divides the wideband carrier into small orthogonal subcarriers. Unlike the wideband signal, these subcarriers experience flat fading, resulting in significantly reduced receiver complexity and improved spectral efficiency. However, OFDM is very sensitive to time and frequency synchronization errors. These errors can arise in the frequency or time domains due to carrier frequency offset (CFO) and symbol time offset (STO), respectively. This experiment will focus on the carrier frequency offset and its impact on the system, as well as algorithms to estimate and correct the CFO using the synchronization signal block (SSB).

6.1 | What is Carrier Frequency Offset?

The carrier frequency offset is attributed to a mismatch in the local oscillator frequency of the transmitter and receiver. This offset causes a shift in the phase of the received symbols, as shown in Figure 6.1, which leads to incorrect decoding and degradation of performance. When a UE is moving, CFO estimation becomes even more difficult.

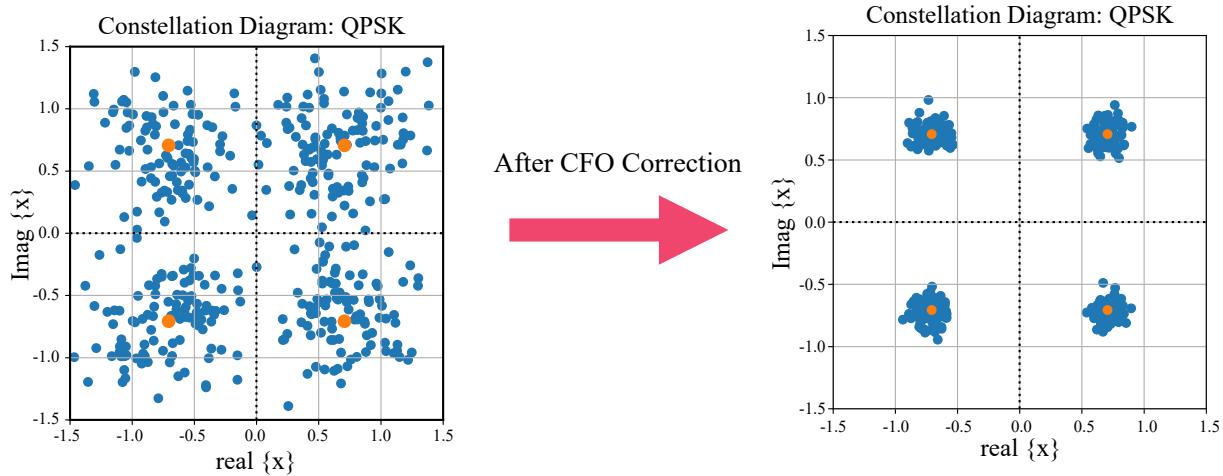


Figure 6.1: Carrier Frequency Offset (CFO) in the SSB/PBCH in 5G Networks.

The primary objective of the UE is to decode the data transmitted by the BS. However, initially, the UE doesn't know the OFDM symbol/frame boundary, which is accomplished using the process of time synchronization detailed in Chapter-3. Once the transmitter and receiver are time synchronized, the receiver estimates the CFO and compensates its effect from the received samples. Inaccuracies in CFO correction result in shifts in subcarrier frequencies, compromising the orthogonality among the sub-carriers and leading to inter-carrier interference (ICI) as shown in Fig-6.2. These errors can accumulate and significantly deteriorate the performance of SSB detection. Hence, it becomes important to estimate and correct CFO accurately at the UE. In the next section, we will discuss different techniques that have been widely used to estimate CFO.

6.2 | CFO Estimation Techniques

The most commonly used methods to estimate CFO are cyclic prefix (CP)-based CFO estimation and pilot-based CFO estimation. CFO estimation can be performed in the time domain or frequency domain. In the frequency domain, only the integer part of CFO can be estimated, while the fractional CFO can only be estimated in the time domain. Fractional CFO results in phase rotation across time samples, which can be estimated using either a pilot-based scheme or a CP-based scheme.

6.2.1 | System Model

For the estimation and correction of CFO, we assume that the wireless channel between the transmitter and receiver undergoes flat fading, modelled as follows,

$$y[n] = h \cdot x[n] + w[n], \quad (6.1)$$

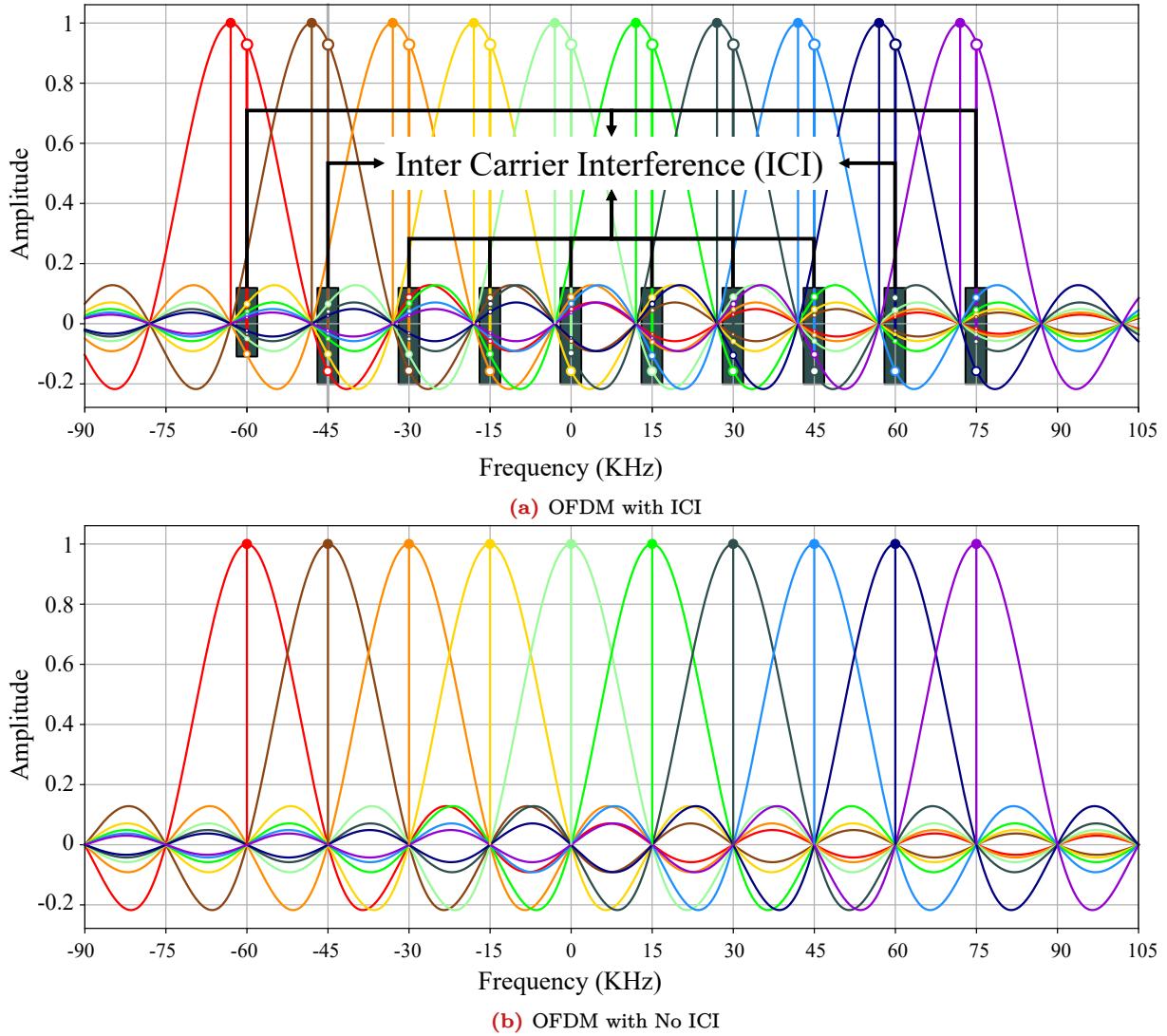


Figure 6.2: Inter carrier interference (ICI) in OFDM system

where $y[n]$, $x[n]$, $w[n]$ and h are received signal, transmitted signal, noise and flat fading channel between transmitter and receiver respectively at time instant n . This system model proposed in equation-6.1 can be modified to capture the effect of CFO at the receiver as follows,

$$y[n] = h \cdot x[n] e^{j2\pi\epsilon n} + w[n]. \quad (6.2)$$

where ϵ denotes the fractional part of the CFO resulting in phase rotation of transmitted symbols.

6.2.2 | CP Based CFO Estimation

The cyclic prefix (CP) is employed in an OFDM system, where the last L_{CP} samples of the OFDM symbol are appended to the front of it, as shown in Fig-6.3. The CP helps the OFDM system mitigate the effects of intersymbol interference (ISI) caused by multipath propagation. The large number of multi-path components result in a higher delay spread. To prevent ISI, the length of L_{CP} must be larger than the delay spread of the channel.

The length of one OFDM symbol is $N_{FFT} + L_{CP}$, as shown in Fig-6.3, where N_{FFT} is the size of the FFT, and L_{CP} is the length of the cyclic prefix (CP) used in OFDM modulation. In a OFDM symbol, the first and the last L_{CP} sample are same,

$$x[n] = x[n + N_{FFT}], n = 0, 1, 2, \dots, L_{CP} - 1.$$

This property of OFDM can be utilized for CFO estimation at the receiver. It can be observed from the equation 6.1, the first L_{CP} received samples of the OFDM symbols can be written as,

$$y[n] = hx[n]e^{j2\pi\epsilon n} + w[n], n = 0, 1, 2, \dots, L_{CP} - 1. \quad (6.3)$$

the last L_{CP} received samples are given as,

$$\begin{aligned} y[n + N_{FFT}] &= hx[n + N_{FFT}]e^{j2\pi\epsilon(n+N_{FFT})} + w[n + N_{FFT}], n = 0, 1, 2, \dots, L_{CP} - 1. \\ &= hx[n]e^{j2\pi\epsilon(n+N_{FFT})} + w[n + N_{FFT}] \end{aligned}$$

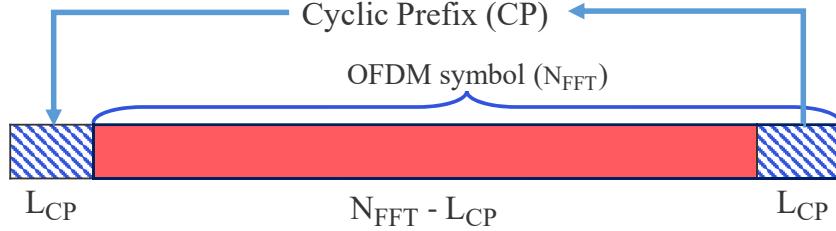


Figure 6.3: Time domain OFDM symbol with information and CP.

The differential phase of the last and first L_{CP} samples can be computed as follows,

$$\begin{aligned} \phi(n) &= \frac{y[n + N_{FFT}]}{y[n]}, n = 0, 1, 2, \dots, L_{CP} - 1. \\ &= e^{j2\pi\epsilon N_{FFT}} + \frac{w[n + N_{FFT}]}{w[n]} \end{aligned} \quad (6.4)$$

The differential phase of $\phi(n)$ can be used to estimate the CFO at time instant n as follows,

$$\hat{\epsilon}(n) = \frac{\arg\{\phi(n)\}}{2\pi N_{FFT}}. \quad (6.5)$$

To reduce the effect of noise, the ML estimation is performed, The ML estimate of CFO ($\hat{\epsilon}$) is computed as,

$$\hat{\epsilon} = \frac{1}{L_{CP}} \sum_{n=1}^{L_{CP}} \hat{\epsilon}(n), \quad (6.6)$$

Algorithm: The CP based CFO Estimation Method

1. Collect the first L_{CP} samples from OFDM symbol

$$\{y[n]\}_{n=1}^{L_{CP}} = \{hx[n]e^{j2\pi\epsilon N_{FFT}}\}_{n=1}^{L_{CP}}.$$

2. Collect the last L_{CP} samples from OFDM symbol.

$$\{y[n + N_{FFT}]\}_{n=1}^{L_{CP}} = \{hx[n + N_{FFT}]e^{j2\pi\epsilon N_{FFT}}\}_{n=1}^{L_{CP}}.$$

3. Compute the differential phase,

$$\phi[n] = \left\{ \frac{y[n + N_{FFT}]}{y[n]} \right\}_{n=0}^{n=L_{CP}-1} = e^{j2\pi\epsilon N_{FFT}}.$$

4. Compute the CFO estimate for time instant-n,

$$\hat{\epsilon}[n] = \frac{\arg\{\phi[n]\}}{2\pi N_{FFT}}.$$

- 5.** Compute the ML estimate of CFO,

$$\hat{\epsilon} = \frac{1}{L_{\text{CP}}} \sum_{n=1}^{L_{\text{CP}}} \hat{\epsilon}[n],$$

- 6.** Perform CFO correction,

$$\hat{y}[n] = y[n]e^{-j2\pi\hat{\epsilon}n},$$

The accuracy of CFO estimation using the CP-based technique depends on the length of the CP. When the length of the CP is shorter, the estimate of CFO tends to be inaccurate, leading to poor CFO correction. This can be significantly improved using a pilot-based scheme, which provides a larger number of observations available for CFO estimation.

6.2.3 | Pilot Based CFO Estimation

As discussed above, in CP based CFO estimation the first and last L_{CP} samples are considered. However, in pilot based CFO estimation all $N_{\text{FFT}} + L_{\text{CP}}$ samples of OFDM symbol are considered. The $N_{\text{FFT}} + L_{\text{CP}}$ received samples of OFDM symbol is given by

$$y[n] = hx[n]e^{j2\pi\epsilon n} + w[n], n = 1, 2, \dots, N_{\text{FFT}} + L_{\text{CP}} \quad (6.7)$$

The channel tap estimate for time index n is estimated as follows,

$$y[n + N_{\text{FFT}}] = hx[n + N_{\text{FFT}}]e^{j2\pi\epsilon(n+N_{\text{FFT}})} + w[n + N_{\text{FFT}}], n = 1, 2, \dots, L_{\text{CP}} \quad (6.8)$$

$$= hx[n]e^{j2\pi\epsilon(n+N_{\text{FFT}})} + w[n + N_{\text{FFT}}] \quad (6.9)$$

$$\begin{aligned} h[n] &= \frac{y[n]}{x[n]} \\ &= he^{j2\pi\epsilon n} + \frac{w[n]}{x[n]} \\ &= he^{j2\pi\epsilon n} + \bar{w}[n], \end{aligned} \quad (6.10)$$

where $n = 1, 2, \dots, N_{\text{FFT}} + L_{\text{CP}}$. The differential phase of channel taps can be estimated as,

$$\phi[n] = \frac{h[n+1]}{h[n]} = e^{j2\pi\epsilon n} + \tilde{w}[n] \quad (6.11)$$

$\tilde{w}[n]$ denotes the noise component. The equation 6.15 assumes that the channel is changing smoothly across time. The CFO for time sample n can be estimate as,

$$\hat{\epsilon}(n) = \frac{\angle \phi[n]}{2\pi} \quad (6.12)$$

In order to mitigate the effect of noise, the mean of all $N_{\text{FFT}} + L_{\text{CP}} - 1$ samples are taken. The resulting maximum likelihood CFO estimation is given as,

$$\hat{\epsilon} = \frac{1}{N_{\text{FFT}} + L_{\text{CP}} - 1} \sum_{n=1}^{N_{\text{FFT}} + L_{\text{CP}} - 1} \hat{\epsilon}[n] \quad (6.13)$$

Algorithm: The Pilot based CFO Estimation Method

- 1.** Extract all the $N_{\text{FFT}} + L_{\text{CP}}$ received samples of OFDM symbol: $\{y[n]\}$
- 2.** $x[n]$ = Extract all the $N_{\text{FFT}} + L_{\text{CP}}$ transmitted samples of OFDM symbol.
- 3.** Estimate the channel taps $h[n] = \frac{y[n]}{x[n]}$

4. Compute differential phase for time instant n ,

$$\phi[n] = \frac{h[n+1]}{h[n]} = e^{j2\pi\epsilon n} + \tilde{w}[n] \quad (6.14)$$

5. CFO estimate at time instant n ,

$$\hat{\epsilon}[n] = \frac{\angle \phi[n]}{2\pi} \quad (6.15)$$

6. Maximum Likelihood (ML) estimation of CFO is given by,

$$\hat{\epsilon} = \frac{1}{N_{\text{FFT}} + L_{\text{CP}} - 1} \sum_{n=1}^{N_{\text{FFT}} + L_{\text{CP}} - 1} \hat{\epsilon}[n] \quad (6.16)$$

In this subsection, we discussed the CP-based and pilot-based CFO estimation techniques in detail. In the upcoming results section, we will analyze the performance of both CFO estimation techniques and compare them.

6.3 | Results

The hardware emulations are conducted using the simulation parameters specified in table-6.1. The evaluations are performed under various link conditions, including line of sight (LoS) or non-line of sight (NLoS), and at different distances between the transmitter and receiver.

Table 6.1: Simulation parameters and evaluation methodology

Parameter	Value
Carrier frequency (f_c)	1000 MHz
Bandwidth (B)	5 MHz
FFT size (N_{FFT})	1024
Subcarrier spacing (Δf)	15 KHz
Transmitter-Receiver separation	10 cm, 1 m, 5m, 10m
Physical Channel	PBCH
Channel Estimation	PBCH-DMRS
Channel Estimation/Interpolation Method	Least Squares with linear interpolation
Symbol Equalization Method	Zero forcing
Time Synchronization	PSS based time correlation
CFO Estimation	CP-based

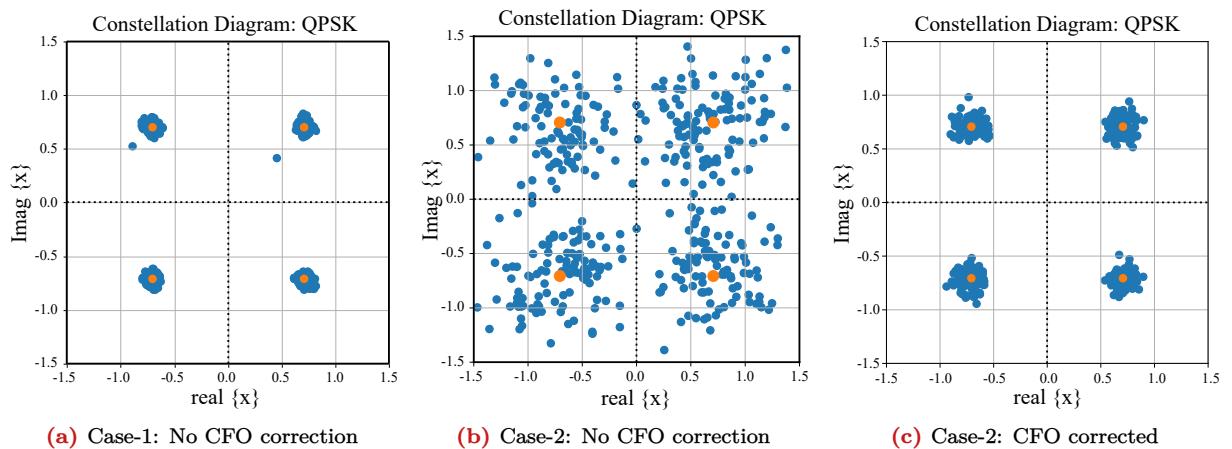


Figure 6.4: Constellation diagrams of received QPSK symbols for case-1 and case-2. | Case-1: Transmitter and receiver are on same SDR | Case-2: Transmitter and Receiver are on two different SDRs.

Observation 1: When the same SDR is used as both the transmitter (BS) and the receiver (UE), no CFO is observed because both the transmitter and receiver use the same local oscillator.

It can be clearly observed from Figure 6.4a that when the same SDR is employed for transmission and reception, there is no effect of CFO. When the same SDR is used at both the transmitter and receiver, the frequency of the local oscillator and the clocks used by the digital-to-analog converter and analog-to-digital converter are exactly the same. This results in nearly perfect time and frequency synchronization between the transmitter and receiver, resulting in very clean constellation of the received symbols.

Observation 2: When two different SDRs using two different clock sources are used for decoding data, the carrier frequency offset (CFO) becomes non-negligible.

It is practically very difficult to manufacture identical devices or components, such as clock frequencies and local oscillators. This results in mismatches in the clock rate and carrier frequency of the clock source and local oscillator between the transmitter and the receiver. These hardware impairments can severely corrupt the quality of the received symbols, as shown in Fig-6.4b. The effect of this carrier frequency offset (CFO) is low when the pilot density is high, which is the case for PBCH-SSB. However, CFO can significantly degrade the quality of the constellation for PDSCH as shown in Fig-11.16, where the density of pilots across time is low. This impact can be even more severe for higher constellation orders.

Observation 3: Quality of the constellation of the received equalized symbols improves drastically when a CFO correction algorithms is applied.

The effect of the CFO can be reduced using CFO estimation algorithm as shown in Fig[6.4c].

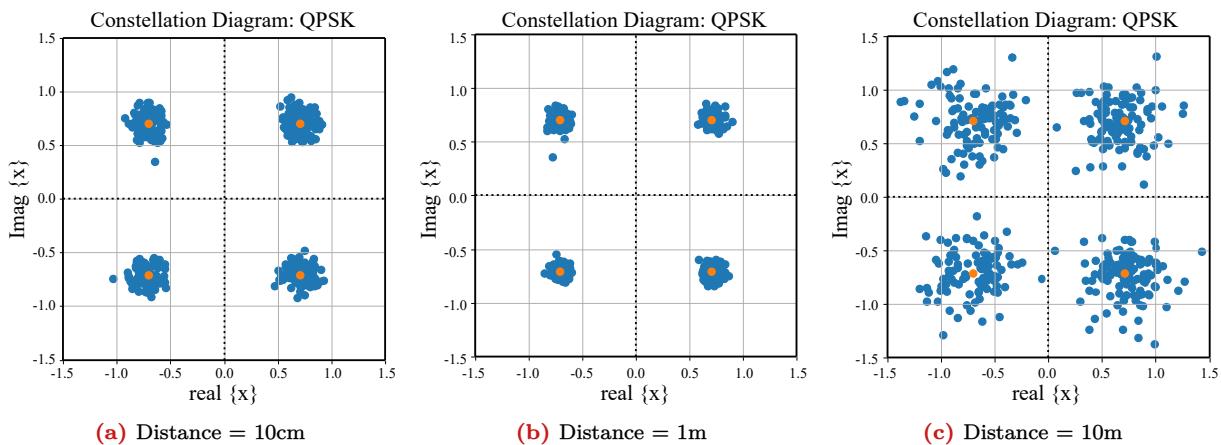


Figure 6.5: Quality of QPSK constellation after CFO correction for different Tx-Rx (LoS link) separation.

Observation 4: The performance of the CFO estimation algorithms changes significantly with channel conditions (LoS/NLoS links) and the transmitter and receiver separation.

CFO algorithms are designed with certain assumptions, such as CP-based CFO estimation algorithms is optimal in flat-faded channel conditions. The performance of these CFO algorithms changes with channel conditions, as illustrated in Fig-6.5 and Fig-6.6. As the separation between the transmitter and receiver increases, the signal-to-noise ratio (SNR) degrades. Additionally, in general, for non-line-of-sight (NLoS) links, the frequency selectivity in the channel increases with distance. Both of these factors contribute to the degradation in performance of the employed CFO estimation algorithm.

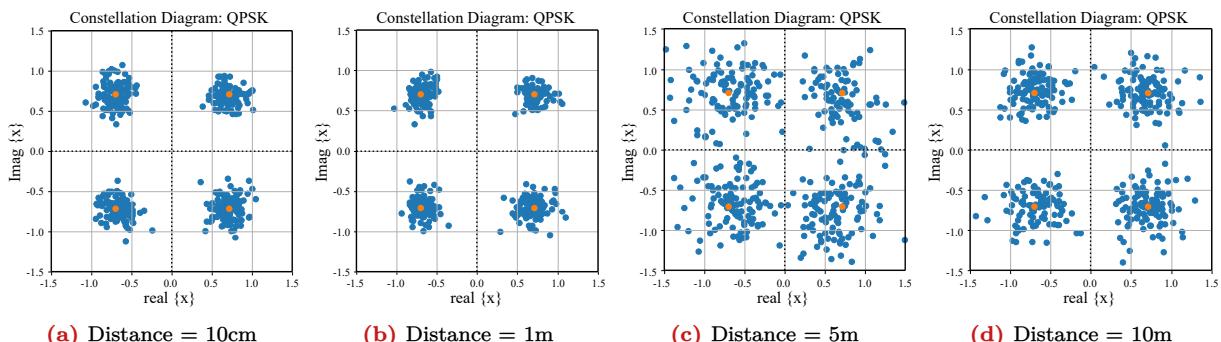


Figure 6.6: Quality of QPSK constellation after CFO correction for different Tx-Rx(NLoS link) separation.

An important metric that determines the quality of equalized symbol constellation is Error Vector Magnitude (EVM). EVM computation measures the spread of received symbols with respect to transmitted constellation symbols. The lower the EVM, the better the performance of the CFO estimation method. The estimated CFO and EVM performance of the CFO estimation method for different channel conditions are listed in table-6.2.

Error Vector Magnitude (EVM): The EVM computes the euclidean distance between each received samples with nearest symbol(in the transmitted constellation). The EVM gives the measure of how well the CFO algorithm works. Ideally, the EVM must be 0(typically when using single SDR as transmitter and receiver), indicating the all received lies on the point as transmitted samples. The procedure to compute the EVM is detailed below:

- Compute the distance of all received symbols from constellation points of modulation.
- For each received symbol consider the minimum distance.
- Compute mean of the minimum distances of all received symbols from previous step to get EVM.

Table 6.2: CFO Estimation Performance for different channel conditions

Tx-Rx Separation (m)	Link state (LoS/NLoS)	Estimated CFO (ϵ)	EVM
0	Same SDR (LoS)	0.013	0.045
0.1	Different SDR (LoS)	-1.233	0.129
1	Different SDR (LoS)	-1.244	0.143
10	Different SDR (LoS)	-1.1841	0.0789
50	Different SDR (LoS)	-0.697	0.605
0.1	Different SDR (NLoS)	-1.132	0.0424
1	Different SDR (NLoS)	-1.254	0.117
10	Different SDR (NLoS)	-1.180	0.2509
50	Different SDR (NLoS)	-1.105	0.627

Table 6.2 shows the observation table, students must fill the values of estimated CFO and EVM for each distance between SDRs. It can be noted from results that as distance between the two SDRs increases the EVM increases. The above experiment was performed for 0 dB transmitter gain and 50 dB receiver gain.

6.4 | Useful links

- Carrier frequency synchronization in the downlink
- CFO Estimation and Correction in 5G Toolkit.
- Video tutorial of this chapter.

6.5 | Exercise

1. For the diagram given in Fig-6.7, compute the EVM,
2. Estimate the CFO using the OFDM symbols based on CP based CFO estimation.

- The received samples and the transmitted samples can downloaded from the [link](#).
- $N_{FFT} = 1024$
- $\Delta f = 15 \text{ kHz}$
- $L_{CP} = 72$
- Assumption: Channel is flat-faded.

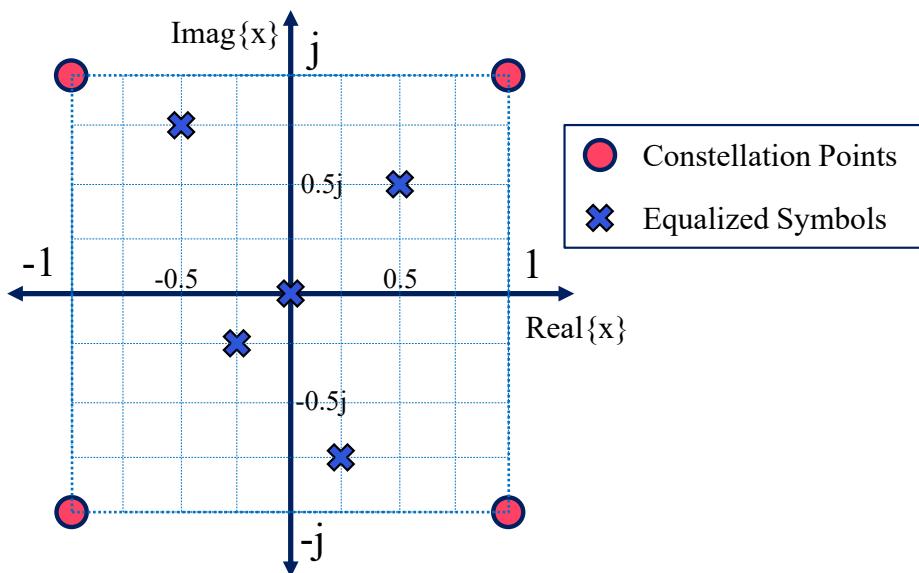


Figure 6.7: Transmitted vs received Symbols

3. Estimate the CFO using the OFDM symbols based on Pilot based CFO estimation.

- Assume the same data as in previous example.

7 | Channel Estimation and Equalization for SSB/PBCH using PBCH-DMRS

The objective of the wireless communication system is to decode the transmitted data at the receiver. Before decoding the data, the receiver must estimate the channel between the transmitter and itself. However, wireless channels are dynamic and subject to various environmental factors such as multipath fading, shadowing, and interference. By accurately estimating the wireless channel, the receiver can decode the transmitted data without any error. Error-free decoding of data results in higher throughput and lower BLER, thus making accurate channel estimation in wireless communication crucial. In this experiment, we will discuss different techniques for channel estimation and use this estimated channel to decode the SSB.

7.1 | Why Estimate Wireless Channel?

As radio signal propagate from transmitter to receiver, the signal propagate through the multi paths to reach the receiver. The signal received at the k^{th} subcarrier is given by,

$$Y[k] = H[k]X[k] + N[k], \quad (7.1)$$

where $Y[k]$, $H[k]$, $X[k]$ and $N[k]$ are the received signal, channel between transmitter and receiver, transmitted signal and noise at the receiver respectively. As discussed, before decoding the data, the receiver must obtain the accurate estimate of wireless channel. The channel estimation is performed using the pilot sequences, which are known both at the transmitter and receiver. These pilots are called Demodulation Reference Signal (DMRS) in 5G. As discussed in 4, OFDM modulation is performed on 2D time-frequency grid as shown in figure 7.1 carrying DMRS and payload. The resources allocated to DMRS and PBCH symbols are demonstrated in figure 7.1.

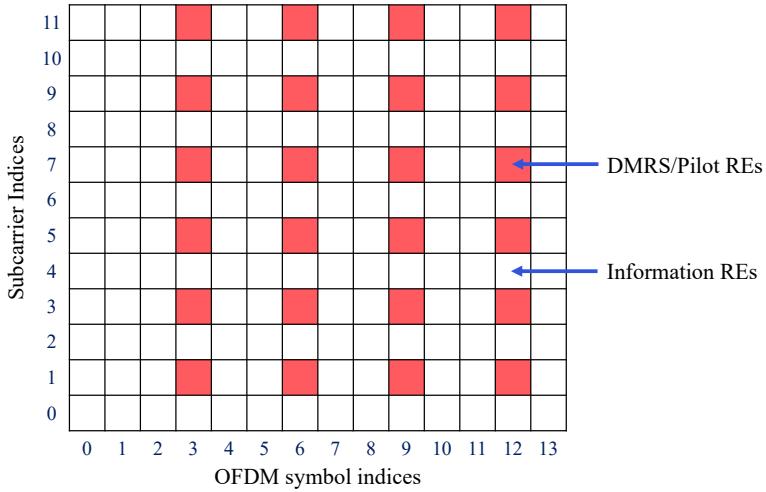


Figure 7.1: 5G Resource Grid

7.2 | Channel Estimation Techniques

The two most commonly used channel estimation technique in wireless communication are,

1. least square (LS) based channel estimation
2. minimum mean square error (MMSE) based channel estimation

The channel estimation is performed in two steps, as explained below.

1. The channel is estimated at pilot locations using known DMRS sequences.
2. Interpolate the channel at data locations using the channel estimated at DMRS locations. Different interpolation schemes can be used based on the rate of change in channel across time and frequency. [5G Toolkit](#) supports nearest neighbour (NN), linear, cubic and spline interpolation. The interpolation techniques are discussed in detail in section 7.3.

7.2.1 | Least Square Based Channel Estimation

The LS estimation framework tries to minimize the error between the estimated channel and the actual channel based on first order approximation. The generalized LS channel estimate is given by,

$$\hat{H}[k] = Y[k](X[k]^H X[k])^{-1} X[k]^H, \quad (7.2)$$

where $\hat{H}[k]$ is the estimated channel matrix, $Y[k]$ is the received matrix and $X[k]$ is the transmitted pilot matrix. For a single layer transmission equation 7.2 reduces to,

$$\hat{H}[k] = Y[k]/X[k] \quad (7.3)$$

7.2.2 | Minimum Mean Square Error(MMSE) Channel Estimation

The MMSE estimator provides an estimate of the channel matrix $H[k]$ that minimizes the mean square error (MSE) between the estimated channel and the true channel. Mathematically, the MMSE estimator is given by,

$$\hat{H}_{MMSE} = R_{h,y} R_y, \quad (7.4)$$

Where \hat{H}_{MMSE} is the MMSE estimate of the channel matrix, $R_{h,y}$ is the cross-correlation matrix between the true channel and the received signal, R_y is the autocorrelation matrix of the received signal. $R_{h,y}$ and R_y is estimated from DMRS.

The choice between LS and MMSE technique depends on following tradeoffs:

- 1. Performance:** The MMSE estimator minimizes the mean square error between the estimated channel and the true channel. It takes into account the statistics of the noise and the channel, resulting in potentially better performance, especially in scenarios where there's significant noise or fading. Whereas, the LS estimator minimizes the sum of squared errors between the estimated and observed signals without considering the noise statistics. It is simpler and computationally less intensive but may suffer from higher error rates, particularly in noisy or frequency-selective fading channels.
- 2. Robustness to Noise:** MMSE estimator considers noise statistics, it tends to be more robust in noisy environments. It can mitigate the effects of noise and provide better estimation accuracy. While LS estimator may be more sensitive to noise as it doesn't explicitly account for noise characteristics during estimation. MMSE is more robust to ill-conditioned channel matrix.
- 3. Computational Complexity:** Generally, MMSE estimation involves more complex calculations, including matrix inversions and multiplications, making it computationally more expensive compared to LS estimation. While LS estimation involves simple matrix operations and is computationally less expensive compared to MMSE estimation.

7.3 | Channel Interpolation Techniques

The 5G Toolkit employs four types of channel interpolators to accommodate different channel scenarios. The channel interpolators are listed below,

- 1. Nearest Neighbour (NN):** It is the simplest channel interpolator with the least computational complexity. Once the channel is estimated at pilot locations, applying the nearest neighbour interpolator results in the channel estimate at data locations being equal to the channel estimated at the nearest pilot. NN works best for flat faded channels and lower subcarrier spacing. An example of NN interpolator is shown in figure 7.2a below. As observed from figure 7.2a, the NN interpolator estimated channel is not able to keep track of actual channel accurately.
- 2. Linear:** Linear interpolation is a method of curve fitting using linear polynomials to construct new data points within the range of a discrete set of known data points. The computational complexity of linear interpolator is higher than that of NN interpolator. Linear interpolation works well for moderately varying channels and for lower subcarrier spacing. The magnitude plot of linear interpolator is shown in figure 7.2b below, the linear interpolated estimated channel is able to track the actual channel for most part except for some subcarriers.

3. Cubic: The cubic interpolator estimates the values between known data points by fitting a cubic polynomial curve through those points. The computational complexity of the cubic interpolator is higher than that of NN and linear interpolator. However, it can track rapidly varying channels and works satisfactorily for higher subcarrier spacing.
4. Spline: The spline interpolator involves fitting a piecewise polynomial function to the data points in such a way that the resulting curve passes through each data point smoothly. This smoothness is typically achieved by enforcing continuity of derivatives up to a certain order across adjacent polynomial segments. There are different types of spline interpolators, with cubic splines being one of the most common. Cubic splines use cubic polynomials between each pair of adjacent data points and ensure continuity of the first and second derivatives at each data point. This creates a smooth curve that passes through all the given data points without oscillations. The spline interpolator are best interpolators which can track rapidly varying channel and works best for higher subcarrier spacing. The magnitude plot of cubic spline interpolator is shown in figure 7.2c below. As observed, the cubic spline interpolated channel gives the best performance as it is able to track the channel of all the subcarriers. It must be noted that the cubic spline interpolator shown in figure 7.2c is equivalent to cubic interpolator. In general, the spline interpolators can be of higher order, however, it is observed that the the channel estimation performance of third order spline, a.k.a cubic spline is best and saturates for higher order splines.

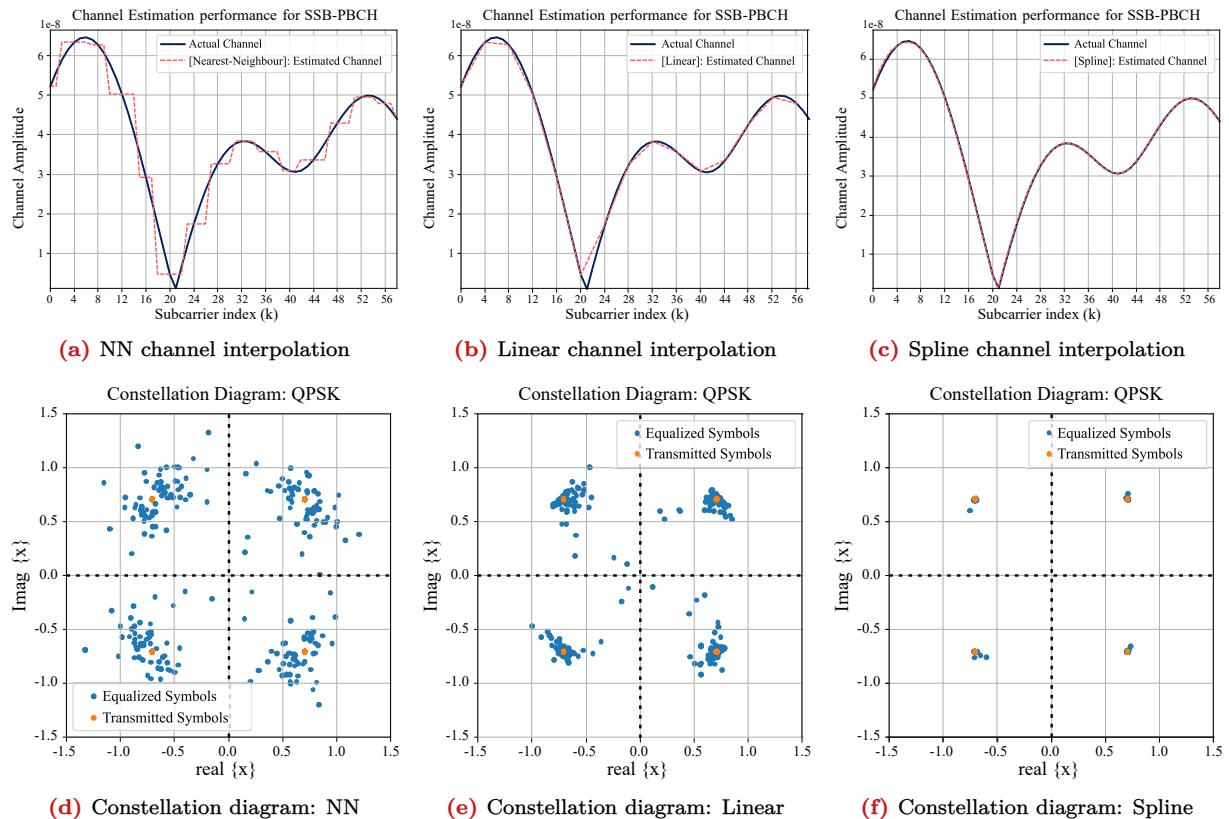


Figure 7.2: Performance evaluation of different channel interpolation schemes for $\Delta f = 15$ kHz.

7.4 | What is Symbol Equalization?

Symbol equalization or channel equalization refers to a signal processing technique used to mitigate the effects of channel impairments on transmitted symbols. In wireless communication systems, signals transmitted from a transmitter to a receiver undergo various distortions and impairments due to factors such as multipath fading, shadowing, interference, etc. These phenomena results in the distortion and corruption in the transmitted symbols. Symbol equalization aims to neutralize these effects and recover the original symbols. By performing symbol equalization, the receiver can improve the accuracy of symbol decoding, thereby enhancing the overall performance of the communication system. This is particularly important in 5G systems, which aim to provide high data rates, low latency, and reliable connectivity in

diverse and challenging deployment scenarios. Using LS based channel equalization, the symbols can be retrieved using the equation below,

$$\hat{X}[k] = (\hat{H}[k]^H \hat{H}[k])^{-1} \hat{H}[k]^H Y[k] \quad (7.5)$$

When receiving only single layer, equation 7.5 can be simplified to:

$$\hat{X}[k] = Y[k]/\hat{H}[k] \quad (7.6)$$

The complete process of channel estimation, interpolation, and channel equalization is illustrated in Figure 7.3.

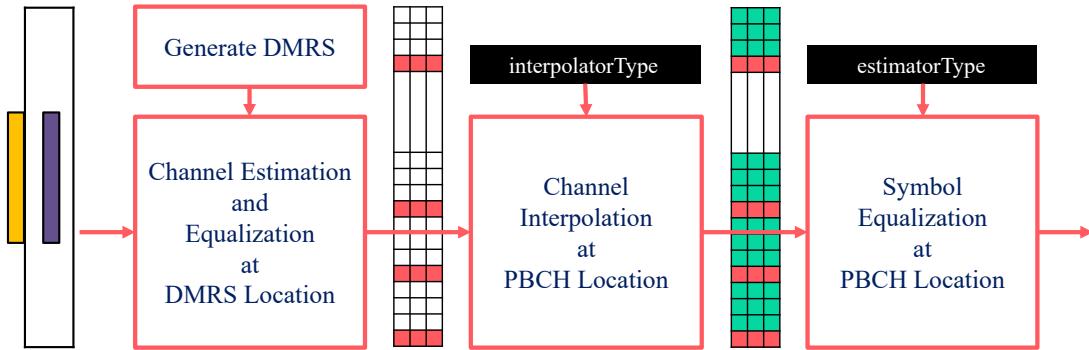


Figure 7.3: Channel estimation and equalization for PBCH.

7.5 | Results

The parameters used for hardware emulation are stated in table-7.1.

Table 7.1: Simulation parameters of PBCH channel estimation and symbol decoding

Parameters	Value
Carrier frequency (f_c)	1000 MHz
Bandwidth (B)	5/10/15 MHz
FFT size (N_{FFT})	1024
subcarrier spacing (Δf)	15/30/60* KHz
Transmitter-Receiver separation (d)	1 m
Channel estimator	zero forcing (ZF)
Channel interpolator	“NN”/“Linear”/“Cubic”

The above parameters hold unless explicitly stated. The tutorial explores channel estimation, channel interpolation, and symbol equalization for the physical broadcast channel (PBCH), which is the simplest and most important channel for communication in 5G networks. PBCH is carried by the synchronization signal block (SSB) in 5G networks, transmitting crucial network information. Further details about this channel will be discussed in the next chapter (Chapter-8).

The process at the receiver begins with time synchronization, followed by carrier frequency offset (CFO) correction. Once synchronization is achieved and the effects of hardware impairments are mitigated, the OFDM resource grid is reconstructed using the OFDM demodulator. From this resource grid, the SSB grid is extracted, which appears as shown in Fig [7.4].

The channel estimation for PBCH requires two components: the demodulation reference signal (DMRS) sequence and the physical cell ID (N_{ID}^{cell}). The cell ID helps the UE identify the time-frequency locations where PBCH-DMRS is loaded for channel estimation. The cell ID is computed using two components: cell-ID-1 (N_{ID}^1) and cell-ID-2 (N_{ID}^2), which are detected using time synchronization (PSS detection) and synchronization signal (SSS) detection procedures respectively, as detailed in the Python code. The channel estimation is performed using DMRS, which is generated using the SSB index, half-frame index, and cell ID. These parameters are detected/estimated using the DMRS parameter detection module.

Observation-1: Errors in the detection of cell-ID, SSB index, or half-frame index will result in failure to decode the synchronization signal block (SSB) and, ultimately, the master information block (MIB).

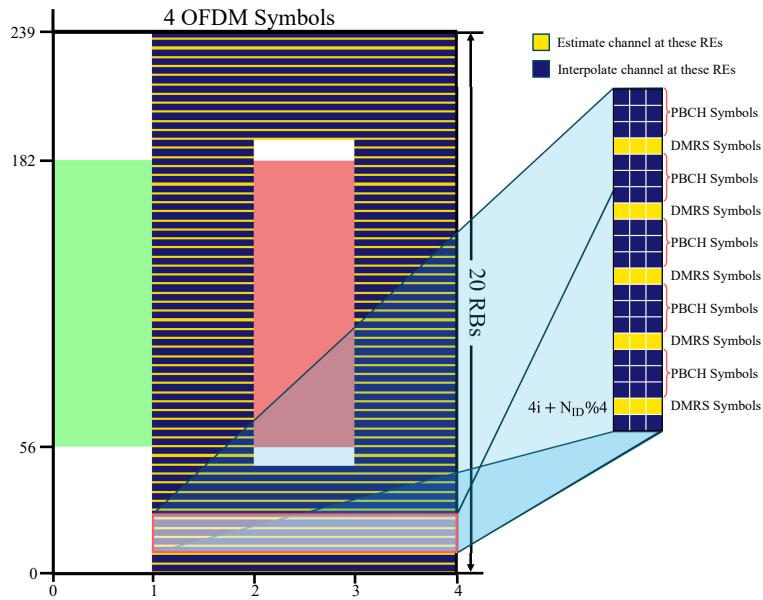


Figure 7.4: 5G Synchronization Signal Block (SSB).

The SSB, as shown in Fig-7.4, is designed very high DMRS density in time and frequency which provides robustness against time and frequency selective channels. As the distance between the transmitter and receiver increase, the link budget deteriorates and frequency selectivity increases. However, it can be seen from the results in Fig-7.7q, that SSB can be detected even in harsh link budget conditions.

Observation-2: *The SSB is robust even in weak links, highly frequency-selective channels, and with high-mobility users.*

As the distance between the transmitter and receiver increases, the received power by the UE decreases, leading to significantly degraded signal-to-noise ratios, as depicted in Fig-[7.7a], [7.7e], [7.7i], [7.7m], [7.7q], and [7.8a]. Furthermore, Fig-[7.7m] and Fig-[7.8a] clearly show that SNRs are much lower in NLoS links compared to LoS links, alongside higher frequency selectivity.

In LoS scenarios where links are typically strong and channels are nearly flat, the nearest neighbor interpolator delivers the best performance along with the least squares estimator. However, the results in Fig-[7.7d], [7.7h], [7.7l], [7.7p], [7.7t], and [7.8d] clearly demonstrate that linear interpolators are the most robust performers for most cases, even in weak frequency-selective channels. However, if the channel is highly frequency selective but the SNR is high, which occurs rarely, the spline interpolator delivers the best performance, as demonstrated by Fig-[7.2].

Observation-3: *The nearest neighbor interpolator is suitable when the channels are mostly flat, but higher-order interpolators such as linear or spline interpolation are required to track the rapid variations in wireless channels resulting from frequency selectivity.*

As the subcarrier spacing increases, the variations in the channel become more abrupt due to higher sample rate and wider frequency bin in frequency domain as shown Fig-[7.2] and Fig-[7.6]. In such scenarios, spline interpolators are the most suitable when the SNRs are high, while linear interpolators are more appropriate when the SNR is low.

Observation-4: *As the subcarrier spacing increases, the channel amplitude varies much more rapidly compared to lower subcarrier spacing, requiring higher-order interpolators for accurate channel estimation and equalization.*

7.6 | Appendix

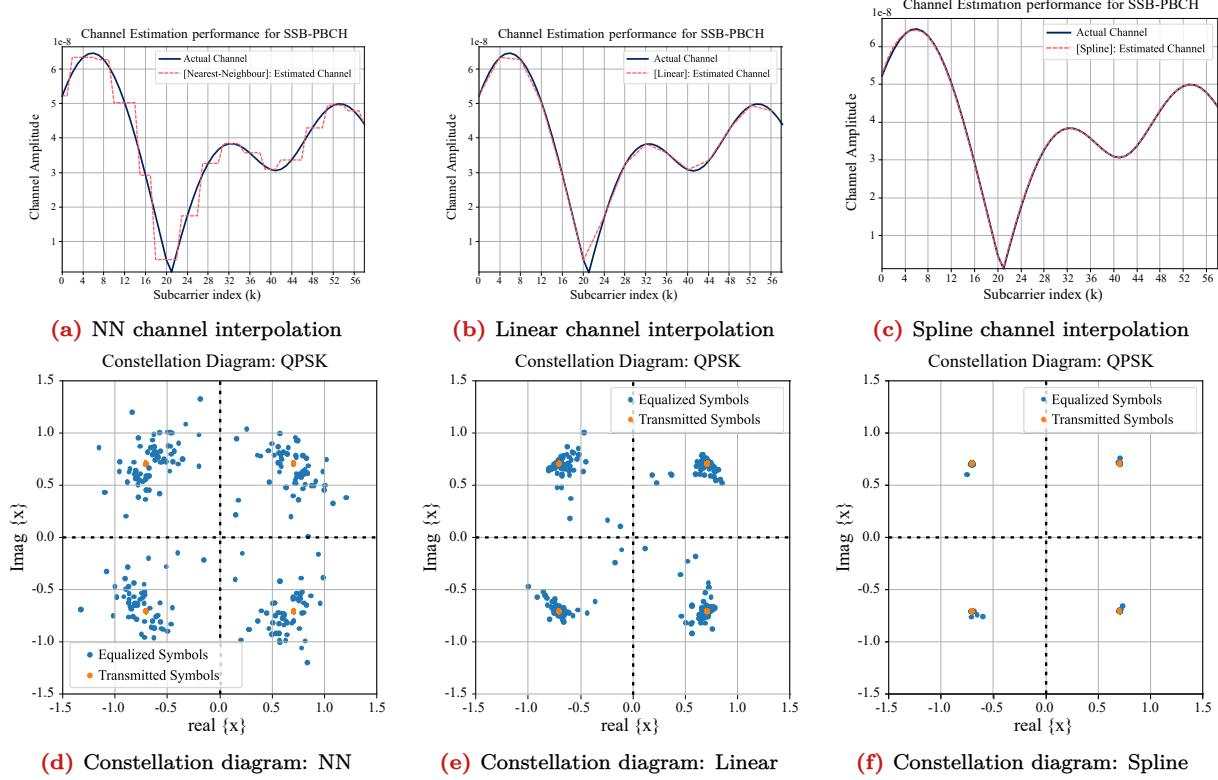


Figure 7.5: Performance evaluation of different channel interpolation schemes for $\Delta f = 30$ kHz.

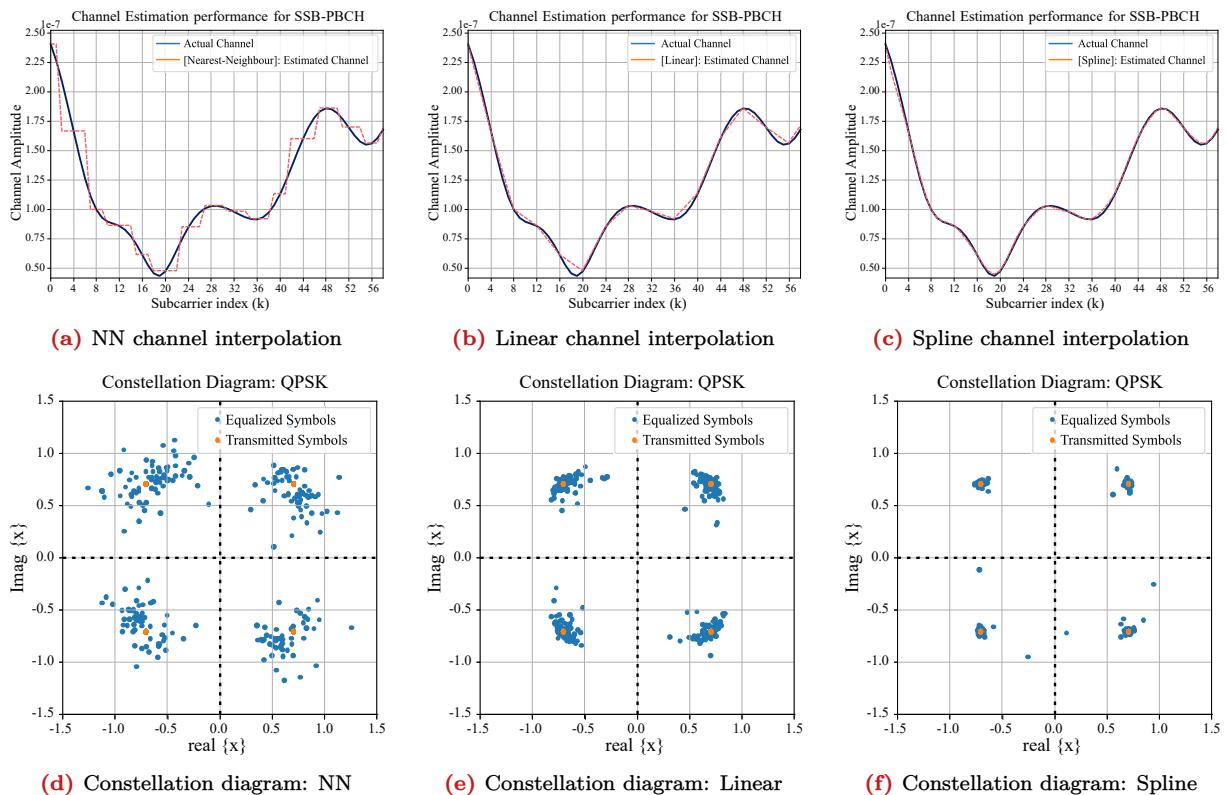


Figure 7.6: Performance evaluation of different channel interpolation schemes for $\Delta f = 60$ kHz.

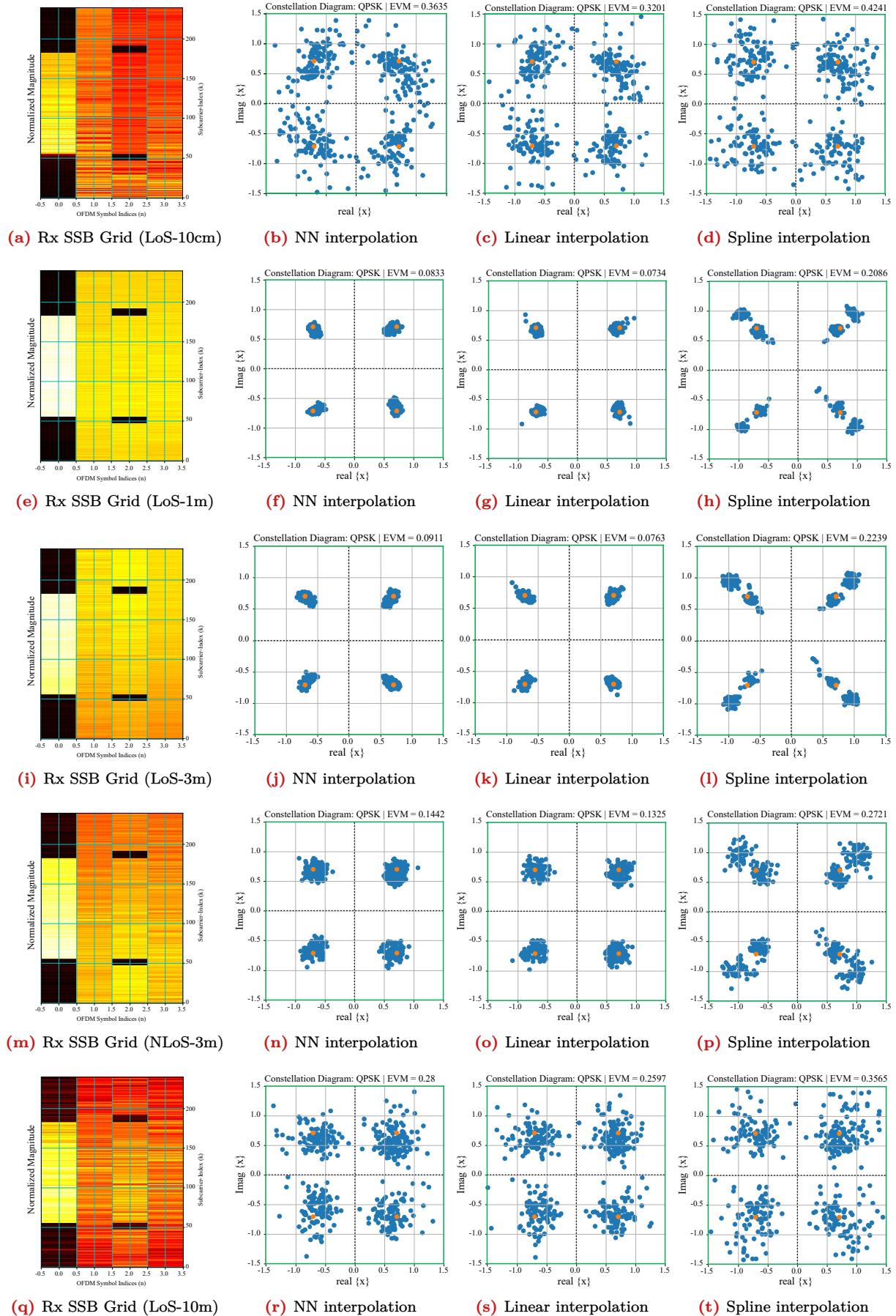


Figure 7.7: Performance evaluation of least square channel estimator combined with nearest neighbour (NN), linear and spline interpolator for different link state (NLoS/LoS) and Tx-Rx separations.

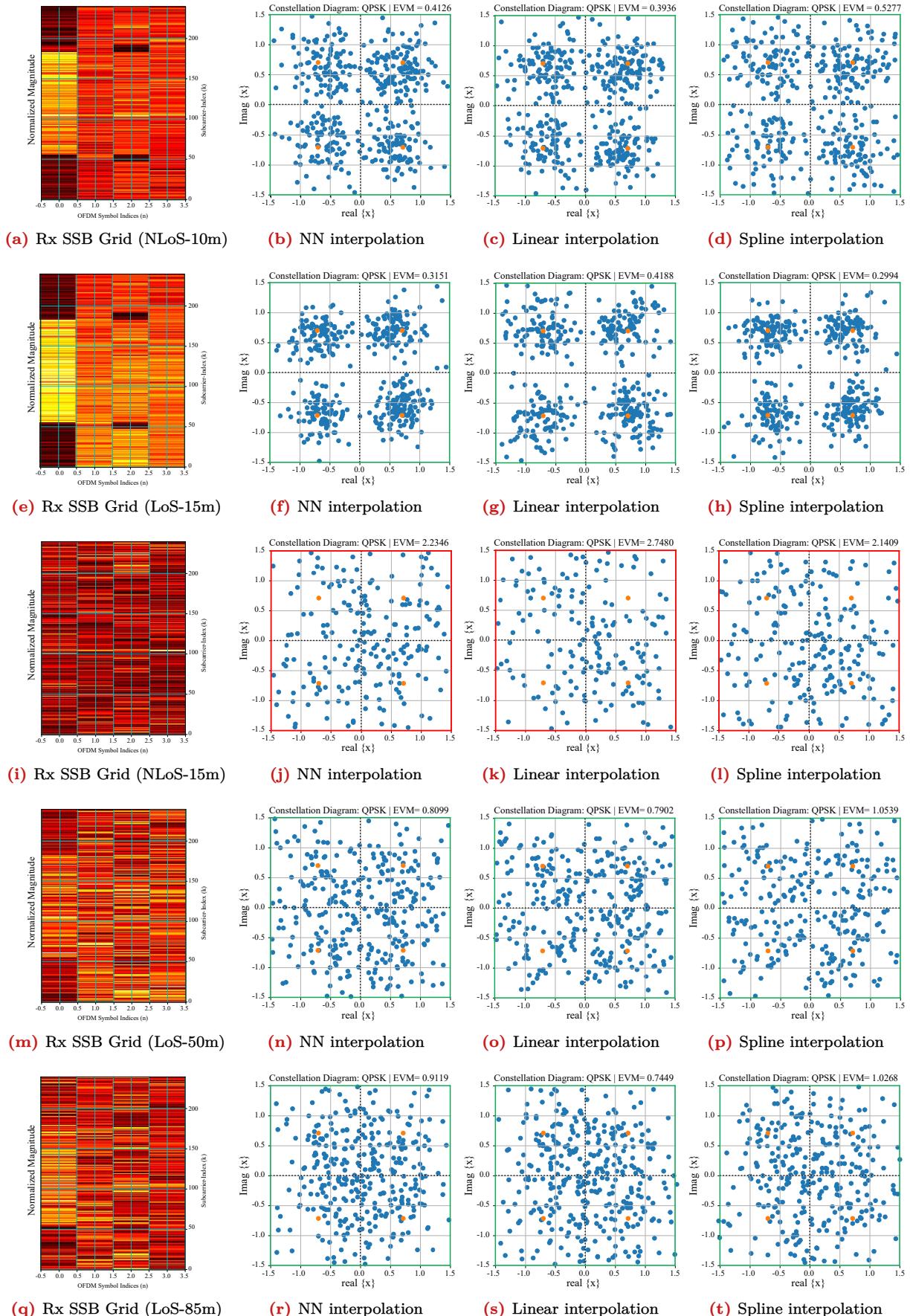


Figure 7.8: Performance evaluation of least square channel estimator combined with nearest neighbour (NN), linear and spline interpolator for different link state (NLoS/LoS) and Tx-Rx separations.

7.7 | Further Reading

- Read more on [Interpolation](#).
- Read about [PBCH](#)
- Read about [PBCH DMRS](#)

8 | Implementation of PBCH Chain in 5G Networks

The Physical Broadcast Channel (PBCH) carries the Master Information Block (MIB). The MIB contains essential information for downlink synchronization of user equipment, such as cell identity, system bandwidth, and configurations of other channels. In this experiment, the complete implementation of the PBCH chain is discussed in detail.

8.1 | What is PBCH Chain?

PBCH chain encompasses a series of steps to prepare, transmit and receive the PBCH information. The PBCH chain consists of: PBCH encoder(processed at base station), PBCH decoder(processed at user equipment).

Table 8.1: Content of master information block (MIB)

-	Parameter	Data-type	Range of Values
MIB	Choice bit	bool	{0, 1}
	Subcarrier spacing common (Δf)	enum	{15_or_120kHz, 30_or_240kHz}
	DMRS-Type-A Position	enum	{DMRS-Type-A, DMRS-Type-B}
	PDCCH-Config-SIB1:CORESET-0	int	{0, 1, ..., 15}
	PDCCH-Config-SIB1:SearchSpace-0	int	{0, 1, ..., 15}
	Cell barred	bool	{0, 1}
	Intra-frequency reselection	bool	{0, 1}
	Spare bit	bool	0
MIB/ATI	System frame number (SFN)	int	{0, 1, ..., 1023}
	SSB-subcarrier offset (k_{SSB})	int	{1, 23} for FR-1 ($f_c < 6\text{GHz}$) {0, 11} for FR-2 ($f_c > 6\text{GHz}$)
ATI	Half radio frame bit (n_{HRF})	bool	{0, 1}
	SSB-PBCH index (L_{SSB})	int	{0, 3} for FR-1 ($f_c < 2\text{GHz}$)
			{0, 7} for FR-1 ($2 < f_c(\text{GHz}) < 6$)
			{0, 63} for FR-2 ($f_c > 6\text{GHz}$)

8.2 | Design of PBCH Chain

8.2.1 | PBCH Chain: Transmitter

The PBCH encoder is shown in figure 8.1 below.

PBCH processing at base station is explained below:

- **PBCH payload generation:** This is the first stage of processing at base-station wherein payload data intended for broadcast is prepared. The size of PBCH payload generation is 32 bits.
- **PBCH Interleaver:** The objective of PBCH interleaver is to achieve time diversity. The interleaver disperses bits to minimize effect of burst error. The size of output of interleaver is 32 bits.
- **1st Scrambling:** The purpose of scrambler is to break long sequence of 0's and 1's. Scrambling is achieved by ex-or-ing the input sequence with scrambling sequence. The length of scrambling sequence and input sequence both are 32 bits, which produces a 32 bits scrambled sequence.
- **CRC addition:** 24 bits of CRC bits are added to the scrambled payload to enable error detection at user-equipment. The size of payload along with CRC bits are 56 bits.

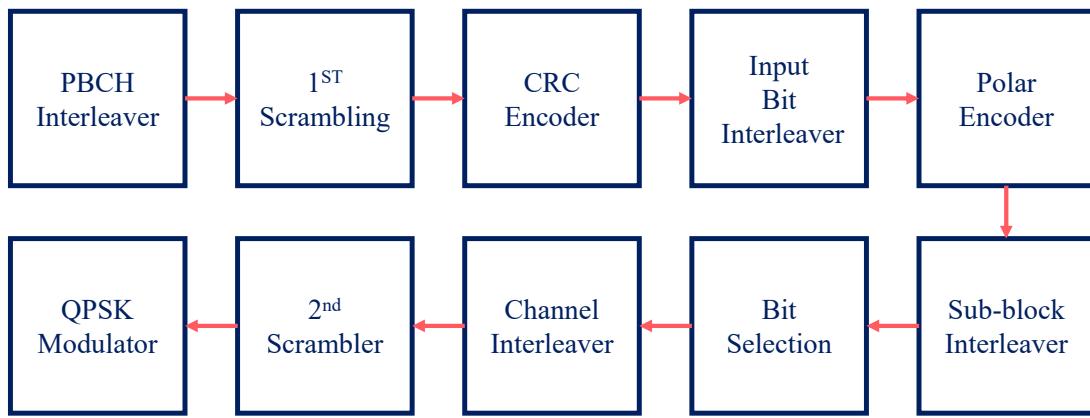


Figure 8.1: PBCH chain: Transmitter

- **Input bit Interleaver:** Bit interleaver is a crucial component used in the physical layer to enhance the reliability of data transmission over the air interface. The purpose of the bit interleaver is to disperse consecutive bits of information across time and frequency domains, thereby mitigating the effects of burst errors and improving the system's ability to recover from transmission impairments like fading, noise, and interference. The size at output of bit interleaver is 56 bits.
- **Polar Encoder:** Polar coding is forward error correction (FEC) technique used to improve the reliability of data transmission over the wireless channel. The polar coded adds the redundant bits to information bits. The output of polar coder has 512 bits.
- **Rate Matching:** Rate matching primarily has two functions, control the code rate (redundancy) and enable hybrid-ARQ. The output size of rate matching is such that it is equal to numbers of available resources. The output size of rate matching is 864 bits.
- **Scrambling:** Scrambling breaks the long sequence of consecutive 1's or/and 0's. The output of rate matching is ex-ored with the scrambling sequence of same length. The output size remains 864 bits.
- **QPSK modulator:** Upon QPSK modulation, which has modulation order of 2, the 864 bits converts into 432 symbols.
- **DMRS generator and resource mapping:** The PBCH DMRS sequence are generated and mapped on the resource grid.

8.2.2 | PBCH Chain: Receiver

The PBCH decoder is shown in figure 8.2 below.

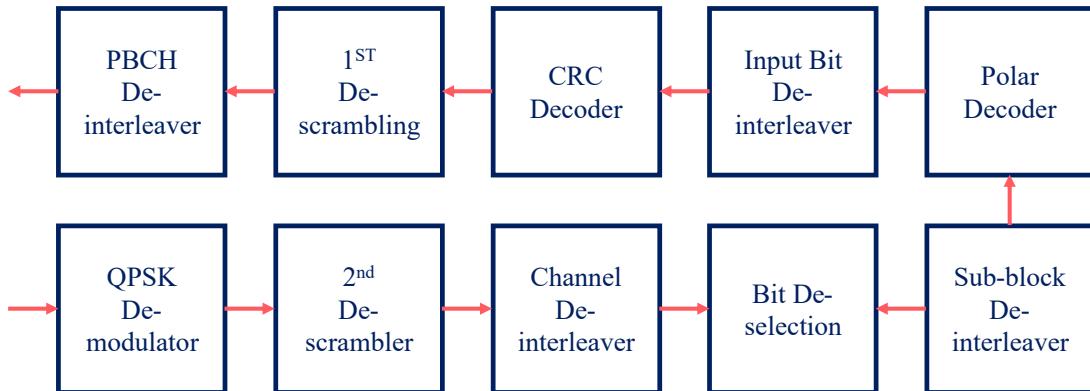


Figure 8.2: PBCH chain: Receiver

PBCH processing at user equipment is explained below:

- **Estimate SSB grid:** Upon receiving samples from base-station SDR, time synchronization is performed by detecting PSS, followed by CFO estimation and OFDM demodulation. Upon OFDM demodulation the SSB grid is extracted.
- **Channel Estimation:** Upon extracting the SSB, the channel across PBCH is estimated using PBCH DMRS. The **ChannelEstimationAndEqualization** class has two parameters 'estimatorType' and 'interpolatorType'. 'estimatorType' specifies the channel estimation technique used, for SSB only Zero-Forcing (ZF) is employed. 'interpolatorType' specifies channel interpolation type used, for SSB 'nearest neighbour' interpolator is used. Channel is estimated using relation:

$$\hat{H} = Y/X \quad (8.1)$$

- **Equalization:** Transmitted symbols is retrieved using:

$$\hat{X} = Y/\hat{H} \quad (8.2)$$

Equalization return 432 QPSK modulated symbols.

- **QPSK De-modulator:** The QPSK demodulator is used to demodulate the PBCH and extract its content the MIB information. The demodulation converts 432 QPSK modulated symbols into 864 bits.
- **De-Scrambling:** The demodulated bits are ex-ored with the scrambling sequence. This step is done to nullify the effect of scrambling. The size of de-scrambling output remains 864 bits.
- **Rate De-matching:** The rate de-matching reverses the effect of rate matcher. The size of de-matching output is 512 bits.
- **Polar Decoder:** Polar decoder removes the redundant bits added at the base station. The redundant bits added at base station are removed to obtain 56 bits at the output of polar decoder.
- **Input bit de-interleaver:** The bit de-interleaver is the counterpart of a bit interleaver. While a bit interleaver rearranges bits in a specific pattern before transmission to mitigate burst errors, a bit de-interleaver is used at the receiving end to revert the rearranged bits back to their original order. The size of output remains 56 bits.
- **CRC Decoder:** At this step, the CRC bits of 24 bits is checked for any presence of error. These 24 bits CRC bits are removed to obtain 32 bits at the output.
- **1st De-scrambling:** The 32 bits from CRC decoder is ex-ored with scrambling sequence of 32 bits to reverse the effect of 1st scrambling at the base station.
- **PBCH De-interleaver:** Another de-interleaving operation is performed. The output size remains 32 bits.
- **Decoded PBCH payload:** Finally, the PBCH payload of 32 bits is obtained.

8.3 | Results

The general simulation parameters are given below,

Table 8.2: General simulation parameters.

Name of parameter	Value
center frequency/carrier frequency	1000 MHz
Bandwidth	5 MHz
FFT size	1024
subcarrier spacing	15 KHz
Transmitter-receiver separation	1 m

The above parameters holds for every results unless otherwise specified.

Observation:1- To verify whether Master Information Block(MIB) parameters at transmitter and receiver are same.

The PBCH contains Master Information Block(MIB), which contains necessary information (*SystemFrameNumber*, *subCarrierSpacingCommon*, *ssb-SubCarrierOffset*, etc.) for initial cell selection procedure. Firstly, UE needs to successfully detect the SSB, detect and decode PBCH. The contents of MIB should be matched at both transmitter(base-station) and receiver(user-equipment) end to ensure successful receipt of MIB.

The MIB parameters at transmitter(base station) is shown in figure 8.3 below.

```
pbchObject.mib.displayParameters(0)
Carrier Frequency: 1000000000.0
ChoiceBit: 1
nSSBCandidatesInHrf: 4
subCarrierSpacingCommon:30000
DMRSTypeAPosition: typeB
controlResourceSet0: 5
searchSpace0: 13
cellBarred: notBarred
intraFreqReselection: notAllowed
systemFrameNumber: 315
ssbSubCarrierOffset: 0
HRFBit: 1
iSSBindex: 0
```

Figure 8.3: MIB parameters: Transmitter

After successful detection of SSB, detection and decoding of PBCH, the MIB is extracted as shown in figure 8.4.

```
pbchDecoder.mibRx.displayParameters(0)
Carrier Frequency: 1000000000.0
ChoiceBit: 1
nSSBCandidatesInHrf: 4
subCarrierSpacingCommon:30000
DMRSTypeAPosition: typeB
controlResourceSet0: 5
searchSpace0: 13
cellBarred: notBarred
intraFreqReselection: notAllowed
systemFrameNumber: 315
ssbSubCarrierOffset: 0
HRFBit: 1
iSSBindex: 0
```

Figure 8.4: MIB parameters: Receiver

Observation:2- The distance between transmitter and receiver are dynamic in nature. It becomes important to determine the impact of distance between the transmitter and receiver SDRs on detection and decoding of PBCH (which contains the MIB information).

Table 8.3: MIB decoding at UE

Transmitter-Receiver distance	Whether MIB information is decoded correctly at UE
10 cm	Yes
50 cm	Yes
1 m	Yes

From table 8.3, it is clearly observed that for distances 10 cm, 50 cm and 1 m the MIB information is decoded correctly at UE.

9 | Uplink Synchronization using physical Random Channel (PRACH) in 5G Networks

A curated set of project topics is provided, requiring foundational knowledge of communication systems and wireless communications acquired through undergraduate-level coursework. This non-exhaustive list aligns with the technical areas explored in prior chapters of this manual.

9.1 | Why Uplink Synchronization?

A curated set of project topics is provided, requiring foundational knowledge of communication systems and wireless communications acquired through undergraduate-level coursework. This non-exhaustive list aligns with the technical areas explored in prior chapters of this manual.

9.2 | What is PRACH?

A curated set of project topics is provided, requiring foundational knowledge of communication systems and wireless communications acquired through undergraduate-level coursework. This non-exhaustive list aligns with the technical areas explored in prior chapters of this manual.

1. Waveform denoising for **AM/PM/FM**.
2. Modulation order classification for **ASK/PSK/QAM** using machine learning.
3. Low complexity symbol demapping based on artificial neural networks in **IoT** devices 5G networks.
4. Machine learning based low complexity decoder for linear block codes.
5. Deep learning based low complexity decoder for linear cyclic codes.
6. Recurrent neural networks based low complexity decoder for convolutional codes.
7. AI based adaptive modulations and coding.
8. AI based adaptive quantization and reconstruction for variable dynamic range signal.
9. Learning site specific modulations/constellation for communication.
10. Optimizing the raised cosine waveform filter parameters to improve the **PAPR** and **ACLR**.

9.3 | Design of PRACH Chain

A curated set of project topics is provided, requiring foundational knowledge of communication systems and wireless communications acquired through undergraduate-level coursework. This non-exhaustive list aligns with the technical areas explored in prior chapters of this manual.

1. Waveform denoising for **AM/PM/FM**.
2. Modulation order classification for **ASK/PSK/QAM** using machine learning.
3. Low complexity symbol demapping based on artificial neural networks in **IoT** devices 5G networks.
4. Machine learning based low complexity decoder for linear block codes.
5. Deep learning based low complexity decoder for linear cyclic codes.
6. Recurrent neural networks based low complexity decoder for convolutional codes.
7. AI based adaptive modulations and coding.
8. AI based adaptive quantization and reconstruction for variable dynamic range signal.
9. Learning site specific modulations/constellation for communication.
10. Optimizing the raised cosine waveform filter parameters to improve the **PAPR** and **ACLR**.

9.3.1 | PRACH Chain: Transmitter

This curated list presents diverse student projects aligned with the **Introduction to 5G Standards** (see section 9.3.1) course. Each project focuses on a specific aspect of the 5G air interface, encompassing key stages like algorithmic thinking, link-level simulations, and physical layer implementation. Students can explore beamforming techniques for enhanced wireless coverage, delve into network synchronization in uplink and downlink, or analyze the performance of NR numerology configurations under realistic channel conditions. These projects provide hands-on experience with a spectrum of 5G technologies, equipping students with a comprehensive understanding of the end-to-end procedures within 5G networks.

1. Downlink synchronization in **5G** networks.
2. Uplink synchronization in **5G** networks.
3. SSB based beam-management (**P1** procedure) in **5G** networks.
4. Control resource set (**CORESET**) and search space set in 5G networks.
5. Physical downlink channel (**PDCCH**) blind decoding (**CORESET**) in 5G networks.
6. Hybrid automatic request (**HARQ**) in 5G networks.
7. Reciprocity based uplink and downlink beam-forming in 5G **TDD** networks.
8. **Type-I** and **Type-II** based downlink precoding in 5G **TDD** networks.
9. Beam-refinement (P2 procedure) using CSI-RS in 5G networks.
10. Channel quality indicator(**CQI**) estimation for link adaptation in 5G networks.
11. Rank adaptation for downlink **massive MIMO** in 5G networks.
12. Downlink and uplink power control for massive MIMO in 5G networks.
13. Downlink scheduling and resource allocation in 5G networks.

9.3.2 | PRACH Chain: Receiver

A curated set of project topics is provided, requiring foundational knowledge of communication systems and wireless communications acquired through undergraduate-level coursework. This non-exhaustive list aligns with the technical areas explored in prior chapters of this manual.

1. Waveform denoising for **AM/PM/FM**.
2. Modulation order classification for **ASK/PSK/QAM** using machine learning.
3. Low complexity symbol demapping based on artificial neural networks in **IoT** devices 5G networks.
4. Machine learning based low complexity decoder for linear block codes.
5. Deep learning based low complexity decoder for linear cyclic codes.
6. Recurrent neural networks based low complexity decoder for convolutional codes.
7. AI based adaptive modulations and coding.
8. AI based adaptive quantization and reconstruction for variable dynamic range signal.
9. Learning site specific modulations/constellation for communication.
10. Optimizing the raised cosine waveform filter parameters to improve the **PAPR** and **ACLR**.

9.4 | Results

This curated list presents diverse student projects aligned with the **Introduction to 5G Standards** (see section 9.3.1) course. Each project focuses on a specific aspect of the 5G air interface, encompassing key stages like algorithmic thinking, link-level simulations, and physical layer implementation. Students can explore beamforming techniques for enhanced wireless coverage, delve into network synchronization in uplink and downlink, or analyze the performance of NR numerology configurations under realistic channel conditions. These projects provide hands-on experience with a spectrum of 5G technologies, equipping students with a comprehensive understanding of the end-to-end procedures within 5G networks.

1. Downlink synchronization in **5G** networks.
2. Uplink synchronization in **5G** networks.
3. SSB based beam-management (**P1** procedure) in **5G** networks.
4. Control resource set (**CORESET**) and search space set in 5G networks.
5. Physical downlink channel (**PDCCH**) blind decoding (**CORESET**) in 5G networks.
6. Hybrid automatic request (**HARQ**) in 5G networks.
7. Reciprocity based uplink and downlink beam-forming in 5G **TDD** networks.
8. **Type-I** and **Type-II** based downlink precoding in 5G **TDD** networks.
9. Beam-refinement (P2 procedure) using CSI-RS in 5G networks.
10. Channel quality indicator(**CQI**) estimation for link adaptation in 5G networks.
11. Rank adaptation for downlink **massive MIMO** in 5G networks.
12. Downlink and uplink power control for massive MIMO in 5G networks.
13. Downlink scheduling and resource allocation in 5G networks.

10 | PDCCH Implementation and Blind Decoding in 5G Networks

In 5G New Radio (NR), the **PDCCH (Physical Downlink Control Channel)** plays a crucial role in downlink transmission. It is like the heart of data communication. If the UE is unable to decode PDCCH, then no matter what radio conditions are, UE can never be able to know the downlink/uplink grants for Physical Downlink Shared Channel (PDSCH)/Physical Uplink Shared Channel (PUSCH) and hence a user can not decode data.

Consider a scenario where we are watching a video on our mobile phone. Now the question that comes to our mind is how does our mobile phone knows on what time-frequency resources my data is coming, what modulation order and code-rate that a BS picks for data transmission, how many number of layers were used for transmission etc. All this critical information is defined as **downlink control information (DCI)**. Now if our mobile phone fails to decode this information, then our mobile phone does not where to look for on a time-frequency resource grid and hence will not be able to display the video.

10.1 | What is the purpose of PDCCH?

- PDCCH carries downlink control information (DCI), which is essential for the user equipment (UE) to interpret and process the data transmitted on the Physical Downlink Shared Channel (PDSCH).
- Essentially, it is the channel that delivers critical instructions to the UE regarding resource allocation, scheduling, and other control information.

Typically, DCI contains scheduling information such as the time-frequency resources that a base station (BS) allocates for UE, PDSCH related parameters such as rank indicator (RI), which specifies the number of layers that a BS is configured to transmit, Modulation order (M) and code rate (r) etc.

3GPP specifies 5 DCI formats depending on the type of the content that it carries and configuration of the network. These formats are denoted with a number followed by another number which we denote with x . For instance, a DCI Format i_x , where $i \in 0, 1, 2, 3, 4$ and $x \in 0, 1, 2, 3, 4, 5, 6, 7$. Table 10.1 elaborates DCI formats supported. For instance, DCI format 1_0 carries scheduling information of PDSCH in a single cell and Format 2_0 specifies the slot format, available RB sets to a group of UEs. Since DCI carries crucial information, it must be decodable at UE regardless of how poor the channel conditions are. To ensure desired reliability requirements DCI is processed through a sequence of signal processing steps, which is defined as PDCCH Tx chain

10.2 | PDCCH Tx Chain

The block diagram is shown in the figure 10.1 which captures the sequence of signal processing steps. The detailed explanation of these steps is as follows

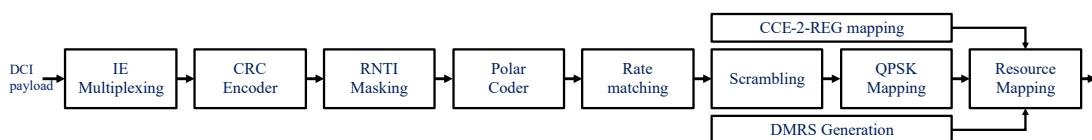


Figure 10.1: PDCCH transmitter chain

1. The first step is information element (IE) multiplexing, where a DCI payload bit sequences having different formats are multiplexed into a single bit sequence for further processing.
2. The next step is to cyclic redundancy check (CRC) encode the bits, which enables the receiver to detect any errors in the transmitted bits. In this step, a 24-bit CRC is attached to the multiplexed DCI payload bit sequence.
3. In the next step, the last 16 bits of CRC encoded DCI bit sequence is masked with a unique Radio Network Temporary Identifier (RNTI). The purpose of this step is to make a UE understand whether the transmitted DCI is intended to it or to some other UE under unicast transmission and to a group of UEs under multicast transmission.

Table 10.1: DCI Formats

DCI format	Usage
0_0	Scheduling of PUSCH in one cell
0_1	Scheduling of one or multiple PUSCH in one cell, or indicating downlink feedback information for configured grant PUSCH (CG-DFI)
0_2	Scheduling of PUSCH in one cell
1_0	Scheduling of PDSCH in one cell
1_1	Scheduling of one or multiple PDSCH in one cell, and/or triggering one shot HARQ- ACK codebook feedback
1_2	Scheduling of PDSCH in one cell
2_0	Notifying a group of UEs of the slot format, available RB sets, COT duration and search space set group switching
2_1	Notifying a group of UEs of the PRB(s) and OFDM symbol(s) where UE may assume no transmission is intended for the UE
2_2	Transmission of TPC commands for PUCCH and PUSCH
2_3	Transmission of a group of TPC commands for SRS transmissions by one or more UEs
2_4	Notifying a group of UEs of the PRB(s) and OFDM symbol(s) where UE cancels the corresponding UL transmission from the UE
2_5	Notifying the availability of soft resources as defined in Clause 9.3.1 of [10, TS 38.473]
2_6	Notifying the power saving information outside DRX Active Time for one or more UEs
2_7	Notifying paging early indication and TRS availability indication for one or more UEs.
3_0	Scheduling of NR side-link in one cell
3_1	Scheduling of LTE side-link in one cell
4_0	Scheduling of PDSCH with CRC scrambled by MCCHRNTI/G-RNTI for broadcast
4_1	Scheduling of PDSCH with CRC scrambled by G-RNTI/GCS-RNTI for multicast
4_2	Scheduling of PDSCH with CRC scrambled by G-RNTI/GCS-RNTI for multicast

4. Following to this step the bits are polar coded, a forward error correction (FEC) code adopted in 5G for PDCCH. The purpose of this step is to ensure the receiver to correct errors introduced by the channel.
5. The next step is to rate match the polar coded bits. Rate matching ensures that the DCI bits fits into the allocated resources for transmission.
6. Next step is to scramble the rate matched bits. DCI payload is scrambled using a pseudo random sequence generated by cell-id and UE specific RNTI. Scrambling ensures that the interference among different UEs is randomized.
7. In the following step scrambled bits are QPSK modulated, where every 2 bits are mapped to a complex number.
8. Now the last step is to map these QPSK symbols into the resources allocated for PDCCH transmission. Resource mapping in PDCCH is always done in the units of control channel elements or CCEs. We will see the resource mapping in detail in the following section.

10.2.1 | Resource mapping in PDCCH

Resource mapping in PDCCH is always done in the units of CCEs. The notions of CCE, Aggregation Level (AL) and PDCCH candidate is crucial to understand the resource mapping. So, we will cover these concepts in detail next.

What is a CCE?

1 CCE = 6 REG = 72 RE
 1 REG = 1 RB with 1 OFDM symbol = 12 RE

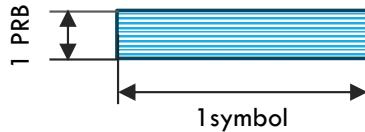


Figure 10.2: REG definition

On a downlink resource grid CCE-REG mapping is shown in the figure 10.3

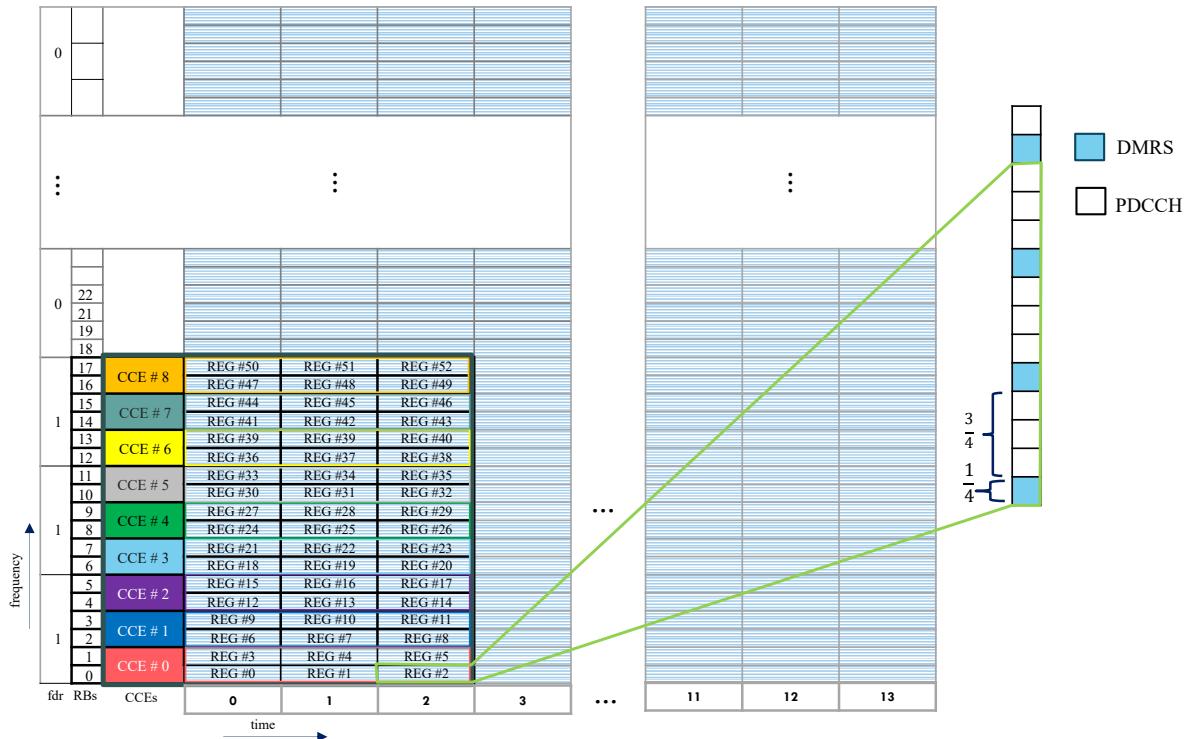


Figure 10.3: CCE-REG Mapping

What is Aggregation Level? The number of CCEs that a BS chose for the transmission of PDCCH is defined as Aggregation Level (AL). PDCCH occupies a specific number of resource elements (REs) according to the AL chosen by BS. Supported ALs are 1, 2, 4, 8 and 16.

BS always chose lower AL for users nearby (typically having LOS path with BS) having good coverage and higher AL for users far away having extreme coverage condition.

How the number of target bits are computed? The number of target bits are computed once the BS picks a particular AL depending on the coverage condition. The number of PDCCH symbol and its corresponding DMRS are functions of AL.

For an AL of L, BS chooses L consecutive CCEs or 72L REs on a Downlink resource grid for the transmission of PDCCH including its DMRS. Among these 72L REs, 3/4th is allocated for actual PDCCH symbols and the rest 1/4th is allocated to its DMRS, and the number of target bits is 2 times the number of PDCCH symbols.

What is a PDCCH Candidate? L consecutive CCEs for an AL of L is defined as a PDCCH candidate. PDCCH is transmitted through PDCCH candidates and are located within a CORESET. We will explain the CORESET in detail in the next section 10.2.2. The position of different PDCCH candidates for each

AL is determined using a hash equation.

$$L \cdot \left\{ \left(Y_{p,n_{s,f}^{\mu}} + \left\lfloor \frac{m_{s,n_{CI}}^{(L)} \cdot N_{CCE,p}}{L \cdot M_{s,max}^{(L)}} \right\rfloor + n_{CI} \right) \bmod \left\lfloor \frac{N_{CCE,p}}{L} \right\rfloor \right\} + i \quad (10.1)$$

where $i = 0, 1, \dots, L - 1$.

10.2.2 | CORESET

CORESET or Control Resource Element Set is defined as the set of REs where PDCCH is transmitted. The number of RBs and the number of OFDM symbol that a CORESET occupy is configured through higher layer parameters *frequencyDomainResources*, *duration* respectively.

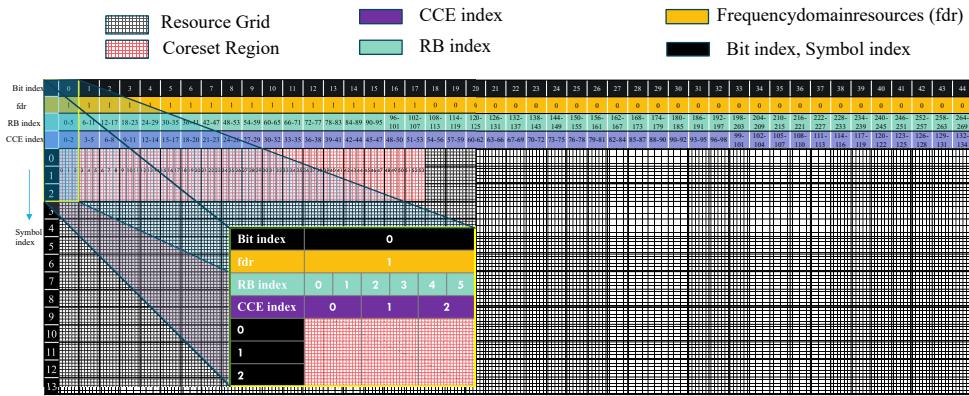


Figure 10.4: CORESET frequencyDomainResources

frequencyDomainResources is a bit string of length 45, where each bit correspond to 6 contiguous RBs and *duration* specifies contiguous time duration in number of OFDM symbols and can be either 1, 2, or 3 symbol.

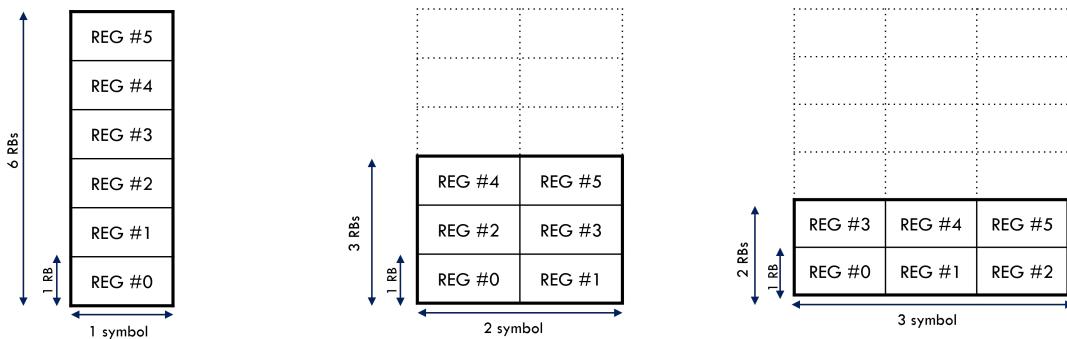


Figure 10.5: CORESET duration

Example:

$$frequencyDomainResources = [111111111111111100000000000000000000000000000]$$

$$\begin{aligned} \text{number of RBs} &= \text{number of 1s in } frequencyDomainResources \times 6 \\ &= 18 \times 6 \\ &= 108 \end{aligned}$$

$$\text{number of REGs} = \text{number of RBs} \times \text{duration}$$

$$\begin{aligned} &= 108 \times 3 \\ &= 324 \end{aligned}$$

$$\begin{aligned} \text{number of CCEs} &= \frac{\text{number of REGs}}{6} \\ &= \frac{324}{6} \\ &= 54 \end{aligned}$$

10.2.3 | PDCCH Rx Chain

PDCCH Rx chain follows exactly opposite to PDCCH Tx chain as shown in the block diagram 10.6

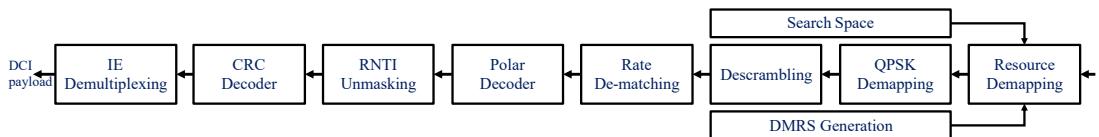


Figure 10.6: PDCCH Receiver Chain

10.2.4 | Search Space Set

The search space set is defined as the set of CCEs where a UE monitors for PDCCH continuously. The UE performs blind decoding of PDCCH candidates corresponding to the search space set. Search space set can be either a common search space set (CSS) or a UE specific search space set (USS). DCI that is common to a group of UEs are generally mapped to CSS and UE specific DCI is mapped to USS. BS generally use ALs 4, 8, 16 for CSS and 1, 2, 4, 8, 16 for USS.

Timing of CORESET is specified using search space set parameter known as *pdcchMonitoringSymbolsWithinASlot*, which is bit string of size 14 and each bit corresponds to one symbol within a slot.

We have seen that the CORESET parameter *duration* specifies the number of OFDM symbol that a CORESET occupies, but the start symbol is not specified. The start symbol of CORESET within a slot is specified using *pdcchMonitoringSymbolsWithinASlot*.

If *pdcchMonitoringSymbolsWithinASlot* = 00100000000000

Since a set bit exists at position 2 with a zero based indexing, the start symbol index of CORESET i is 2.

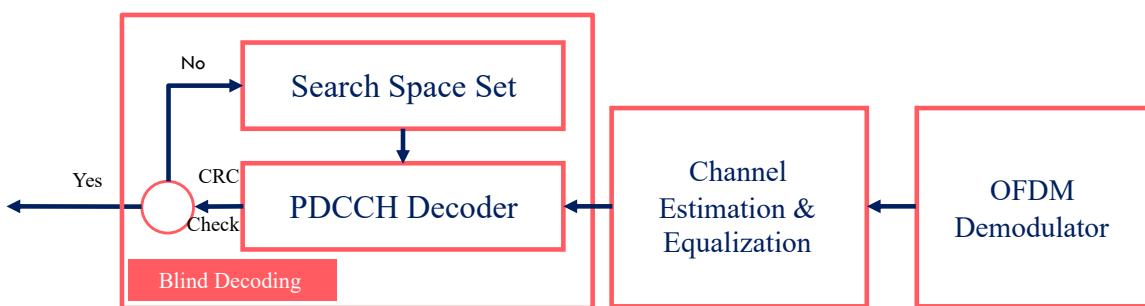


Figure 10.7: PDCCH blind decoding

10.2.5 | PDCCH Blind Decoding

UE blindly decodes PDCCH in the search space associated with a CORESET. If UE fails to decode the PDCCH, then UE will not know the location of PDSCH resources and it will keep attempting to decode the PDCCH using different set of PDCCH candidates.

UE blindly checks for CRC assuming that its DCI was mapped to a particular PDCCH candidate with each AL associated with in that search space. If it fails it repeats the same step of checking for CRC with next PDCCH candidate in sequence until the CRC check passes. Once CRC check passes, UE is able to decode its DCI. If a UE fails in CRC check of all the available candidates, then it must not be able to decode its DCI.

Example: Consider a CORESET size of 16 and the search space set associated with a CORESET is USS. Assume that the BS uses an AL of 2 and the number of PDCCH candidates that a UE has to search is 4. For simplicity we assume that all the candidates of AL other than 2 are 0. i.e., number of candidates per AL = [0, 4, 0, 0, 0, 0] and the CCE indices corresponding to candidates are [[0,1], [4,5], [8,9], [12,13]] as shown in [10.8](#). Also assume that BS maps the DCI corresponding to UE with CCE indices of [4,5] or candidate numbered 1 with a RNTI of 51585.

CCE #0	CCE #1	CCE #2	CCE #3	CCE #4	CCE #5	CCE #6	CCE #7	CCE #8	CCE #9	CCE #10	CCE #11	CCE #12	CCE #13	CCE #14	CCE #15
Candidate 0	Candidate 1	Candidate 2	Candidate 3	Candidate 4	Candidate 5	Candidate 6	Candidate 7								

Figure 10.8: Blind Decoding of PDCCH Candidates

Now UE starts decoding blindly by CRC check assuming that its DCI was mapped to first candidate having CCE indices [0,1]. Since its DCI was not mapped to this candidate, CRC check fails as shown in [10.9](#). Since CRC check fails with the previous candidate, now UE repeats blind decoding with next

CCE #0	CCE #1	CCE #2	CCE #3	CCE #4	CCE #5	CCE #6	CCE #7	CCE #8	CCE #9	CCE #10	CCE #11	CCE #12	CCE #13	CCE #14	CCE #15
Candidate 0	Candidate 1	Candidate 2	Candidate 3	Candidate 4	Candidate 5	Candidate 6	Candidate 7								



Figure 10.9: PDCCH blind decoding fails

candidate in sequence having CCE indices [4,5]. Here CRC check passes as this was the candidate that BS indeed used for mapping its DCI and UE is able to decode its DCI as shown in figure [10.10](#).

CCE #0	CCE #1	CCE #2	CCE #3	CCE #4	CCE #5	CCE #6	CCE #7	CCE #8	CCE #9	CCE #10	CCE #11	CCE #12	CCE #13	CCE #14	CCE #15
Candidate 0	Candidate 1	Candidate 2	Candidate 3	Candidate 4	Candidate 5	Candidate 6	Candidate 7								



Figure 10.10: PDCCH blind decoding successful

10.3 | References

11 | Data Communication using PDSCH in 5G Networks

The primary objective of communication in 5G networks is to transmit data from the transmitter to the receiver. This information is conveyed in the downlink and uplink using the physical downlink shared channel (PDSCH) and physical uplink shared channel (PUSCH) respectively. The scope of this chapter is limited to downlink data communication using PDSCH. We will discuss the design of the air interface for PDSCH and all the MAC and RRC parameters that influence the throughput and reliability performance of the PDSCH.

11.1 | What is PDSCH?

The Physical Downlink Shared Channel (PDSCH) is a downlink physical channel responsible for carrying user equipment (UE) data, system information blocks (SIB), and random access responses (RARs). As the term 'downlink' suggests, it is transmitted from the Base Station (BS) to the User Equipment (UE). PDSCH is scheduled for a specific UE by the BS either using the DCI carried by the Physical Downlink Control Channel (PDCCH) or through RRC signaling. PDSCH performs a sequence of signal processing steps, as shown in Fig-11.1, to ensure

- high reliability,
- maximum throughput,
- secure communication, and
- adaptability to channel conditions.

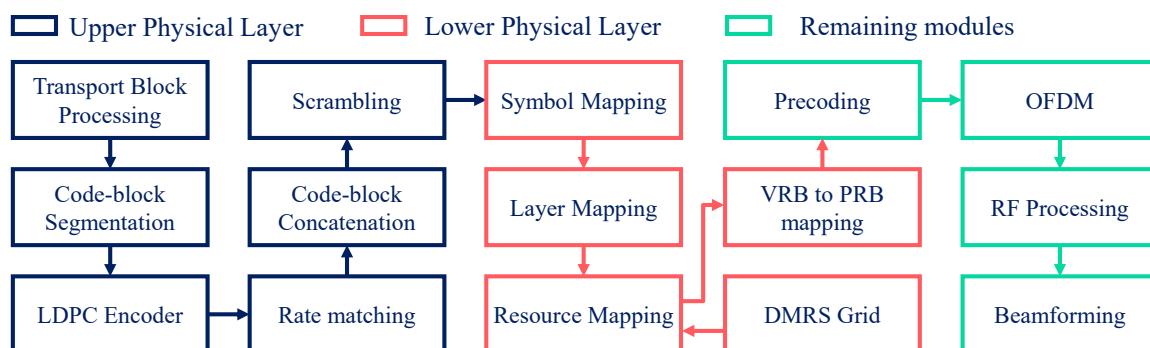


Figure 11.1: PDSCH Transmitter Chain: At BS

The PDSCH receiver at the UE performs complementary operations to accurately decode the data, as illustrated in Fig-11.2. Additionally, the receiver undertakes steps to mitigate the effects of hardware impairments such as carrier frequency offset [6.2], time offset correction [3], and frequency offset correction [3]. While these modules are generally not specified in the 3GPP standards, they are crucial for normal system operations.

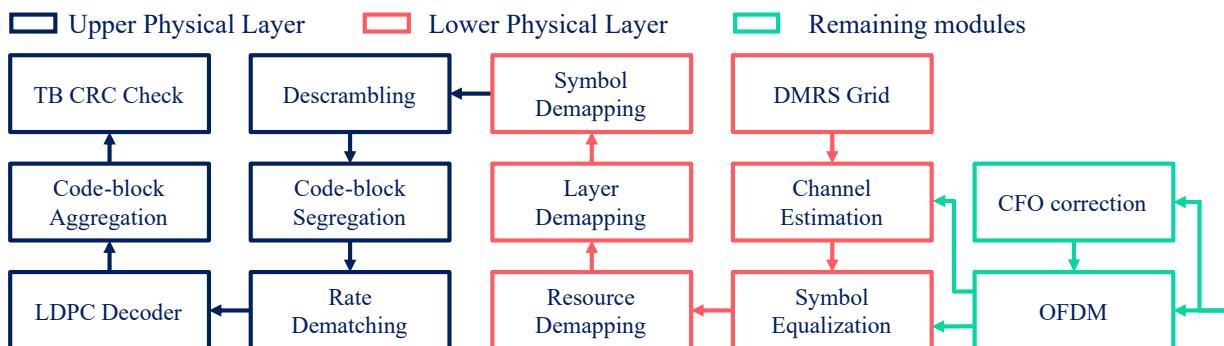


Figure 11.2: PDSCH Receiver Chain: At UE

11.2 | Design of the PDSCH Chain: Transmitter

The PDSCH chain is divided into upper physical layer PDSCH chain and lower physical layer PDSCH chain. Upper physical layer PDSCH chain consists of Transport Block(TB) Processing, code block segmentation, LDPC encoder, rate matching and code block concatenation. On the other hand, the lower PHY blocks consists of scrambling, symbol mapping, layer mapping, resource mapping and virtual resource block (VRB) to physical resource block (PRB) mapping. Each of these modules serves a specific role in the PDSCH chain. In the upcoming subsections, we will discuss the importance and function of each of these modules.

11.2.1 | Transport Block Processing

The packet to be transmitted to each UE is passed to the physical layer by the MAC layer. This packet, when in the physical layer, is referred to as a transport block. The size of this transport block is computed by the MAC scheduler using parameters such as the code rate (r), modulation order (Q_m), number of MIMO layers (N_L) configured, bandwidth (B), number of OFDM symbols (N_{symb}) allocated to the user, and the reference signal overhead. This module attaches cyclic redundancy check (CRC) bits to the transport block received from the MAC layer, as shown in Fig-11.3a. These bits are attached to the transport block for error detection and enabling hybrid automatic repeat request (HARQ) for re-transmission of the transport block. They cannot be used for error correction in the transport block. The exact length of the CRC attached is selected based on the size of the transport block, as described in equation-11.1, with a higher number of CRC bits required as the size of the transport block increases.

$$L_{\text{TB}} = \begin{cases} 24 & \text{TB length } > 3824 \\ 16 & \text{TB length } \leq 3824 \end{cases} \quad (11.1)$$

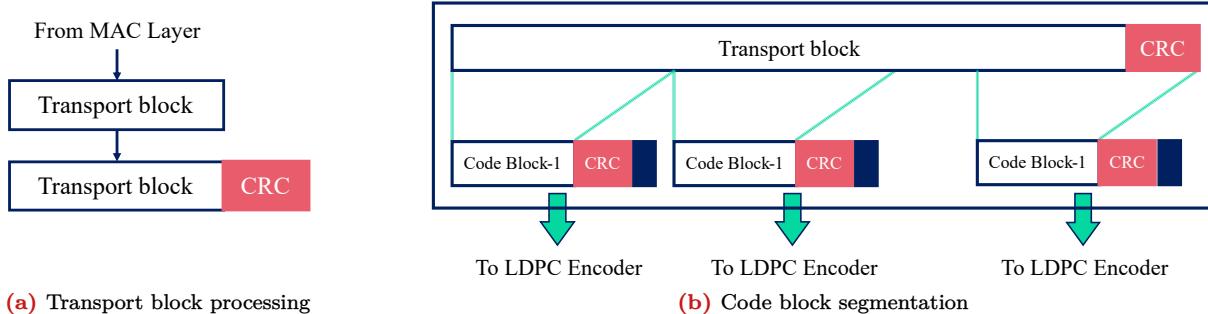


Figure 11.3: Transport block processing and code block segmentation for PDSCH in 5G

11.2.2 | Code Block Segmentation

The bits received from the TB processing are expected to be channel encoded using LDPC encoder. This encoder can support a certain maximum size of input block (N_{CB}^{\max}) to keep the complexity of LDPC decoding low at the UE. The value of N_{CB}^{\max} depends on the base-graph selected for LDPC encoding,

$$N_{\text{CB}}^{\max} = \begin{cases} 8448 & \text{if BG-1 is selected} \\ 3840 & \text{if BG-2 is selected} \end{cases} \quad (11.2)$$

If the TB size exceeds this maximum size limit N_{TB}^{\max} , the TB is segmented into $K_{\text{CB}} = \left\lceil \frac{N_{\text{TB}} + L_{\text{TB}}}{N_{\text{CB}}^{\max} - 24} \right\rceil$, code-block. Furthermore, 24 bit CRC is attached to each code-block for error detection. The block can be summarized as follows,

$$N_{\text{CB}} = \begin{cases} N_{\text{TB}} + L_{\text{TB}} & N_{\text{TB}} + L_{\text{TB}} > N_{\text{CB}}^{\max} \\ \left\lfloor \frac{N_{\text{TB}} + L_{\text{TB}}}{K_{\text{CB}}} \right\rfloor + 24 & \text{else} \end{cases} \quad (11.3)$$

LDPC encoder accepts inputs of only a certain discrete lengths. To match the supported input length this module adds some filler bits. These bits are removed after channel encoding as shown in Fig-11.3b.

11.2.3 | Channel Coding: Low Density Parity Check Codes

PDSCH uses LDPC as inner channel code for correcting the errors introduced by fading channels and poor link budgets. The decoder for the proposed LDPC channel codes can be implemented efficiently in-terms of silicon foot-print and power consumption by splitting the parity check matrix into smaller units supporting large parallelization in comparison to its predecessor. The LDPC codes standardized in 3GPP is designed have to have following properties [2]:

- **Systematic code:** Computes the parity bits and append them at the end of the information bits. The generator matrix is of the form $[I_k|P_{n-k}]$. This property is desirable for supporting IR-HARQ and low complexity implementation of the code-word.
- **Protograph-based code:** The LDPC codes in 5G uses two different base-graphs, BG-1 and BG-2, supporting the mother code-rates of $\frac{1}{3}$ and $\frac{1}{5}$ respectively. The PCM for both the base-graphs is constructed by lifting the respective base-matrices $H_{BG1} \in \mathbb{1}^{46 \times 68}$ and $H_{BG2} \in \mathbb{1}^{42 \times 52}$ as shown in Fig-11.4b.

Table 11.1: The dimensions of Input-output interface and LDPC hyper-parameters

Parameter	BG-1	BG-2
Mother code-rate	$\frac{1}{3}$	$\frac{1}{5}$
Parity check matrix	$46Z_c \times 68Z_c$	$42Z_c \times 52Z_c$
Generator matrix	$66Z_c \times 22Z_c$	$50Z_c \times 10Z_c$
Number of Information bits	$22Z_c$	$10Z_c$
Number of parity bits	$44Z_c$	$40Z_c$
Number of encoded bits	$66Z_c$	$50Z_c$

- **Quasi-cyclic code:** The cyclic rotation of the code-word will result in another valid code-word. This property is useful in breaking the complex generation and decoding process into simpler unit decoding.
- **Irregular code:** A code is termed as regular codes if number of 1s in rows of the parity check matrix, $w_r = w_c \cdot \frac{n-k}{n}$ where w_c is the number of ones in the columns of the parity check matrix. Irregular LDPC codes prioritize more important data by mapping them into higher degree protection classes, thereby providing increased protection. The use of different degree protection classes in an LDPC code enhances the overall performance of data transmission against channel errors.

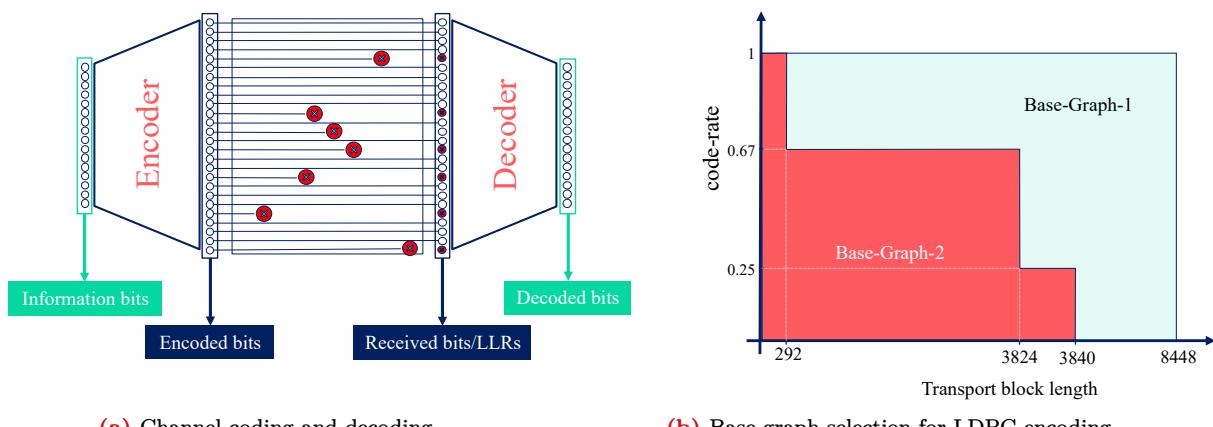


Figure 11.4: LDPC Channel coding and Base-graph selection for PDSCH

The 5G LDPC encoder lifts the base protograph matrix of 46×68 or 42×52 to $46Z_c \times 68Z_c$ or $42Z_c \times 52Z_c$ for BG-1 or BG-2 respectively where Z_c is smallest integer that satisfy $20Z_c \geq N_{CB}$. The first $2Z_c$ columns of the generator matrix are punctured to maintain the mother code-rate of $\frac{1}{3}$ and $\frac{1}{5}$ for BG-1 and BG-2 respectively.

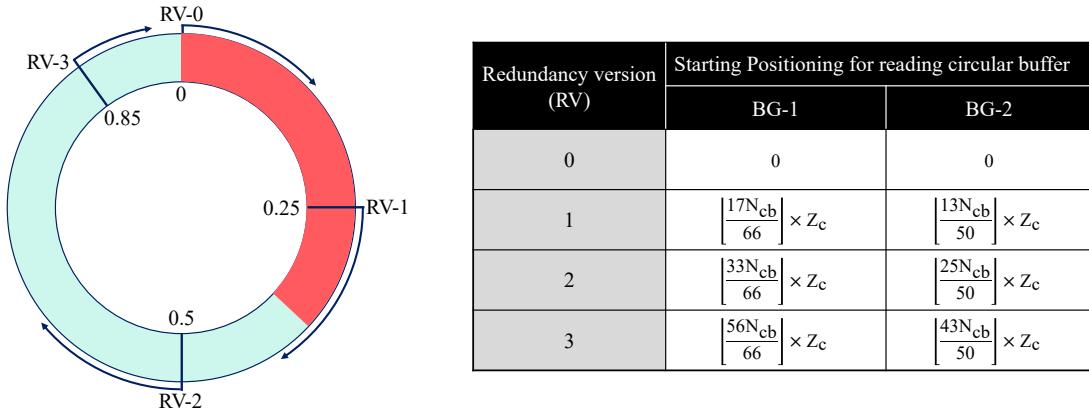


Figure 11.5: Redundancy version for HARQ in 5G-NR

11.2.4 | Rate Matching

The rate-matching is process of shortlisting and processing the encoded bits to match a target number of bits (E) that can fit into the time, frequency, and space resources allocated to a specific UE. The rate matcher comprises two sub-modules: **bit selection** and **bit interleaver**.

In bit selection, the encoded bits are mapped into a circular buffer, as shown in Fig-11.6, where systematic bits are loaded first followed by parity bits. The size of the circular buffer, N_{buff} , can be as large as $66Z_c/50Z_c$ for BG-1/BG-2, respectively. However, in the uplink for PUSCH, the buffer size can be reduced using limited buffer rate matching (LBRM) features to control the buffering of data for HARQ to reduce the cost. This reduction in cost comes at the expense of channel decoding performance when the size of the code-block is large. The bits to be transmitted from the circular buffer are selected based on the redundancy version configured using $RV-ID$, enabling HARQ. 5G-NR supports four versions of redundancy for HARQ, as shown in Fig-11.5, which defines the starting bit location for fetching the next E bits from the circular buffer. Among these versions, only RV-0 is completely self-decodable. Hence, the UE might not decode data transmitted using RV-1, RV-2, and RV-3 alone, especially when E is very small. The RV-ID is configured to the BS by the scheduler based and to the UE by DCI.

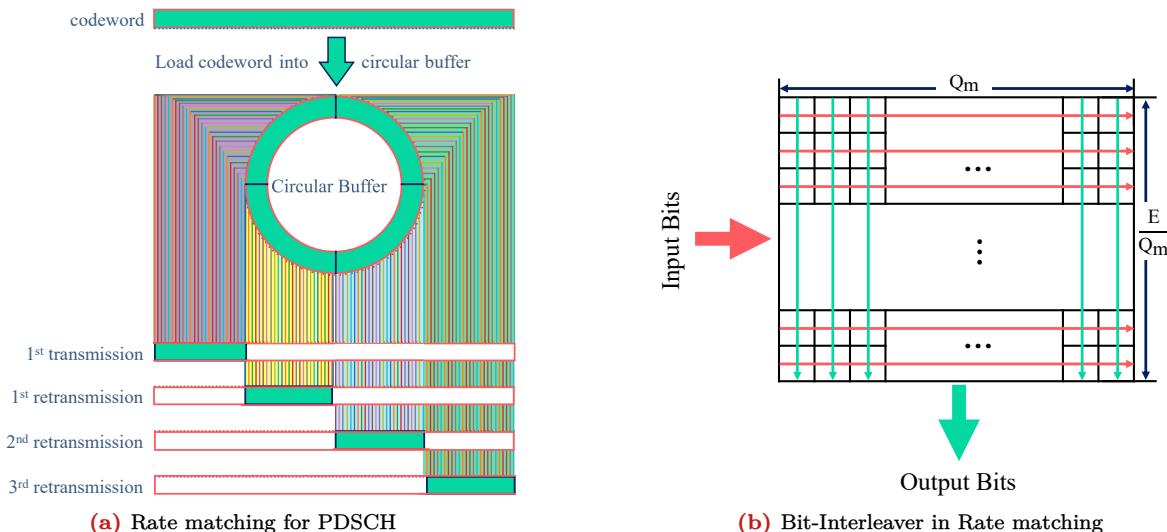


Figure 11.6: Bit Selection and Bit Interleaving in rate matching for PDSCH

The bit interleaver utilizes a rectangular interleaver with Q_m columns and E/Q_m rows. It writes the

information row-wise and reads it column-wise. This interleaver is introduced in the LDPC rate matcher to protect against burst errors introduced by frequency-selective deep-fading. This fading occurs due to very low SNR across multiple consecutive REs, resulting in poor channel estimation performance and, eventually, poor equalization results. This phenomenon leads to errors in decoding consecutive constellation symbols. The bit interleaver spreads these burst errors across the entire codeword before passing it to the LDPC decoder, which performs poorly against burst errors.

11.2.5 | Code Block Concatenation

The rate-matched bits outputted by each parallel LDPC codec (LDPC encoder + rate-matcher) are concatenated to create a single code-word (stream of bits) in a round-robin fashion, as shown in Fig-11.7. It's worth noting that the number of target bits generated by each LDPC codec can be different.

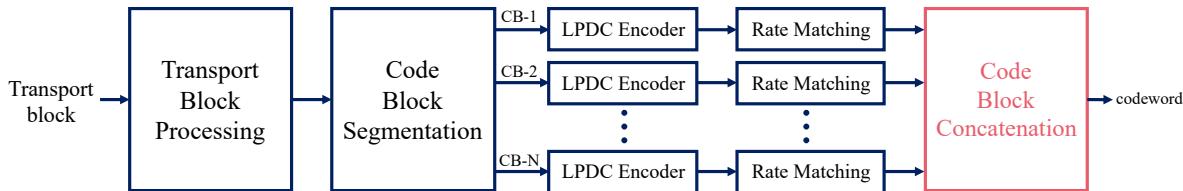


Figure 11.7: Code Block Concatenation for PDSCH

11.2.6 | Scrambling

The concatenated sequence is passed to the scrambler which xor the input bit sequence with random binary sequence as shown in Fig-11.8. This helps in breaking the long sequences of ones and zeros, and randomize the interference of PDSCH.

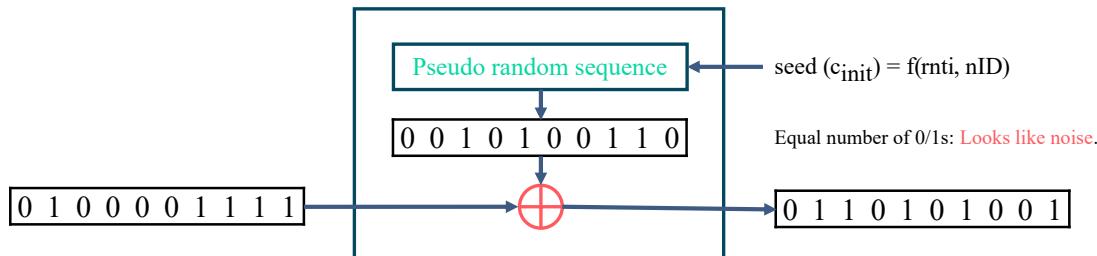


Figure 11.8: Scrambling for PDSCH

The random sequence at the BS is generated using a seed, which is computed using a *scramblingID* and *RNTI*. This ID is configured to a UE via RRC messages in RRC connected mode for data decoding. During the initial access phase, this ID is configured using Master Information Block (MIB) or System Information Block (SIB) messages.

11.2.7 | Symbol Mapping

The symbol mapper converts the bits into complex modulation symbols. The PDSCH supports PSK and QAM, as detailed in table-11.2. The modulation order (Q_m) defines the number of bits mapped onto a single symbol. It is configured to the BS and UE using *mcs-Index* and the *mcs-table* by the scheduler and DCI, respectively. This parameter is selected based on the quality of the channel between the BS and UE and the transmit power. The constellation diagram of 16-QAM is shown in Fig-11.9a.

Table 11.2: Modulation orders supported in 5G

Constellation	$\frac{\pi}{2}$ -BPSK	QPSK	16-QAM	64-QAM	256-QAM	1024-QAM
Modulation order (Q_m)	1	2	4	6	8	10

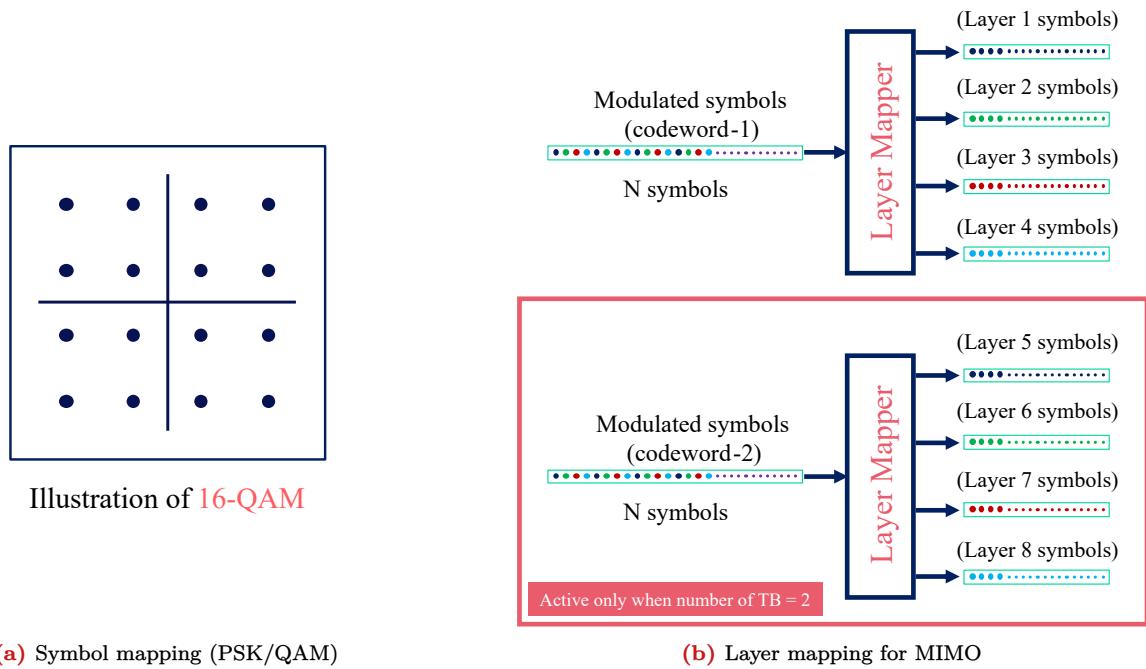


Figure 11.9: Symbol mapping and layer mapping in PDSCH

11.2.8 | Layer Mapping

The layer mapper is utilized in MIMO systems where multiple layers (N_L) are transmitted for PDSCH. It converts the serial information symbol streams of length (N_{symb}) into parallel (N_L) streams, with each stream carrying an equal number of information symbols ($\frac{(N_{\text{symb}})}{N_L}$), as illustrated in Fig-11.9b. The details of the codeword to layer mapping are provided in table-7.3.1.3-1 and section 7.3.1.3 of [1]

In 5G-NR, up to 8 layers can be transmitted for PDSCH based on the rank indicator configured to the BS by the scheduler and to the UE by DCI. For more than 4 layers, the UE expects the transmission of 2 transport blocks, with each transport block processed independently by the upper physical layer. The segregation of transport blocks helps in reducing the re-transmission overhead via HARQ. The number of layers for PDSCH transmission is selected based on the channel condition between the transmitter and receiver antennas. These channel conditions are determined either by the UE using CSI-RS or by the BS using SRS, assuming TDD reciprocity.

11.2.9 | Resource Mapping

5G-NR uses OFDM waveform as the part of air interface. In OFDM, the time frequency and space (MIMO) resources are organized in the form of orthogonal 3D grids, ($N_{\text{port}} \times N_{\text{symbols}} \times N_{\text{sc}}$) called resource grids as discussed in detail in chapter-4. This resource grid defines all the resources that are available with the BS for data transmission. The PDSCH symbols received from the layer mapper are mapped to the resource grid where the number of ports must equal the number of layers. The information symbols corresponding to each layer will be mapped to a specific antenna port. The resource mapping for PDSCH is defined in section 7.3.1.5 of [1]. The PDSCH symbols are expected to not overlap with any reference signal scheduled either by RRC or MAC. The details about the resource mapping are captured in next section.

11.2.10 | VRB to PRB Mapping

5G-NR supports virtual resource block (VRB) to physical resource block (PRB) mapping, which allows networks to divide the UE bandwidth part (BWP) into multiple RB bundles, as shown in Fig-11.10b. These RB bundles can be either directly mapped to PRBs or mapped in an interleaved fashion to exploit frequency diversity for robustness against multi-path fading. However, it's important to note that interleaved VRB-to-PRB mapping is supported only for downlink resource allocation type 1, where frequency domain resources are allocated using resource indication value (*RIV*). Downlink resource allocation type 0 is not supported, as it may allocate non-contiguous VRBs for data transmission, which

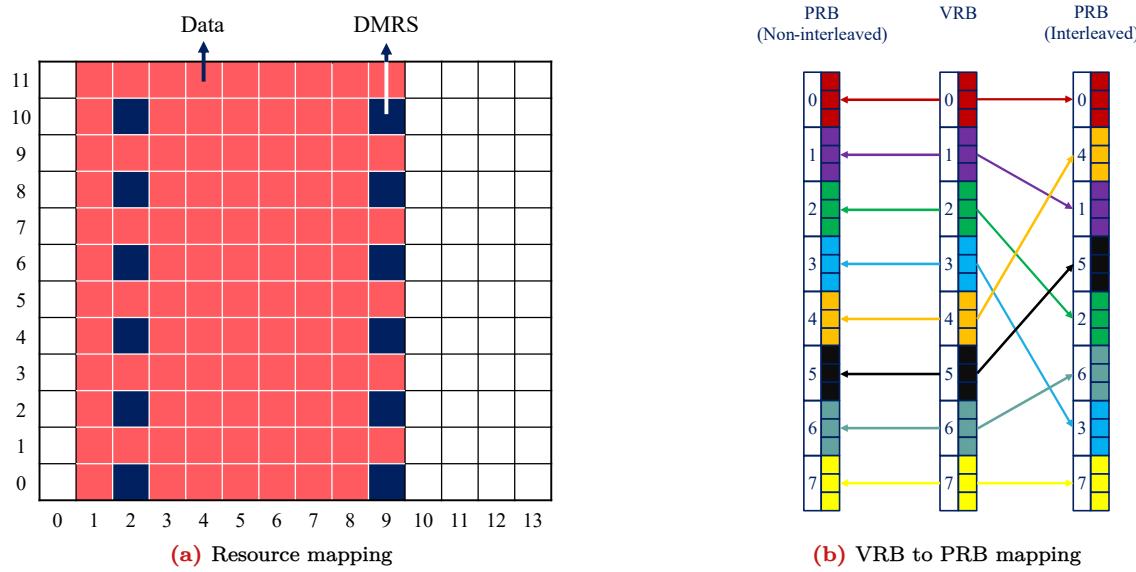


Figure 11.10: VRB to PRB mapping and resource mapping in PDSCH chain

could create issues with precoding within a PRB bundle. In interleaved mapping, the UE assumes that the same precoder is applied to all the resource elements (REs) within a PRB bundle, while the precoder across different bundles can vary.

11.3 | PDSCH Resource Allocation and Resource Mapping

11.3.1 | DMRS Generation

In 5G, the DMRS is generated by exploiting the frequency domain parameters/DMRS configuration type and time domain parameters/PDSCH mapping type. The configuration type controls the DMRS allocations in the frequency domain. 5G supports the following configuration types:

- Configuration Type-I: Supports upto 8 ports(maximum of 8 layers can be transmitted). Every alternate RE are allocated to pilot, as shown in figure 11.11. Each RB has 6 pilots thus having pilot density of 50%.

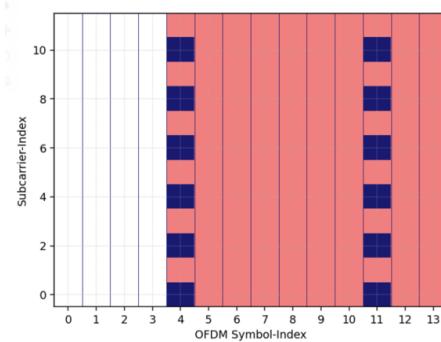


Figure 11.11: An Example of Configuration Type-I from 5G toolkit

- Configuration Type-II: Supports upto 12 ports. Each RB has 4 pilots resulting in pilot density of 33.3%, as shown in figure 11.12

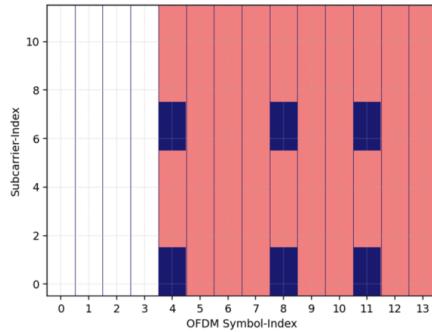


Figure 11.12: An Example of Configuration Type-II from 5G toolkit

The 5G supports two 2 mapping types:

- Mapping Type A: It is an slot based scheduling, the first symbol is determined by parameter *dmrs-TypeA-Position*, which can take value of pos2 and pos3, indicating the first DMRS symbol can be at OFDM symbol number 2 or 3, as shown in figure 11.13 and figure 11.14.

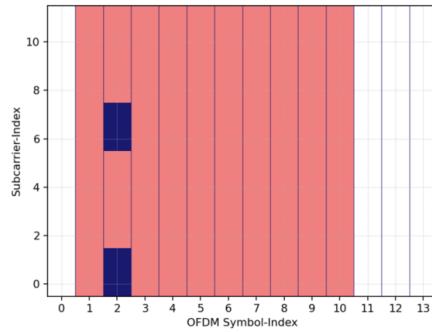


Figure 11.13: An Example of mapping type A with *dmrs-TypeA-Position* = pos2 from 5G toolkit

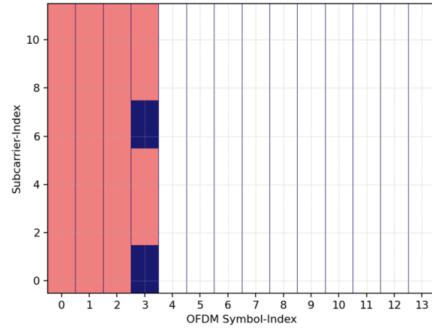


Figure 11.14: An Example of mapping type A with *dmrs-TypeA-Position* = pos3 from 5G toolkit

- Mapping Type B: It is mini slot based scheduling, where the first symbol allocated to PDSCH is allocated to DMRS as shown in figure 11.15.

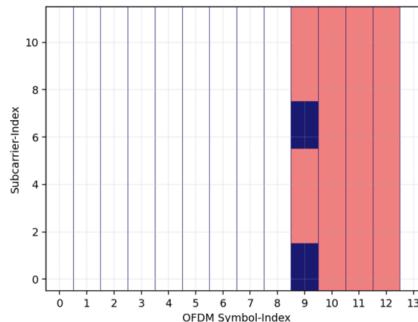


Figure 11.15: An Example of mapping type B from 5G toolkit

Additionally, 5G allows the allocation of additional symbol using the parameter *dmrs-AdditionalPosition*, which can take value of pos1, pos2 and pos3 indicating DMRS at 2, 3 and 4 symbols respectively. Moreover, 5G supports double symbols using the parameter *maxlength*, which takes value len1 and len2 indicating DMRS at single symbol and two symbols respectively.

11.4 | Design of the PDSCH Chain: Receiver

The PDSCH decoder performs mostly the complementary operation to that of PDSCH at the transmitter to decode the data. There are a few additional blocks in PDSCH receiver which are not standardized by 3GPP. These blocks are used to nullify the effect of hardware impairments. Some of these are discussed are discussed as follows.

11.4.1 | CFO Estimation and Correction

The carrier frequency offset arise in all the practical wireless system due to imperfections in the mixers and local oscillators. The mismatch in the local oscillator frequency at the transmitter and receiver results in CFO which translates into time varying phase in frequency/OFDM domain. In this tutorial, we will estimate the CFO using the PSS present in the SSB and use it to offset the phase rotation for PDSCH. The details of CFO estimation using PSS can be find in section-6.2 of chapter-6.

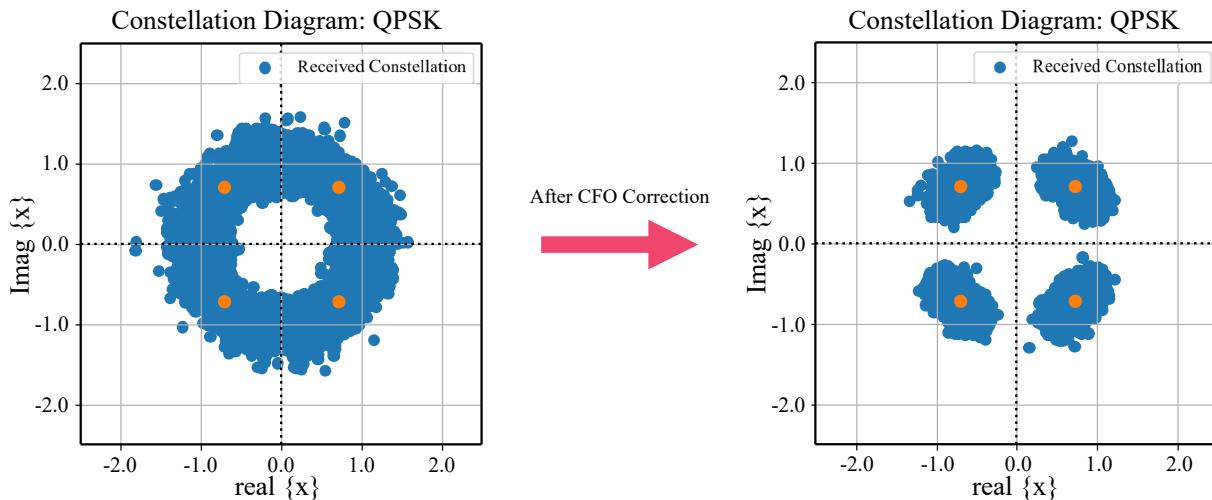


Figure 11.16: Carrier Frequency Offset (CFO) estimation and correction for PDSCH.

11.4.2 | Channel Estimation and Equalization

In 5G-NR, the channel is estimated using the DMRS transmitted alongside PDSCH as detailed in section-11.3.1. The estimated channel is used for equalization of the channel to recover the constellation symbols.

11.5 | Results

The simulation parameters and evaluation methodology considered for this experiment is given in table-11.3.

Table 11.3: Simulation parameters and evaluation methodology

Parameters	Value
center frequency (f_c)	1000 MHz
Bandwidth (B)	30 MHz
FFT size (N_{FFT})	1024
subcarrier spacing (Δf)	30 KHz
Numbers of Resource Blocks (RBs) (N_{RB})	85
Numbers of slots for PDSCH (N_{slot})	7
Numbers of BS (N_{BS})	1
PDSCH mapping type	“mapping type B”
maxLength (single or double DMRS)	‘len1’
startSymbol (OFDM start symbol)	0
configurationType	“Configuration-type-1”
dmrsTypeAPosition	“pos2”
dmrsAdditionalPosition	“pos2”
rank (N_L)	1
mcsIndex	0
mcsTable	“pdschTable1”
Transmitter-receiver separation (d)	10cm/1m

The simulation follows the procedure shown in fig-11.17

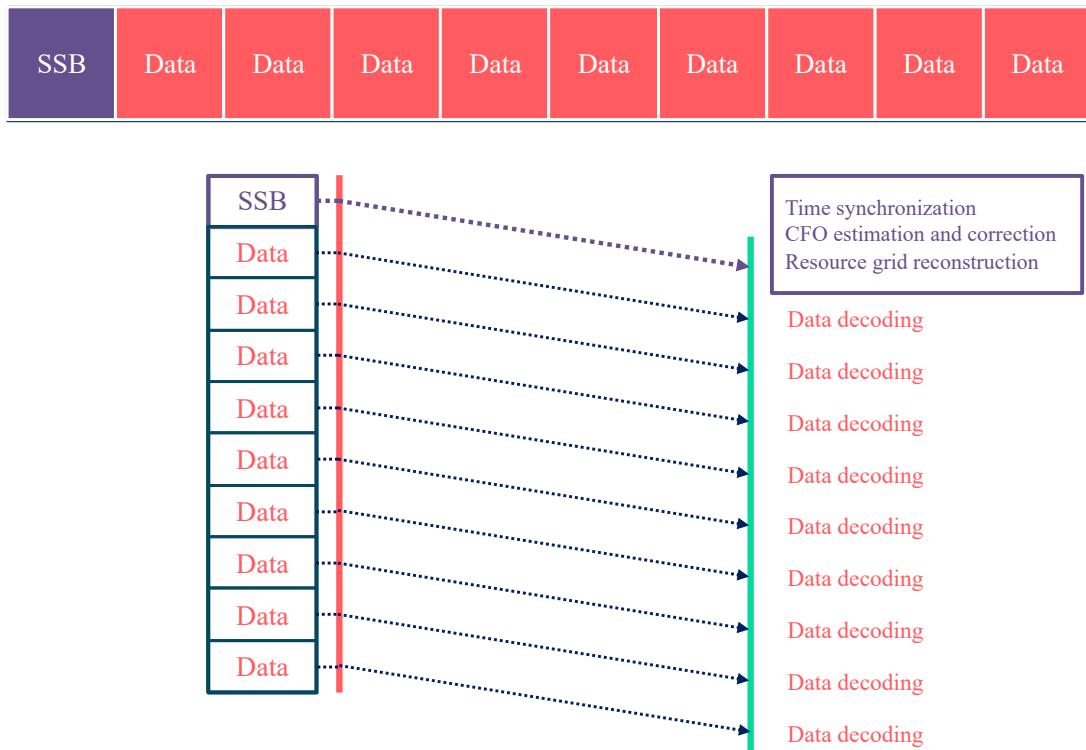


Figure 11.17: Organization of the simulations

Observation 1: *The quality of the constellation of the received signal depends on the reception of the resource grid, precision in mitigating hardware impairments, accuracy of channel estimates and symbol equalization.*

Since the downlink data is contained in PDSCH, it becomes necessary to detect PDSCH. The heat map of the received PDSCH grid is shown in figure 11.18.

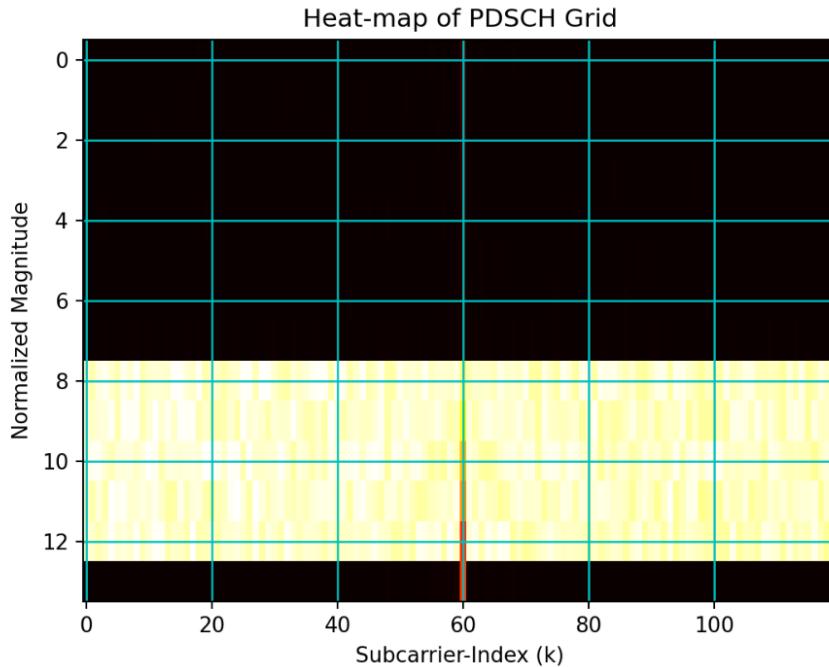


Figure 11.18: Heat map of received PDSCH grid

The figure 11.18 verifies the reception of PDSCH, since PDSCH was transmitted on 5 OFDM symbols(8-12). Recall from figure 11.2, after OFDM demodulation and CFO correction, the channel estimation and symbol equalization is carried out. The channel is estimated using the class **ChannelEstimationAndEqualizationPDSCH**, all relevant parameters are passed on to the class. To compute the equalized symbols, the inputs are passed onto the object of class: pdschGrid(the received PDSCH grid), interpolator(channel interpolator can be spline,linear or cubic, polyOrder (required only for cubic interpolator)).

After symbol equalization, the resource demapping is carried out by carrying out layer demapping and symbol demapping using the class **LayerDemapper** and **Demapper** respectively. Here the term rank refers to the numbers of transmission layers, numTBs refers to numbers of transport block and are always 1 when numbers of transmission layers is less than 4.

Upon demapper, descrambling is performed at UE as shown in figure 11.2, upon descrambling the bits are passed onto class **PDSCHDecoderUpperPhy**, which performs code block segregation, rate dematching, LDPC decoder, code block aggregation and transport block CRC removal. The received bits of PDSCH is shown in figure 11.20a. The result of observation 1 is for mcsIndex = 0 (QPSK).

Observation 2: *The mcs index allows the gNB to adapt the PDSCH transmission to channel conditions.* The modulation order and code rate are configured for transmission using MCS index. When channel conditions are favorable, higher MCS indices are selected to maximize throughput while satisfying reliability requirements. However, if the channel conditions are harsh, the network selects more conservative MCS indices. Table-11.4 clearly shows that throughput increases with MCS index until the channel conditions support it. However, once the MCS index becomes too aggressive for the channel conditions, throughput starts to degrade due to a degradation in reliability (BLER) performance.

Table 11.4: Performance for different mcs index for transmitter gain of 0 dB and receiver gain of 60 dB.

mcs Index	Q_m	r	Throughput	Spectral Efficiency	BLER
0	2	0.117	4.438 Mbps	0.147 bits/sec/Hz	0
4	2	0.3	11.2 Mbps	0.373 bits/sec/Hz	0
10	4	0.332	24.654 Mbps	0.8218 bits/sec/Hz	0
15	4	0.6	44.814 Mbps	1.4938 bits/sec/Hz	0
20	6	0.554	62.748 Mbps	2.091 bits/sec/Hz	0
23	6	0.7	3.758 Mbps	0.1252 bits/sec/Hz	0.952
24	6	0.754	0 Mbps	0 bits/sec/Hz	1

Observation 3: Higher transmit power and receiver gain are required as the distance between transmitter and receiver increases.

As the distance between transmitter and receiver increases, the power reaching at the receiver reduces. This results in degradation in the SNR at the receiving SDR. This limitation of link budget can be overcome by either transmitting higher power or higher amplification at the receiver as shown in table-11.5.

Table 11.5: Variation in performance with Tx-Rx separation for MCS Index 15

Distance		10 cm			1 m		
Tx gain	Rx gain	Throughput	BLER	BER	Throughput	BLER	BER
-20 dB	40 dB	44.814 Mbps	0	0	0 Mbps	1	0.406
-10 dB	50 dB	44.814 Mbps	0	0	0 Mbps	1	0.223
-0 dB	60 dB	44.814 Mbps	0	0	4.801 Mbps	0.892	0.010

Observation 4: The pilot density in time can be configured to manage the temporal variations in the channel arising from residual time and frequency errors or UE mobility.

The higher the time density of pilots, the better the system performance, especially when the channel is varying across time, as shown in figure 11.19. This temporal variation in the channel can result from factors such as UE mobility or residual time-frequency errors like time synchronization error, phase noise (PN), or carrier frequency offset (CFO). The increased number of demodulation reference signal (DMRS) symbols across time helps reduce the aging of channel estimates, improving symbol equalization performance, and consequently reducing the block error rate (BLER) of communications.

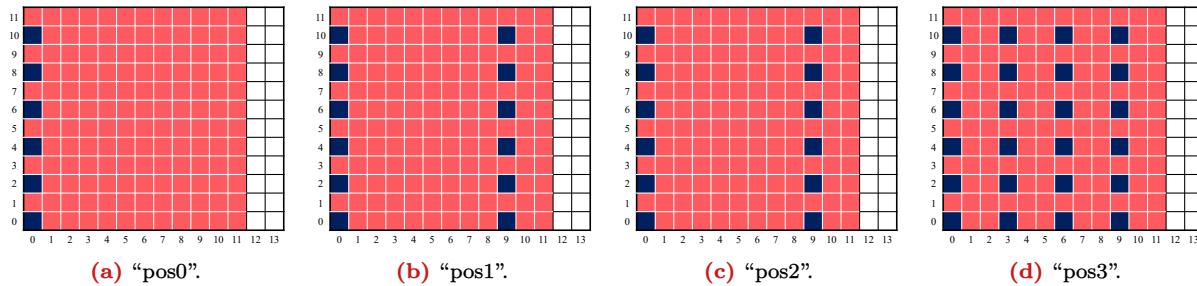
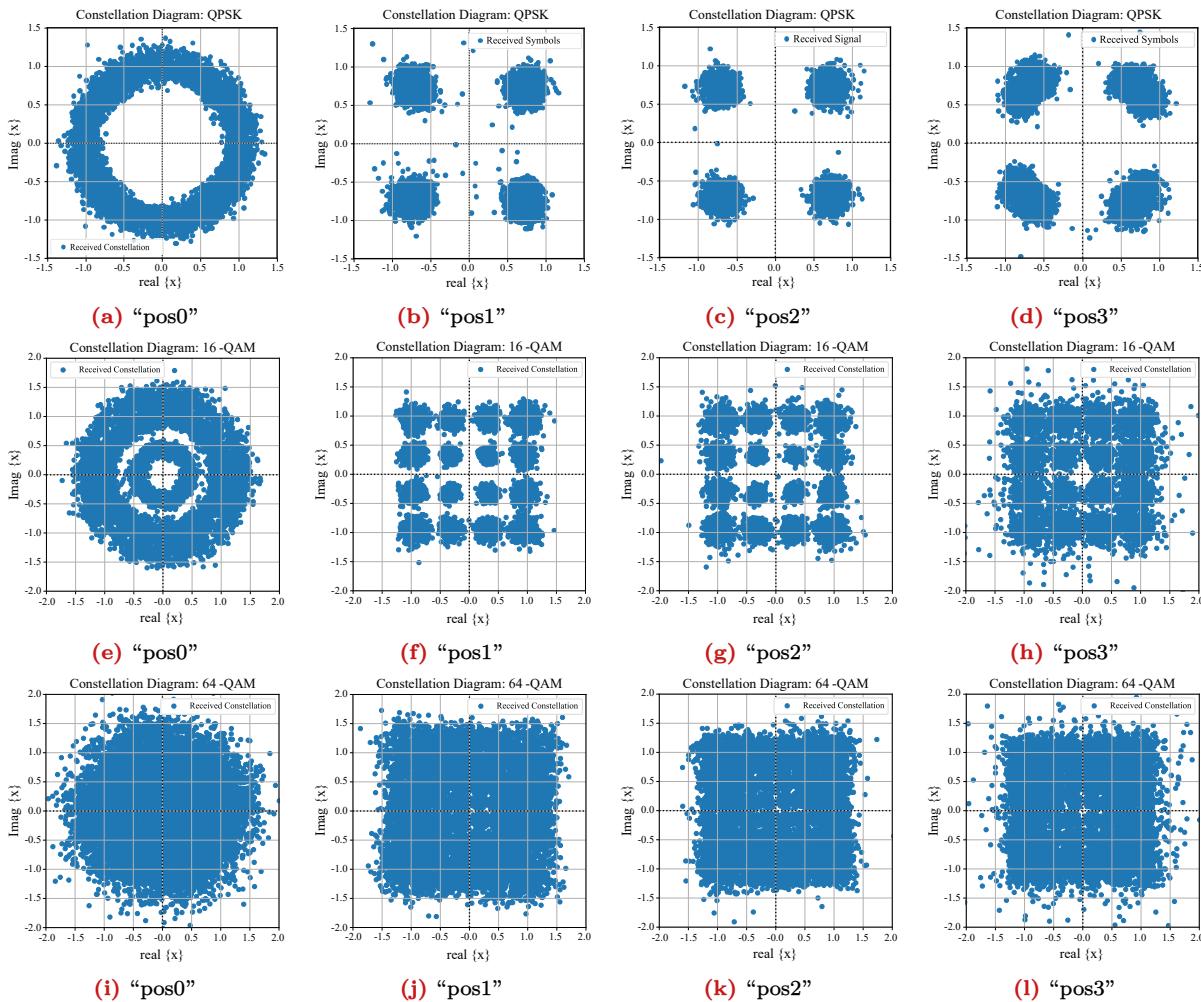


Figure 11.19: PDSCH resource mapping in for different *dmrsAdditionalPosition*.

The parameter which determines the density of pilots in time domain is *dmrsAdditionalPosition*, as also shown in figure 11.19. The table 11.6 shows the performance for different *dmrsAdditionalPosition* in PDSCH. It can be observed clearly from table 11.6 that as pilot density across time increases the performance improves. Moreover, owing to increased overhead due to pilot the throughput for “pos2” is higher than “pos3”.

Table 11.6: PDSCH performance with different *dmrsAdditionalPosition* and MCS indices (I_{MCS}).

<i>dmrsAdditionalPosition</i>	“pos0”		“pos1”		“pos2”		“pos3”	
	η	BLER	η	BLER	η	BLER	η	BLER
10	0	1	12.5475	0.75	—	0	—	—
16	0	1	50.19	0	48.40	0	45.71	0
20	0	1	31.36	0.51	62.75	0.0	49.03	0.17


Figure 11.20: Constellation of the received PDSCH symbols for mcs-index (I_{MCS}) = 4, 16, 20 (row-wise) for different *dmrsAdditionalPosition* (column-wise).

The figure 11.20 shows the constellation of received PDSCH symbols. It is quite evident from figure as pilot density increases across time the constellation looks much clear.

11.6 | Appendix

11.6.1 | CQI tables
Table 11.7: CQI tables | I_{CQI} is CQI index | Q_m is modulation order | r is code-rate.

CQI Index	CQI table 1		CQI table 2		CQI table 3		CQI table 4	
	I_{CQI}	Q_m	$r \times 1024$	Q_m	$r \times 1024$	Q_m	$r \times 1024$	Q_m
0	-	-	-	-	-	-	-	-
1	2	78	2	78	2	30	2	78
2	2	120	2	193	2	50	2	193
3	2	193	2	449	2	78	2	449
4	2	308	4	378	2	120	4	378
5	2	449	4	490	2	193	4	616
6	2	602	4	616	2	308	6	567
7	4	378	6	466	2	449	6	666
8	4	490	6	567	2	602	6	772
9	4	616	6	666	4	378	6	873
10	4	466	6	772	4	490	8	711
11	4	567	6	873	4	616	8	797
12	4	666	8	711	6	466	8	885
13	4	772	8	797	6	567	8	948
14	4	873	8	885	6	666	10	853
15	4	948	8	948	6	772	10	948

11.6.2 | MCS Table
Table 11.8: MCS tables for PDSCH | I_{MCS} is MCS index | Q_m is modulation order | r is code-rate.

MCS index	PDSCH table 1		PDSCH table 2		PDSCH table 3		PDSCH table 4	
I_{MCS}	Q_m	$r \times 1024$						
0	2	120	2	120	2	30	2	120
1	2	157	2	193	2	40	2	193
2	2	193	2	308	2	50	2	449
3	2	251	2	449	2	64	4	378
4	2	308	2	602	2	78	4	490
5	2	379	4	378	2	99	4	616
6	2	449	4	434	2	120	6	466
7	2	526	4	490	2	157	6	517
8	2	602	4	553	2	193	6	567
9	2	679	4	616	2	251	6	616
10	4	340	4	658	2	308	6	666
11	4	378	6	466	2	379	6	719
12	4	434	6	517	2	449	6	772
13	4	490	6	567	2	526	6	822
14	4	553	6	616	2	602	6	873
15	4	616	6	666	4	340	8	682.5
16	4	658	6	719	4	378	8	711
17	6	438	6	772	4	434	8	754
18	6	466	6	822	4	490	8	797
19	6	517	6	873	4	553	8	841
20	6	567	8	682.5	4	616	8	885
21	6	616	8	711	6	438	8	916.5
22	6	666	8	754	6	466	8	948
23	6	719	8	797	6	517	10	805.5
24	6	772	8	841	6	567	10	853
25	6	822	8	885	6	616	10	900.5
26	6	873	8	916.5	6	666	10	948
27	6	910	8	948	6	719	2	R
28	6	948	2	R	6	772	4	R
29	2	R	4	R	2	R	6	R
30	4	R	6	R	4	R	8	R
31	6	R	8	R	6	R	10	R

12 | Performance Analysis of 2x2 Downlink MIMO in 5G Networks

MIMO systems employ multiple antennas at both the transmitter and the receiver. This allows for the transmission of multiple streams of data simultaneously, effectively increasing the capacity and reliability of the communication link. When combined with good signal processing algorithms, MIMO technology enables the system to meet reliability, throughput, and latency requirements. In this tutorial, we will investigate the reliability and throughput performance of a 1×2 , 2×1 , and 2×2 MIMO system when configured in transmit diversity, receive diversity, and spatial multiplexing mode.

12.1 | What is MIMO?

MIMO systems consist of multiple antennas at both the transmitter and receiver, resulting in multiple spatial paths between them. These paths experience correlated or independent fading depending on propagation conditions and the spacing between the antennas at each end. The existence of multiple paths between the reduces the probability of deep fade in the link in comparison to SISO, SIMO, and MISO systems. These spatial links in the flat-faded wireless channel, H , are modelled as,

$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} & \dots & h_{1t} \\ h_{21} & h_{22} & h_{23} & \dots & h_{2t} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{r1} & h_{r2} & h_{r3} & \dots & h_{rt} \end{bmatrix} \in \mathbb{C}^{N_r \times N_t},$$

where h_{ij} denotes the wireless channel link between the i -th receive antenna and j -th transmit antennas as illustrated in Fig-12.1. The presence of these spatial paths allows the transmitter and receiver to configure the MIMO system in different spatial modes based on the channel conditions. These two modes are known as spatial diversity modes and spatial multiplexing mode.

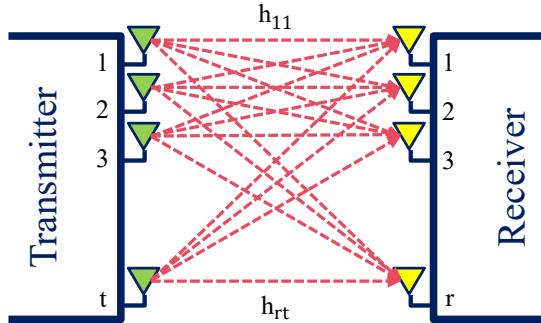


Figure 12.1: MIMO system model

12.1.1 | Basic Reliability analysis of MIMO systems

Lets us assume that the probability of a spatial link failure is p . The reliability of SISO, SIMO, MISO and MIMO are analyzed using probability of reliable communication (p_{rel}) as follows,

- Reliability of SISO systems, $p_{\text{rel}} = 1 - p$.
- Reliability of SIMO systems, $p_{\text{rel}} = 1 - p^r$.
- Reliability of MISO systems, $p_{\text{rel}} = 1 - p^t$.
- Reliability of MIMO systems, $p_{\text{rel}} = 1 - p^{(r+t)}$.

One can observe that MIMO has the highest reliability for any value of of link failure probability ($0 \leq p \leq 1$).

12.1.2 | Spatial Diversity in MIMO Systems

Spatial diversity provides protection against deep fading by combining signals from all transmit antennas which are unlikely to suffer deep fade simultaneously, thus providing reliability. The principle behind spatial diversity is when signal are transmitted from base station through multiple physically separated

antennas, they travel through different paths and thus are uncorrelated (in terms of fading) and as a result are unlikely to fade simultaneously. Upon reception at user equipment, these signals can be combined to maximize the signal-to-noise ratio (SNR). Spatial diversity can achieve:

- Increased reliability (by lowering Block Error Rate (BLER), Bit Error Rate(BER))
- Increased coverage area. In large macro cell, signal transmitted from base station will take highly uncorrelated paths before reaching user equipment.

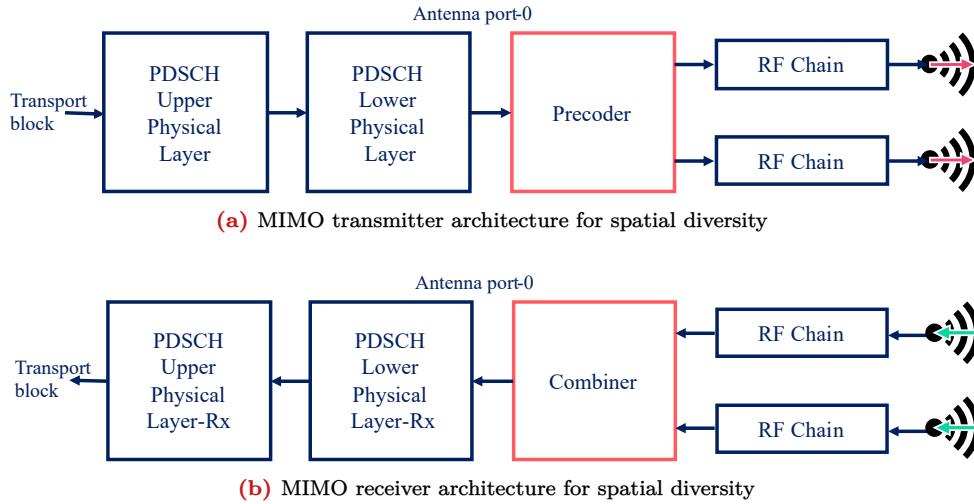


Figure 12.2: MIMO architecture for spatial diversity

12.1.3 | Spatial Multiplexing in MIMO Systems

The primary objective of 5G data communication is to achieve higher throughput (in bits/sec). This can be achieved by spatial multiplexing, wherein multiple parallel distinct streams of data is transmitted from the base station. In contrast to spatial diversity, which achieves higher reliability, spatial multiplexing achieves higher throughput.

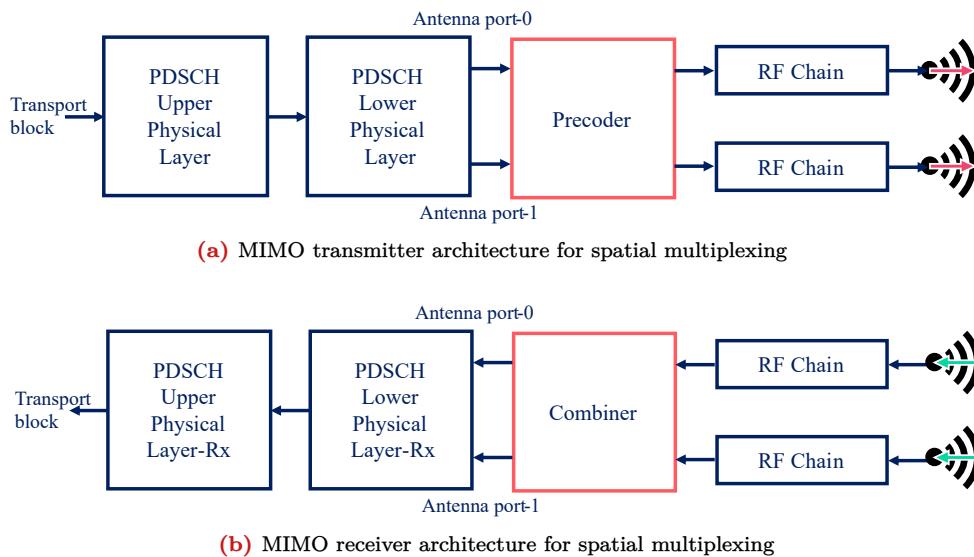


Figure 12.3: MIMO architecture for spatial multiplexing

12.2 | MIMO Aspects in 5G

5G has concept of 'antenna port', which is different from physical 'antenna elements'. At the transmitter, the payload data and demodulation reference signal(DMRS)/pilot are loaded onto antenna ports. The

transmissions using the same antenna port experience the same propagation channel, since transmission of a port is done via a single RF chain. Consequently, the payload data and DMRS in PDSCH are loaded on the same port, which ensure that both payload data and DMRS experience the same propagation channel. This allows user equipment to estimate the channel using DMRS, and thereby decode payload using the same channel estimate.

For instance, for SSB, the PSS,SSS and PBCH use the antenna port 400,i.e. all three transmission share the same antenna port and so are transmitted such that they experience the same propagation channel. Antenna port 400 is mapped onto specific antenna elements which radiate across the air-interface using a single RF chain.

12.2.1 | MIMO: Transmitter

The mapping between port to physical antenna can either be one-to-one or one-to-many. One-to-one mapping is done at lower band, since it does not require beamforming (beamforming requires multiple physical antenna elements). Considering an example for this experiment, transmission of 2×2 MIMO in the downlink.

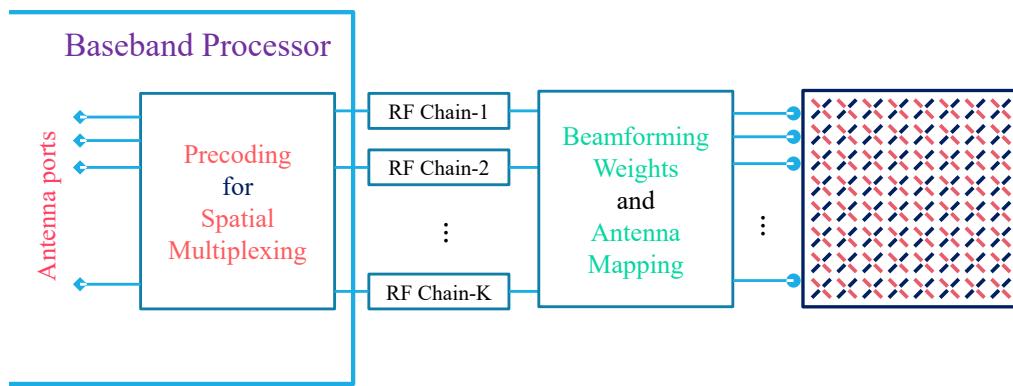


Figure 12.4: One to one mapping between antenna port and physical antenna element for 2×2 MIMO.

The figure 12.4 use an single cross-polar antennas. Antenna port 1000 (has both PDSCH payload and DMRS)is mapped on one physical antenna element while antenna port 1001 is mapped onto another physical antenna element. Both these antenna elements are connected to RF chain respectively, resulting in transmission of 2 streams. Figure 12.5 shows port 1000 which include both the payload and the DMRS with configuration II, the pink shade depicts payload data while the blue shade shows the pilot. Port 1000 and port 1001 has payload and DMRS at same location(s), therefore only port 1000 is depicted.

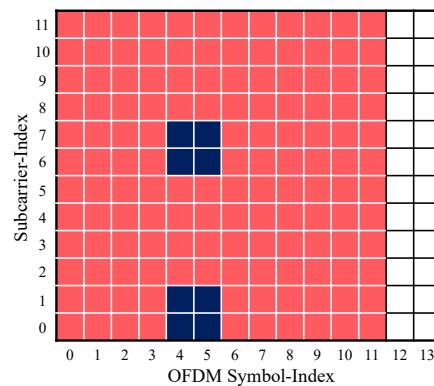


Figure 12.5: An example of port 1000 from 5G toolkit

12.2.2 | MIMO: Receiver

At UE the two parallel streams (for antenna port 1000 and antenna port 1001) are super imposed as shown in equation below:

$$Y = HX + N, \quad (12.1)$$

where H is 2×2 channel matrix, $X = [x_1, x_2]$, $Y = [y_1, y_2]$, where x_1 and x_2 are the payload/DMRS from antenna port 1000 and antenna port 1001 respectively. y_1 and y_2 are the superimposed reception of payload/DMRS from antenna port 1000 and antenna port 1001. Equation 12.1 is pictorially represented in figure 12.6 below.

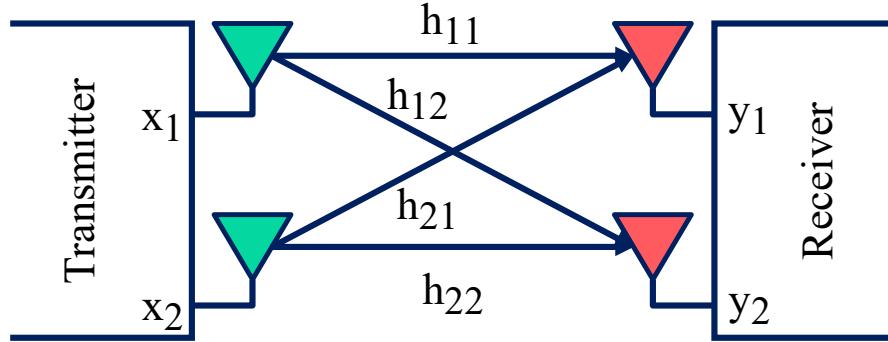


Figure 12.6: Representation of 2×2 MIMO system.

From figure 12.6, the received vectors y_1 and y_2 are given as:

$$y_1 = h_{11} \times x_1 + h_{12} \times x_2 \quad (12.2)$$

$$y_2 = h_{21} \times x_1 + h_{22} \times x_2 \quad (12.3)$$

Here the antenna(s) depicts two RF chain, which transmits two layers/ports of data.

12.3 | Results

The general simulation parameters are given in table 12.1 below:

Table 12.1: Simulation parameters and evaluation methodology

Parameters	Value
Carrier frequency (f_c)	1000 MHz
Bandwidth (B)	30 MHz
FFT size (N_{FFT})	1024
Subcarrier spacing (Δf)	30 KHz
Numbers of Resource Blocks (RBs) (N_{RB})	10
Numbers of Slots (N_s)	7 (PDSCH) + 1 (SSB)
Numbers of base station	1
PDSCH mapping type	“mapping type B”
maxLength (single or double DMRS)	“len1”
startSymbol (OFDM start symbol)	0
configurationType	“Configuration-type-1”
dmrsTypeAPosition	“pos2”
dmrsAdditionalPosition	“pos2”
rank	1/2
mcsIndex	0 to 28
mcsTable	“pdschTable1”
Transmitter-receiver separation	1m

The above parameters hold for every result unless otherwise specified. In this experiment, we will investigate the performance of two modes of MIMO: spatial diversity and spatial multiplexing. The

simulation methodology is exactly same as the one used in last tutorial-[11.17](#) which transmits the SSB first for time-frequency synchronization and CFO estimation. Followed by transmission of 7 slots of PDSCH for performance evaluation. The experiments though has been carried out for 1 or 2 layers.

12.3.1 | Spatial Multiplexing

Observation 1: *Spatial multiplexing enhances the throughput of wireless communication system. The same can be verified through hardware emulations.*

From table [12.2](#), it can be observed that for MCS indices 0 and 4 (QPSK modulation), the throughput and spectral efficiency are 8.75 Mbps, 22.414 Mbps, and 0.291 bits/sec/Hz, 0.747 bits/sec/Hz, respectively. The highest MCS index for which BLER is 0 is 15; beyond this index, the BLER starts to deteriorate. Additionally, the maximum MCS index for which throughput is non-zero is 24. Beyond this MCS index, the BLER is one resulting in zero throughput. Furthermore, it's important to understand that the channel is not very well conditioned due to the collocation of transmit and receive antennas, resulting in highly correlated channels.

Table 12.2: Performance of MIMO in spatial multiplexing mode for different MCS indices

MCS Index	Q_m	Code rate (r)	Throughput	Spectral Efficiency	BLER
0 (QPSK)	2	0.117	8.75 Mbps	0.291 bits/sec/Hz	0
4 (QPSK)	2	0.300	22.414 Mbps	0.747 bits/sec/Hz	0
10 (16 QAM)	4	0.332	50.19 Mbps	1.673 bits/sec/Hz	0
15 (16 QAM)	4	0.601	89.628 Mbps	2.9876 bits/sec/Hz	0
18 (64 QAM)	6	0.455	100.086 Mbps	3.336 bits/sec/Hz	0.02
20 (64 QAM)	6	0.553	117.488 Mbps	3.916 bits/sec/Hz	0.063
22 (64 QAM)	6	0.650	4.198 Mbps	0.139 bits/sec/Hz	0.971
23 (64 QAM)	6	0.702	4.098 Mbps	0.136 bits/sec/Hz	0.974
24 (64 QAM)	6	0.753	0 Mbps	0 bits/sec/Hz	1

12.3.2 | Spatial Diversity

Observation 2: *Spatial diversity enhances the reliability of wireless communication system. The same can be verified through hardware emulations.*

In spatial diversity mode, a single layer is communicated using multiple transmit antennas and/or multiple receive antennas. The experiment compares the spatial diversity performance of three systems. The first is a 2×2 MIMO system where one single layer is transmitted using two antennas and received using two antennas. The second is a MISO system where the single PDSCH layer is transmitted using two antennas but received using only one antenna. The third is a SIMO system with a single transmit antenna and two receive antennas. All antennas are co-located on a single 2×2 MIMO SDR without transmitting with precoding and no careful combining has been applied at the receiver. All systems outperform 1×1 SISO systems in terms of throughput and reliability [12.3](#). However, it is counter-intuitive that the MISO system outperforms the MIMO system. This could be due to the lack of transmit-precoding and receive combining or near-field effects. Further analysis is required to understand these aspects, and updates will be made in the future to achieve the expected results.

As observed from table [12.3](#), 2×2 MIMO with spatial diversity improves the reliability of the system as BLER is 0 till MCS index of 22. 22 is highest MCS index for which BLER is 0. For spatial diversity, even for MCS index of 28 the BLER is not equal to 1, thereby providing higher reliability.

12.4 | Useful Links

The following texts provides some useful links for further reading

- [Fundamentals of Massive MIMO networks](#)
- [MIMO in 5G Networks](#)

Table 12.3: Performance evaluation (Throughput (η , in Mbps), BER and BLER) for spatial diversity for SIMO (1×2), MISO (2×1) and MIMO (2×2) transmitting single layer for different MCS indices (I_{MCS}).

MCS Parameters			MIMO			MISO			SIMO		
I_{MCS}	Q_m	r	η	BER	BLER	η	BER	BLER	η	BER	BLER
0	2	0.117	4.44	0	0	4.44	0	0	4.44	0	0
4	2	0.300	11.20	0	0	11.20	0	0	11.20	0	0
10	4	0.332	24.65	0	0	24.65	0	0	24.65	0	0
15	4	0.601	44.81	0	0	44.81	0	0	44.81	0	0
22	6	0.650	73.53	0	0	73.53	0	0	73.53	0	0
27	6	0.888	36.88	0.002	0.63	94.26	7.4×10^{-6}	0.061	49.17	0.005	0.51
28	6	0.925	18.57	0.009	0.82	24.14	0.016	0.767	13.00	0.008	0.875

Note: η is system throughput in Mbps. I_{MCS} is MCS index used to compute the modulation order (Q_m) and code-rate (r).

- Link level simulations for MIMO systems with SVD based precoding and combining in 5G networks
- Downlink beam-management in 5G Networks
- AI-ML for CSI compression and reconstruction in 5G networks.

13 | Important System Parameters and their effect on the 5G Networks Performance

In previous chapters we understood the downlink data communication in 5G networks. 5G networks are designed to adapt to changing conditions and meet the quality of services requirements for diverse use-cases. The primary categorization of user equipment are enhanced mobile broadband (eMBB), ultra-reliable low latency communication (uRLLC), and massive machine type communication (mMTC). However, there many use-cases which are have intermediate QoS requirements to these use-cases such as vehicle to everything (V2X) networks, extended reality devices (XR) etc. We will focus primarily on main three use-cases of 5G in this tutorial.

These three use-cases has very different QoS requirements. The eMBB primarily focus on delivering very the data-rate or network/user-perceived throughput. Whereas, the uRLLC aims to deliver highly reliable communication reacting very quickly ensured by very low end to end latency. On the other hand, mMTC is designed to serve massive number of devices constrained in power, complexity and cost which can connect to the network simultaneously requiring small data communications with clement latency requirements.

This tutorial will introduce the users to some very important physical layer parameters which allows the network meet the requirements of all these use-cases. We will discuss these parameters in detail and demonstrate how these parameters influences the network performance. Furthermore, the tutorial will exhibit the fundamental limitations on selection of these parameters and their dependency of the other system aspects

13.1 | What is Flexible numerology?

5G is designed to support diverse use-cases and flexible numerology plays a crucial part in it. The large subcarrier spacing results in shorter sample period and hence short **OFDM** symbols periods and slot duration which are crucial for low latency.

Furthermore, the wider subcarrier spacing are robust to carrier frequency offsets and Doppler frequency offsets making them desirable for low cost receivers and high mobility scenarios. On the other hand, the low subcarrier spacing are particularly useful in rich scattering and large cell deployments. 5G networks supports, 7 numerology as shown in table 13.1 till release 18 for frequency range 1 (FR1, less than 5.4 GHz), frequency range 2 (FR2) and frequency range 2-2 (above 52.6 GHz).

Table 13.1: Supported transmission numerologies in 5G

μ	$2^\mu \times 15\text{kHz}$	Cyclic Prefix	Comments
0	15 kHz	Normal	FR1
1	30 kHz	Normal	FR1
2	60 kHz	Normal, extended	FR1/FR2
3	120 kHz	Normal	FR2
4	240 kHz	Normal	FR2 (only for SSB)
5	480 kHz	Normal	FR2-2
6	960 kHz	Normal	FR2-2

13.2 | What is Frame structure?

The frame structure is the organization of radio frames as a unit of time resource in 5G. Each radio unit consists of multiple slots which contains 14 OFDM symbols as shown in figure [?]. The number of slots in a frame depends in the subcarrier spacing (Δf)/numerology (μ).

$$N_{\text{slot}}^{\text{Frame}} = 2^\mu \times 10 \quad (13.1)$$

13.2.1 | Proof

The number of slots in frame changes with subcarrier spacing purely due to variation in sample duration with subcarrier spacing. The sample duration in 5G systems is

$$T_s = \frac{1}{\Delta f \times N_{\text{FFT}}} \quad (13.2)$$

where, the $\Delta f = 2^\mu \times 15000$ relates the subcarrier spacing to numerology. The slot consists of 14 symbols where each symbol contains L_{CP} cyclic prefix samples and N_{FFT} information samples resulting in slot duration ($T_{\text{slot}} = 14 \times (N_{\text{FFT}} + L_{\text{CP}}) \times T_s$) frame duration is 10 ms. Using both these information, the number of slots per frame can be computed as:

$$N_{\text{slot}}^{\text{Frame}} = \frac{10 \times 10^{-3}}{T_{\text{slot}}} \quad (13.3)$$

$$\begin{aligned} &= \frac{10 \times 10^{-3}}{14 \times (N_{\text{FFT}} + L_{\text{CP}}) \times T_s} \\ &= \frac{10 \times 10^{-3} \times \Delta f \times N_{\text{FFT}}}{14 \times (N_{\text{FFT}} + L_{\text{CP}})} \\ &= \frac{10 \times 10^{-3} \times 2^\mu \times 15000 \times N_{\text{FFT}}}{14 \times (N_{\text{FFT}} + L_{\text{CP}})} \end{aligned}$$

$$= 2^\mu \times 10 \times 10^{-3} \times \left(\frac{15000 \times N_{\text{FFT}}}{14 \times (N_{\text{FFT}} + L_{\text{CP}})} \right) \quad (13.4)$$

$$= 2^\mu \times 10 \times 10^{-3} \times \frac{1}{\bar{T}_{\text{slot}}} \quad (13.5)$$

$$= 2^\mu \times 10 \times 10^{-3} \times \frac{1}{10^{-3}} \quad (13.6)$$

$$= 2^\mu \times 10 \quad (13.6)$$

In equation 13.5, \bar{T}_{slot} is the slot duration for 15 kHz subcarrier spacing which is 1 ms.

Note: In eMBB, the transmission time interval (TTI) is typically equal to slot-duration. On the other hand, the same for uRLLC can be 2, 4 or 7 OFDM symbol. The TTI provides a time framework within which the scheduler operates, making dynamic decisions on resource allocation and scheduling based on the specific requirements of the network, services, and users. The relationship between TTI and the scheduler is crucial for optimizing the use of available resources, meeting quality of service objectives, and adapting to varying network conditions.

13.3 | Impact of parameters on network performance

This section will discuss all the physical layer parameters which influences the network's performance. These parameters are:

- System bandwidth
- Modulation order and code-rate (MCS)
- MIMO: Number of antennas at the transmitter and receiver
- Transmit power
- Subcarrier spacing

13.3.1 | Impact of bandwidth

The bandwidth represents the frequency resources available at the network for communication. The higher bandwidth offers increased resource elements (REs), where each RE is capable of carrying one symbol (denoting a bit of information based on the modulation order) worth of information, resulting in increases information rate. The transmission bandwidth is limited by the maximum bandwidth of the SDR. The higher bandwidth increases the complexity of decoding resulting in higher power consumption. However, the higher bandwidth often require large transport block length pushing the LDPC to operate in asymptotic limit which improve the error correction capability of the LDPC codes. The opposite phenomenon takes place for small resource allocations which degrades the performance of LDPC codec.

13.3.2 | Impact of Modulation Order and Code-rate

The modulation order and code rate are the most important parameters from the network adaptation perspective. These parameters are derived from MCS table using the modulation and code-rate (MCS) index. The parameter is computed by scheduler and configured to the UE either via RRC signalling or DCI using PDCCH. MCS-index helps the scheduler establish a trade-off between the data-rate and reliability based on the channel conditions.

The modulation order control the symbol mapping which maps a group of bits to a complex symbols. As mentioned in previous sub-section, one symbol is carried by a RE. Higher the modulation order more the number of bits mapped to a single RE, hence higher data-rate provided suitable channel conditions. However, if the propagation conditions are harsh higher modulation order might result in high block rates resulting in lower reliability.

On the other hand, the code-rate is controlled by rate-matcher (bit-selection) and LDPC channel codec. Both these modules lies with upper physical layer of physical downlink shared channel (PDSCH). Both these modules control the amount of redundancy, parity bits, introduced in the information bits systematically. Higher code-rate improves the network throughput but offers a poor error correction capability which might result in lower reliability.

13.3.3 | Impact of Transmit Power (P_t)

The transmit power boosts the signal to make it suitable for long-distance transmission. Each modulation order requires a certain minimum link quality (signal to noise ratio (SNR)) to decode the transmit data accurately. The SNR is defined by

$$\rho = \frac{P_t}{N_0} \quad (13.7)$$

where the power of the noise generated by the receiver is parameterized by,

$$N_0 = \frac{K_B * T * B}{N_F} \quad (13.8)$$

where the K_B is Boltzmann constant, T is temperature, B is bandwidth and N_F defines the noise figure that parameterize the quality of the receiver. Higher transmit power improves the SNR which improves the decodability of a data transmitted using MCS index. However, beyond a certain point increasing the transmit power doesn't improve the system performance until either the transmit encodes the data with higher the MCS index or receiver supports required noise figure and receives a strong signal.

Note: Transmitting high power without adapting the modulation order, code-rate and transmission-rank results in poor energy efficiency.

13.3.4 | Impact of MIMO

5G systems are designed to support multiple antenna at the transmitter and receiver. Multiple antennas can be used to transmit either multiple copies of the same data (spatial diversity) or transmit distinct data through each antenna (spatial multiplexing).

■ MIMO: Spatial Multiplexing:

- In spatial multiplexing, multiple data streams are transmitted simultaneously using multiple antennas at the transmitter. Each antenna sends a different stream of data, and the signals traverse multiple spatial paths to reach the antennas at the receiver.
- The receiver uses its set of antennas to capture and separate the individual data streams. By exploiting the spatial dimension, spatial multiplexing increases the overall data rate of the communication system.
- Spatial multiplexing is particularly effective in environments with favorable conditions, such as when there is minimal signal interference and a clear line of sight between the transmitter and receiver.

■ MIMO: Spatial Diversity:

- Spatial diversity, on the other hand, is focused on improving the reliability and robustness of wireless communication by exploiting multiple spatial paths. This is achieved by transmitting the same data over multiple antennas with the goal of increasing the chances that at least one of the signals will reach the receiver with sufficient quality.

- In spatial diversity, each antenna serves as an independent communication channel, and the receiver combines the signals received from different antennas. This helps mitigate the effects of fading, signal attenuation, and other forms of interference.
- Spatial diversity is particularly useful in environments where signal reflections, diffraction, and scattering are prevalent, leading to variations in signal strength at the receiver.

In summary, spatial multiplexing and spatial diversity are both techniques employed in MIMO systems to enhance wireless communication performance:

- **Spatial Multiplexing:** Increases data throughput by transmitting multiple data streams simultaneously over different spatial paths.
- **Spatial Diversity:** Improves reliability and robustness by using multiple antennas to transmit the same data over independent spatial paths, mitigating the effects of fading and signal variations.

Both techniques can be used together in a MIMO system to achieve even greater improvements in data rates and reliability, depending on the specific characteristics of the communication environment.

13.3.5 | Impact of subcarrier spacing

The subcarrier spacing is an important parameter for consideration in system design. For a constant bandwidth, the amount of time-frequency resources for communication doesn't change irrespective of the value of the subcarrier spacing. The higher subcarrier spacing reduces the number of subcarriers/resource elements (REs) available per OFDM symbols but increases the number of OFDM symbols available per frame proportionally. On the other hand, the number of REs per OFDM symbols reduces for smaller numerologies but small duration increases. However, the smaller numerology are more susceptible to carrier frequency offset and Doppler frequency offset. Hence, the network performance improves for devices with higher mobility or unstable clocks. However, such networks have small coverage footprints.

13.4 | Results

The general simulation parameters are given in table 13.2 below:

Table 13.2: Parameters for performance evaluations

Parameters	Value
Carrier frequency (f_c)	1000 MHz
Bandwidth (B)	30 MHz
FFT size (N_{FFT})	1024
subcarrier spacing (Δf)	30 KHz
Numbers of RBs (N_{RB})	85
Numbers of batches ($N_{\text{slot}}^{\text{PDSCH}}$)	7
Numbers of base station (N_{BS})	1
PDSCH mapping type	“mapping type B”
maxLength (single or double DMRS)	“len1”
startSymbol (OFDM start symbol)	0
configurationType	“Configuration-type-1”
dmrsTypeAPosition	“pos2”
dmrsAdditionalPosition	“pos2”
rank	1
mcsIndex	0
mcsTable	“pdschTable1”

The above parameters are general parameters and valid for all results unless specified.

Observation-1: *The network's throughput both improves as the bandwidth allocated to a UE increases.*

The emulation in this observation is carried out for transmitter gain of 0 dB and receiver gain of 60 dB, for transmitter receiver distance of 50 cm. The MCS index is set to 15. From table 13.3 it can be clearly observed as the bandwidth increases the throughput increases owing the transmission of higher numbers of Resource Blocks (RBs). In other words the higher bandwidth allows transmission of larger chunks of data. The numbers of RBs available at BS for data transmission for a specific system bandwidth is determined from *TimeFrequency5GParameters* module in **5G-Toolkit**, which returns the numbers of RBs for the input bandwidth.

Table 13.3: Performance analysis of PDSCH for different bandwidths.

Bandwidth (in MHz)	Numbers of Resource Blocks (RBs)	Throughput	BLER	BER
10	24	12.768 Mbps	0	0
15	38	20.174 Mbps	0	0
20	51	26.894 Mbps	0	0
30	78	41.244 Mbps	0	0

Observation-2: *The network's throughput increases with MCS-Index till the link-budget is adequate to decode the information symbols accurately and code-rate can correct the errors in the decoded information.*

From Table 13.4, for MCS index (I_{MCS}) 0 and 4 (both QPSK), the throughput increases due to the higher code rate, rising from 0.117 to 0.300. This increase allows the BS to reduce redundant bits, replacing them with information bits. Similarly, transitioning from $I_{MCS} = 4$ to $I_{MCS} = 10$ improves network performance by using higher modulation, allowing twice the bits to be packed in the same bandwidth. However, increasing the MCS index to 25 (from PDSCH table 4) leads to performance degradation due to an inadequate link budget to support very high modulation orders (Q_m) and code rates (r). This results in the receiver being unable to decode the densely packed information, resulting in 0 Mbps throughput.

Table 13.4: KPIs for different mcs index in PDSCH.

MCS Parameters			Throughput	Spectral Efficiency	BLER
I_{MCS}	Q_m	r	η	r/B	p_B
0 (QPSK)	2	0.117	4.438 Mbps	0.1479 bits/sec/Hz	0
4 (QPSK)	2	0.300	11.2 Mbps	0.373 bits/sec/Hz	0
10 (16 QAM)	4	0.332	24.654 Mbps	0.821 bits/sec/Hz	0
24 (64 QAM)	6	0.753	0 Mbps	0 bits/sec/Hz	1

Observation-3: *Increasing transmitter power improves networks performance for coverage constrained UEs. For UEs with strong links, increasing transmit power doesn't offer any performance gains instead it degrades the energy efficiency.*

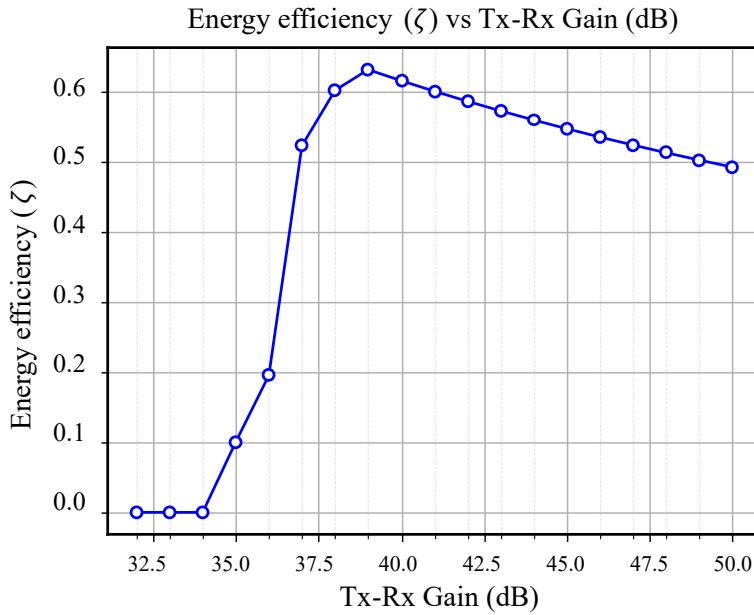
The emulations carried out below considers mcs index 9. The tables below shows the throughput and BLER for fixed receiver gain of 60 dB. For distance of 1m, -20 dB of transmitter gain is required to get BLER of 0, whereas for shorter distance of 0.1m, -30 dB of transmitter gain is enough for BLER to be 0. Since, the throughput remains constant despite the increase in transmit power the energy efficiency reduces. The energy efficiency is given by,

$$\begin{aligned}\text{Energy Efficiency}(\zeta) &= \frac{\text{Throughput}}{\text{Total Power}} \\ &= \kappa \times \frac{\text{Throughput}}{\text{Tx-Rx Gain}}\end{aligned}$$

where, Tx-Rx Gain (G_o) = Transmitter PA gain (G_{tx}) + Receiver LNA gain (G_{rx}). The EE vs Tx-Rx Gain curve is shown in figure 13.1. The EE for total power of 30 dB, 40 dB and 50 dB are 0.1095, 0.0876, 0.073. Thus, despite increasing the transmission power, EE decreases thus wasting the useful power.

Table 13.5: Energy efficiency (ζ) for different transmitter and receiver gain for $I_{MCS} = 9$.

Transmitter gain	Receiver gain	Tx-Rx gain	Distance (m)	Throughput	BLER	ζ
-28 dB	60 dB	32 dB	1 m	0 Mbps	1	0
-27 dB	60 dB	33 dB	1 m	0 Mbps	1	0
-26 dB	60 dB	34 dB	1 m	0 Mbps	1	0
-25 dB	60 dB	35 dB	1 m	3.521 Mbps	0.857	0.1006
-24 dB	60 dB	36 dB	1 m	7.044 Mbps	0.714	0.195
-23 dB	60 dB	37 dB	1 m	19.371 Mbps	0.214	0.523
-22 dB	60 dB	38 dB	1 m	22.893 Mbps	0.0714	0.587
-21 dB	60 dB	39 dB	1 m	24.654 Mbps	0	0.616
-20 dB	60 dB	40 dB	1 m	24.654 Mbps	0	0.616
-19 dB	60 dB	41 dB	1 m	24.654 Mbps	0	0.601
-15 dB	60 dB	45 dB	1 m	24.654 Mbps	0	0.493
-10 dB	60 dB	50 dB	1 m	24.654 Mbps	0	0.493
-35 dB	60 dB	25 dB	0.1 m	0 Mbps	1	0
-34 dB	60 dB	26 dB	0.1 m	17.61 Mbps	0.285	0.677
-33 dB	60 dB	27 dB	0.1 m	22.893 Mbps	0.071	0.847
-31 dB	60 dB	29 dB	0.1 m	22.893 Mbps	0.071	0.789
-30 dB	60 dB	30 dB	0.1 m	24.654 Mbps	0	0.821

**Figure 13.1:** Energy Efficiency vs Total Power at distance of 1m.

Observation-4: Higher order MIMO improves the reliability when configured for **spatial diversity** and offers higher data-rate when configured for **spatial multiplexing**. MIMO can be used either for providing higher reliability (spatial diversity) or higher data rate (spatial multiplexing).

In table 13.6 the key performance indicators (KPIs) of SISO (1 X 1) and MIMO (2 X 2) with spatial multiplexing (SM) and spatial diversity (SD) are given. It can be noted from table 13.6 that as MCS index increases, the throughput and BLER improves and then starts to deteriorates after certain MCS index. For SISO the maximum throughput is achieved at MCS index of 20, afterwards the BLER starts to increase and becomes 1 at MCS index of 24. For SM MIMO, the BLER becomes 1 at MCS index of 24.

For SM MIMO the peak throughput of 117.4 Mbps is achieved thus providing higher throughput, whereas for SD MIMO the BLER does not go to 1 even for MCS index of 28 thus providing higher reliability.

Table 13.6: Performance comparison of SISO, MIMO(SM) and MIMO(SD) systems.

MCS Parameters			SISO		MIMO(SM)		MIMO(SD)	
I_{MCS}	Q_m	r	Throughput	BLER	Throughput	BLER	Throughput	BLER
0	2	0.117	4.438 Mbps	0	8.75 Mbps	0	4.438 Mbps	0
20	6	0.553	62.748 Mbps	0	117.488 Mbps	0.063	62.748 Mbps	0
23	6	0.702	3.758 Mbps	0.952	4.098 Mbps	0.974	78.918 Mbps	0
24	6	0.753	0 Mbps	1	0 Mbps	1	84.294 Mbps	0
28	6	0.925	0 Mbps	1	0 Mbps	1	18.57 Mbps	0.821

Observation-5: Higher subcarrier spacing (Δf) improves the network performance for lower end receiver and mobile users.

The above experiment is carried out for mcs index 4 with a distance between transmitter and receiver SDR at 1m. The size of FFT for 15 KHz and 30 KHz are 2048 and 1024 respectively. As observed from table 13.7 as the subcarrier spacing increases there is small change in throughput and spectral efficiency. The numbers of RBs is determined from TimeFrequency5GParameters, which returns the numbers of RBs for each configuration.

Table 13.7: Variation on Performance with subcarrier spacing (Δf) for MCS index of 4.

(Δf)	Numbers of RBs	Throughput	Spectral Efficiency	BER	BLER
15 KHz	160	10.535 Mbps	0.351 bits/sec/Hz	0	0
30 KHz	78	10.304 Mbps	0.343 bits/sec/Hz	0	0

13.5 | Exercise

1. Prove the following relation for the number of slots in a frame.

$$N_{\text{slot}}^{\text{Frame}} = 2^\mu \times 10$$

2. Prove that the duration of the slot for any numerology μ , with normal cyclic prefix, is 2^μ fraction of 1 ms where T_s is the sample duration.

$$\sum_{i=0}^{i=13} (N_{\text{FFT}} + L_{\text{CP}}[i]) \times T_s = 2^\mu \times 10^{-3}$$

Hint: Use 3GPP TS 38.211 section 5.3.1 for length of the cyclic prefix (CP).

3. Find the throughput and block-rate error (BLER) for the following modulation-order and code-rates.

- [a] MCS-index = 0, SNR = 10 dB.
- [b] MCS-index = 5, SNR = 10 dB.
- [c] MCS-index = 10, SNR = 10 dB.
- [d] MCS-index = 15, SNR = 10 dB.
- [e] MCS-index = 20, SNR = 10 dB.
- [f] MCS-index = 24, SNR = 10 dB.
- [g] MCS-index = 28, SNR = 10 dB.

4. Find the best modulation-order and code-rate (MCS) index and the corresponding throughput for the following system parameters,

- [a] $P_t = 43 \text{ dBm}$, bandwidth = 10MHz at temperature = 300°C .

[b] $P_t = 53\text{dBm}$, bandwidth = 80MHz at temperature = 300°C.

[c] $P_t = 46\text{dBm}$, bandwidth = 2MHz at temperature = 300°C.

5. (MIMO) Find the best rank, modulation-order and code-rate (MCS) index and the corresponding throughput for the following system parameters,

[a] $P_t = 43\text{dBm}$, bandwidth = 10MHz at temperature = 300°C.

[b] $P_t = 53\text{dBm}$, bandwidth = 80MHz at temperature = 300°C.

[c] $P_t = 46\text{dBm}$, bandwidth = 2MHz at temperature = 300°C.

14 | References

- [1] TS 38.211 3rd Generation Partnership Project. Physical channels and modulation (Release 17). *Technical Specification Group Radio Access Network*, Version(v17.5.0):35–52, 2023-06.
- [2] Paul Bezner. 5G LDPC Codes. Institute of Telecommunications, University of Stuttgart, Germany, Apr 2024. Webdemo.
- [3] L.H. Crockett, D. Northcote, and R.W. Stewart. *Software Defined Radio with Zynq Ultrascale+ RFSoC*. Strathclyde Academic Media, 2023.
- [4] Douglas H Morais. *Key 5G physical layer technologies*. Springer, 2020.