Machine Learning Based Modulation and Coding Scheme Selection

Master's Thesis

by

Michel Chamat (mi0037ch-s@student.lu.se)
Dimitrios Bersi Kodra (be0078ko-s@student.lu.se)

Department of Electrical and Information Technology
LTH, Lund University

June 2019

LUND UNIVERSITY

ERICSSON

Abstract

In wireless communication, resources like bandwidth and energy are scarce and extremely valuable, any system should serve as many users as possible while preserving high Quality of Service (QoS) for the best user experience. Accordingly, the Base Station (BS) has the responsibility to optimally schedule its resources to the users based on the available information. Consequently, the whole process of scheduling is truly demanding and requires high complex calculations from the overall system. Hence, the request of more sophisticated and effective methods is substantial in order to minimize the challenges of scheduling.

This master’s thesis focuses on the Modulation and Coding Scheme (MCS) selection in a Time Division Duplex (TDD) based mobile network. The main objective is the simplification and optimization of the downlink process at the base station by predicting the MCS index for a single User Equipment (UE), using Machine Learning (ML). The developed machine learning algorithms is in accordance with the LTE-Advanced Pro (release 12, 13,14) lookup tables and is based on similar parameters. For a given frame, this thesis targets predicting the MCS index of future subframes. Thus, the resource allocation process for independent users is becoming quicker and easier for the BS. The results are based on laboratory measurements at Ericsson, where the collection of data logs for several stationary UEs, occurred on a network testing environment and their different cell characteristics investigated thoroughly.

Concluding, the accuracy level which the ML classification algorithm achieved was approximately 50 percent. Therefore, the prediction accuracy can be described as sufficient for the BS to decrease the computation complexity and energy consumption during the downlink process. The data logs that the project took into account cannot be generalized for real-time scenarios as it is explained in detail finally.

Popular Science Summary

Machine Learning (ML), as part of Artificial Intelligence (AI), is the most promising candidate to improve to a whole new perspective our modern world. Hundreds million of dollars are being invested worldwide on this technology, either by private companies or even by entire nations, to make all the existing systems smarter, more efficient and low-cost. An expected economic growth of approximately $40 billion by 2025, from $1.29 billion in 2016, is anticipated by the global ML market [1].

The best way to define ML is given by Arthur Samuel in 1959 and is the ability that computers learn without being explicitly programmed [2]. Nowadays, the majority of Internet users are handling and operating, in one way or another, several ML algorithms, and possibly most of them without being experts in the subject. Google, Netflix and YouTube are some examples of well-known technology giants that are deploying ML methods on their platforms, and we are using them in our everyday life. Whenever someone is typing for a new TV show or a song in a search engine, an ML algorithm is performing a historical exploration in the vast databases in order to, either suggest or find the matching word of the user's wish. Moreover, ML methods are expanded and deployed by various industries, as the vehicular, medical, retail, marketing etc. Therefore, ML is going to upgrade the lives of people in a easier, and more beneficial way than it used to be.

In this thesis, the focus is on integrating this technology in telecommunication, specifically at the Base Station (BS), to decrease the scheduling process and increase the quality of service for the users. Based on the users' behavior, the BS will be able to allocate its resource in an efficient way. Hence, under ML usage, complexity and energy consumption are decreased, while still providing the users with their required resources.

Acknowledgment

This master's thesis would not exist without the continuous support, guidance and feedback of Harish Venkatraman Bhat, our supervisor at Ericsson AB, throughout this thesis. Moreover, we would like to give special thanks to Prof. Fredrik Tufvesson, our supervisor at Lund University, Fredrik Russek for examining our thesis and the colleagues in Ericsson for the support they provided during our research. Furthermore, we would like to show our gratitude to Lars Thorsson for giving us the opportunity to work this master thesis at Ericsson.

At last, we want to express our gratitude one more time to our families and friends for their love, support and encouragement during this whole period of work and studies.

Preface

This Master’s thesis work was conducted entirely at Ericsson AB in Lund, between January and June 2019. The involvement of the authors was equally the same, in all the different subjects of the project across the overall period. The discussion of the process and the various difficulties which were faced in the thesis were reviewed by the two of them, mainly during short, or not, coffee breaks all along the work period.

M. Chamat and D.B. Kodra

Table of Contents

List of acronyms

3GPP	3rd Generation Partnership Project
5G	5th Generation
AI	Artificial Intelligence
AMC	Adaptive Modulation and Coding
BS	Base Station
CQI	Channel Quality Indicator
CSI	Channel State Information
DL	Downlink
ELM	Extreme Learning Machines
eNodeB	Evolved Node B
FD	Full Duplex
FDD	Frequency Division Duplex
FSK	Frequency Shift Keying
GP	Guard Period
HARQ	Hybrid Automatic-Repeat-Request (ARQ)
HD	Half Duplex
HCA	Hierarchical Cluster Analysis
IoT	Internet of Things
k-NN	k-Nearest Neighbor
LDA	Linear Discriminant Analysis
LLE	Locally-Linear Embedding
LTE	Long Term Evolution
MAP	Maximum A Posteriori
MCS	Modulation and Coding Scheme
MDP	Markov Decision Process
MIMO	Multiple-Input-Multiple-Output
ML	Machine Learning
MU-MIMO	Multi User Multiple-Input-Multiple-Output
NDF	New Data Frame
NN	Neural Networks
NR	New Radio
NTE	Network Testing Environment
OFDM	Orthogonal Frequency Division Multiplexing
PAM	Pulse Amplitude Modulation
PCA	Principal Component Analysis
PSK	Phase Shift Keying
QAM	Quadrature Amplitude Modulation
QoS	Quality of Service
RB	Resource Block

RE	Resource Element
RL	Reinforcement Learning
SINR	Signal to Interference and Noise Ratio
SVM	Support Vector Machines
TDD	Time Division Duplex
t-SNE	t-distributed Stochastic Neighbor Embedding
UE	User Equipment
UL	Uplink
WCDMA	Wideband Code Division Multiple Access

List of figures

List of tables

1. Introduction

Nowadays, mobile communication networks are evolving in a tremendous way, constantly increasing their supported bit rates but also the complexity of the network in each upgraded version. A suitable example is the recent Massive Multiple-Input-Multiple-Output (MIMO) technology, which is boosting capacity and throughput in significant rates, while raising the BS processing complexity. Consequently, the need for more intelligent processing in the base station is essential and necessary. An efficient way to approach this problem would be the addition of the capability to predict, in a precise way, the optimal throughput at the Evolved Node B (eNodeB). This tactic would make the scheduling for a single or group of user equipment much more efficient in time and energy perspective.

The scheduler, located at the BS, is responsible for resource allocation in the downlink. Based on the received information from each UE in the uplink, the Channel Quality Indicator (CQI) is used to determine the user's downlink Modulation Coding Scheme (MCS) using a lookup table. With the continuous advance in technology, both at the user and base station sides, the modulation schemes are increasing and the MCS table is growing, reaching 31 distinctive settings in Long Term Evolution (LTE), compared to 15 in Wideband Code Division Multiple Access (WCDMA). Furthermore, with the increasing number of UEs, especially in 5G where the estimated number is expected to be up to 1 billion subscriptions reaching 2023 [3], a base station will have to serve multiple mobile terminals simultaneously and accurately.

The current scheduler, using the MCS lookup table, is relatively accurate but it only gives the MCS for the next subframe, given information at each frame and subframe. Therefore, to decrease the load on the scheduler, Machine Learning (ML) can potentially improve the performance of the scheduler by estimating and predicting the future MCS while taking into consideration the same user's information. Hence, ML algorithms, applied at the eNodeB, can achieve the prediction capability which we are looking for, while focusing on decreasing the complexity level.

1.1. Background and Motivation

The property of channel reciprocity can be used for time division duplex (TDD) based systems using channel state information (CSI). However, the computational complexity can increase in 5G with the expected use of massive MIMO, and some previous works suggest CSI based beamforming to improve the signal transmissions and energy efficiency of the system.

To predict the MCS, ML can be used at the base station and the training and uplink data can be used to simplify the process for the scheduler remarkably. Resulting this way, in faster and more accurate predictions of MCS.

Moreover, an optimal MCS improves the throughput and can be used by the content provider to dynamically adjust the quality of service. Firstly, this feature can improve the efficiency of the base station, and subsequently, the scheduling of the various users which are operating inside the cell controlled by the BS.

In addition, the motivational reason of using ML techniques in our work is not only because of its rapid growth in usage perspective by the researching community, but rather by cause of the increasing capabilities and the potential of implementing different ways of learning into various situations.

1.2. Purpose and Aims

This Master's thesis is focusing on the MCS selection in a cellular system. The main aim is to simplify and optimize the downlink process at the BS for a single UE. Moreover, ML will be used to predict the optimal MCS for this user. The model will take into consideration the training data and the continuous flow of uplink data aiming to determine the channel parameters for the UE. More specifically, our project investigates the capability of predicting an accurate MCS index for independent users while the base station is receiving the uplink feedback from the different UEs across its cell territory. The accuracy of this MCS selection has to be high enough, so the resource allocation can be improved and the overall scheduling process at the BS enhances in energy and speed perspective.

The topic of this thesis has not been found in other works in the engineering community, although similar works tried to explore the advantages of machine learning in mobile communication networks, as in [4]-[8]. Furthermore, those works are using different methods of ML, implementing various algorithms like Support Vector Machines (SVM), k-

Nearest Neighbors (k-NN), or even Principal Component Analysis (PCA) and Reinforcement Learning (RL) which are unsupervised methods, in contrast to our approach of examining the case MCS selection in LTE systems. Thus, our thesis, can be characterized as the continuation that [8] is proposing as we are targeting equivalent objectives, even though multiclass Neural Network learning is being implemented instead of Reinforcement Learning.

The main questions which this thesis is going to research thoroughly and try to answer in the best possible way are:

1. How to predict an MCS index for future transmission?
2. How accurate is the prediction of the MCS index selection of our ML algorithm?
3. What is the future work that could be done to upgrade this ML model?

However, implementing machine learning at the base station will be the main challenge as no similar work was done before. This includes continuous data training and a relatively accurate MCS index prediction. Therefore, another challenge will be minimizing the complexity of the system while getting accurate results. Optimization will be based on some available models already in use, and if necessary, some new ones. The number of users will increase gradually, according to the accuracy of the results.

1.3. Approach and Methodology

This section describes the methods that the thesis will be based on, firstly the generation and after that the capturing of several training sequences using a network testing environment so that we can use these sequences as input to the simulation model. Additionally, the input of the data sequences in combination with the ML algorithm will produce the MCS selection decision for the specific user that it is required. Afterwards, the decision output of the model will be repeatedly fed at the training database in a closed loop form process. Accordingly, MATLAB and Python are going to be our main tools for simulating and testing the ML algorithms and channel conditions. Also, a professional network simulator, provided by the company is used for the previous mentioned generation and capturing of the data sequences. Simplifying our main goal, a single MCS selection must be estimated successfully. Furthermore, for the training of machine learning model, Python will be used, providing it in this way with necessary parameters which will be used at the decision unit and are explained in more detail later on the section 5.2. Various simulations for testing and measuring the accuracy of the model for the user equipment were done in a network simulation laboratory.

The overall system model which will be used in our project and will be explained in detail piece by piece on the following sections is depicted in Fig. 1.

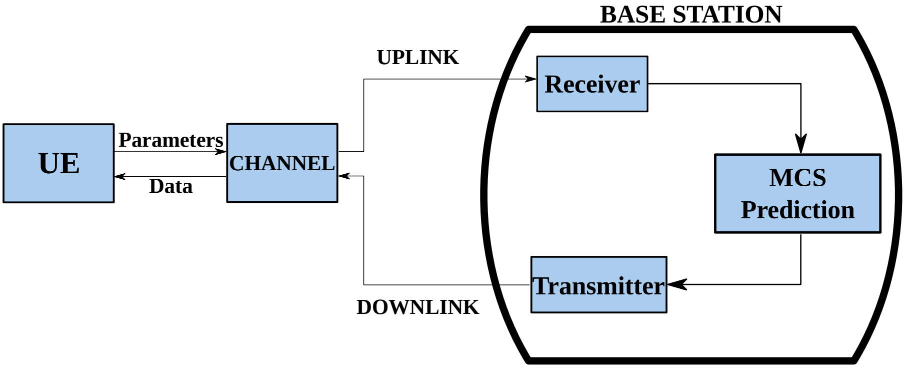

Fig. 1. Generic block diagram of the machine learning model.

1.4. Previous Work

In our master's thesis, the main goal is to investigate the prediction accuracy of our ML algorithm. Related work to our goal, has been done by other researchers in [4]-[8]. Although, all the authors chose different approaches and ML algorithms for their independent problems that had to examine.

In [4][5], the authors are using SVM algorithms to explore the capabilities of them in several scenarios and various alternative parameters to consider. The channel and modulation selection are implemented by the SVM method for cognitive radio [4], and an online Adaptive Modulation and Coding (AMC) scheme that operates in realistic conditions for different channel parameters [5] is further inspected. In [6] [7] the authors are questioning the usage of machine learning in MIMO-OFDM systems and how useful they can become for increasing SNR ordering and average throughput. The methods of k-NN, and a hybrid model of Deep Neural Network with Principal Component Analysis (PCA) are used in [6] and [7], respectively. Finally, in [8] the creators are investigating the AMC selection in LTE systems with purpose to show how inaccurate are the feedbacks and the MCS selection on channel qualities when they are implemented under a real-time model. Moreover, Reinforcement Learning (RL) is applied under the Markov Decision Process (MDP) method, aiming to decrease the channel prediction errors of link adaptation performance.

Consequently, in our project we are trying to explore different channel parameters to successfully predict the MCS index in the future framework using a multi-class NN algorithm. A target which has not been examined by other researchers, but it has been proposed in some extend in [8].

1.5. Limitations

The focus of this project is providing a more efficient algorithmic method for MCS selection than the existing one in LTE. Besides, there are many limitations on the overall system that must be considered before a final decision is made. This thesis is restricted by the nature of the data logs for the training of the model, because of the unavailability of real-time data logs. Thus, our work is limited and cannot be expanded on many realistic scenarios as the data comes from network testing environment of the laboratory. Another limitation that was critically influencing the scope of our work was the stationary position of the simulated UEs inside the laboratory environment. This immobile behavior of the users restricted the overall results of the thesis, as moving scenarios were excluded from the project by cause of insufficient data measurements. Moreover, a significant challenge which we had to face was the time extend of our research which restricted the thesis investigation to the case of Neural Network Multi-Class algorithm exclusively.

2. Technical Background

Some concepts of wireless communication systems used in this thesis will be explained in this section. They are also applied in cellular network systems, as LTE and NR, to transmit data between the eNodeB and UE.

Firstly, the fundamental components which have been used, will be further explained in the following sections.

2.1. Resource Block

In LTE, the shared frequency and time resources are combined to form the resource element (RE) and resource block (RB), as shown in Fig. 2.

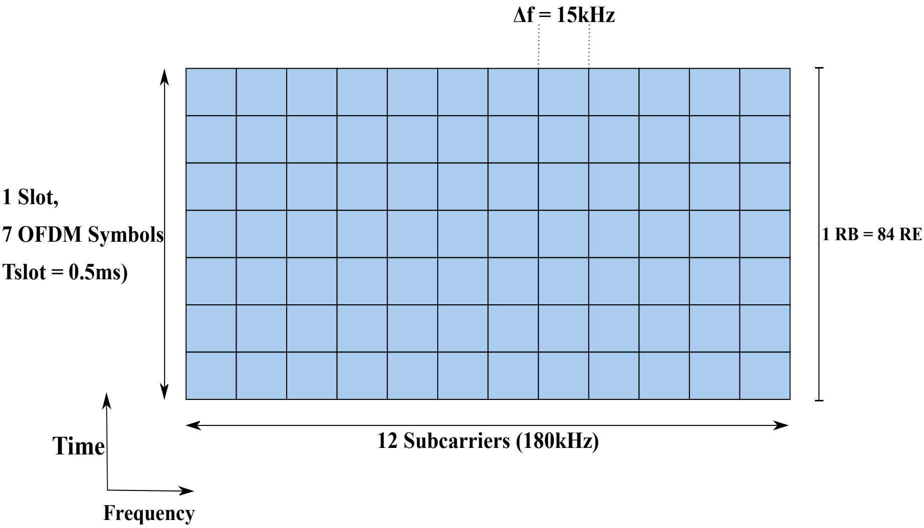

Fig. 2. Time and frequency resources in one resource block.

One RB is spread over 1 slot of 6 or 7 OFDM symbols in time according to their cyclic prefix, as shown in the following section, and 12 subcarriers in frequency domain. Thus, it is 180 kHz wide in frequency and 0.5 ms long in time [9].

For a UE to send or receive data, it should be allocated in one or more physical RBs. This process is called scheduling, which will be explained in section 4.2.

2.2. Duplex Tranmission

In mobile communication systems, the UE and BS exchange information using duplex transmission. Duplexing refers to data transmission in both directions, from the BS to the UE and vice versa. Two types of duplex transmission are available: Half Duplex (HD) and Full Duplex (FD).

FD enables both sides to transmit simultaneously using the same link, while in HD, transmission is only in one direction at a time. Both types are used in several applications, but HD is less complicated and requires less resources, as the same resource can be used in both directions while switching between the transmitter and receiver at given periods.

Time Division Duplex (TDD) and Frequency Division Duplex (FDD) are two duplexing methods used in cellular networks. TDD is HD based as the time domain is shared, but with small time intervals it can emulate FD. However, FDD is always FD given that at the equipment, BS or user side, the uplink and downlink frequencies are predefined. Chapter 4 gives additional information about both methods.

In LTE, channel information at both the UE and the BS is essential to establish a good communication channel and to use the resource blocks effectively. A brief explanation of the two links is available in the following subsections, as the work in this thesis was limited to the CQI only.

CQI, as its name implies, indicates if the channel has a good or bad quality based on the SINR. For a high CQI, a good channel is available and more RBs can be used, when compared to low CQI. More details can be found in section 4.3.

2.2.1. Uplink

The transmission from the UE to the BS is called uplink (UL). The data sent from each UE includes information about the channel condition and status of the received data. In addition to the essential CQI report, the UE sends reference signals to help the BS estimate the channel state. Furthermore, in the case of an erroneous packet, the UE can request retransmission from the BS. Finally, the UE can request RB allocation from the BS for UL and/or DL transmissions.

2.2.2. Downlink

Downlink (DL) is the transmission from the BS to one or more UEs. In addition to the information gathered from the user's UL, the BS sends reference signals to provide the connected UEs with channel conditions. Also, the base station sends control signals to provide specified UL slots for each of the UEs. Further technical details and explanation can be found in [10] and [11].

2.3. Coding Scheme

In wireless communications, several methods are used to increase the robustness of the signal, channel coding is one of these methods. Turbo codes and convolutional codes are some popular channel code examples used in wireless systems.

An important factor which has to be explained is Shannon's limit in capacity, that cannot be surpassed as it is a natural barrier. Although with creative and inventive coding methods, the bandwidth efficiency of some schemes came closer to this limit, as shown in Fig. 3.

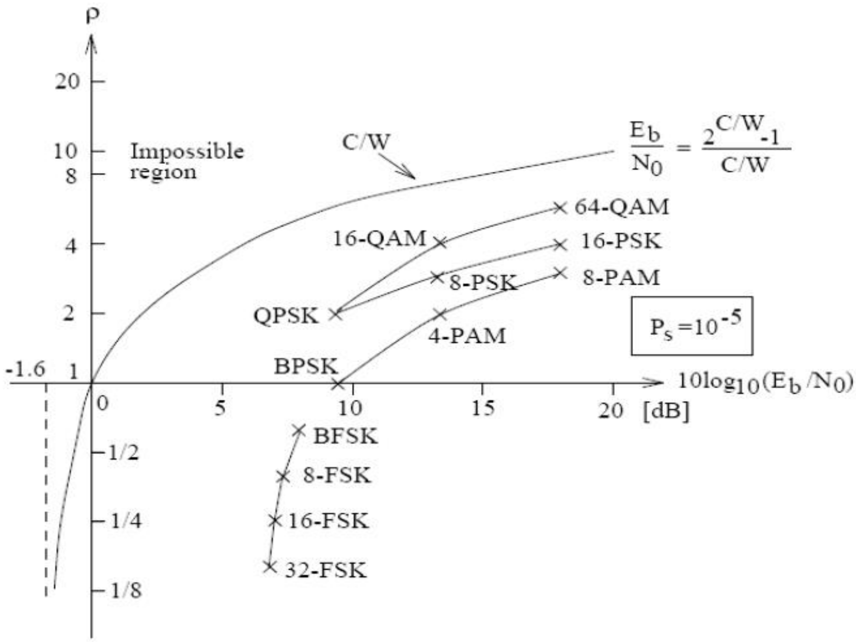

Fig. 3. Comparison of bandwidth efficiency ρ and SNR E_b/N_0 [12].

Where the Signal to Noise Ratio is denoted as E_b/N_0, C/W is the maximum possible bandwidth efficiency for any scheme, and QAM, PAM, PSK, FSK are modulation schemes that are defined at the list of acronyms, page viii. Moreover, coding is the most efficient way to detect and/or correct various bit errors which were transmitted through a wireless communication channel. Accordingly, the error detection and correction are performed by the decoder in the receiver stage. On the transmitter side, the modulated signal constellation points will increase their spatial distance from each other helping thusly, the receiver into distinguishing them easier than without the coding part. [13] Additionally, the code rate is defined as the proportion of the data stream which is useful (non-redundant), in other words, the code rate shows the ratio of the number of input information bits to the overall number of transmitted code symbols [14].

2.4. Modulation Scheme

Modulation is the process of mapping digital data to analog signals, and it is done at the transmitter. At the receiver, demodulation follows the reverse route by extracting the digital data from the analog signals.

To transmit the coded symbols, output of section 2.1, the sequence of symbols is transformed into a complex one based on the modulation scheme [12], [13]. For LTE downlink, the supported modulation schemes are QPSK, 16QAM and 64 QAM, representing two, four and six bits per symbol respectively. Fig. 4 depict the constellation points for the LTE modulation schemes [11].

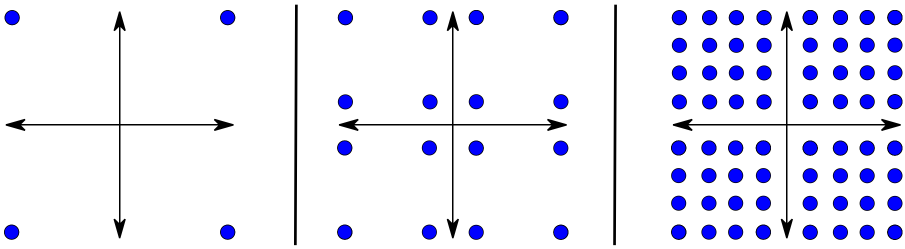

Fig. 4. Constellation diagrams of QPSK, 16QAM and 64QAM for LTE modulation schemes.

Increasing the modulation order results in a higher data rate, as more bits are transmitted. However, the distance between the constellation points gets smaller and the error rate will increase if the signal strength is not good enough [12].

In release 12 [15], 256QAM support was added for small-cell environments and it is not used in this thesis. This modulation scheme doubles the bit rate of 16QAM but is vulnerable to propagation loss and should be used for small distances only [11].

2.5. LTE Transmitter and Receiver

This thesis targets specific parts of LTE transmitter and receiver, while further explanation can be found in [16], [17].

The block diagram in Fig. 5 shows an overview of the system used in both UL and DL, where the coding, modulation and RB mapping blocks were introduced in section 2.3, 2.4, and 2.1 respectively.

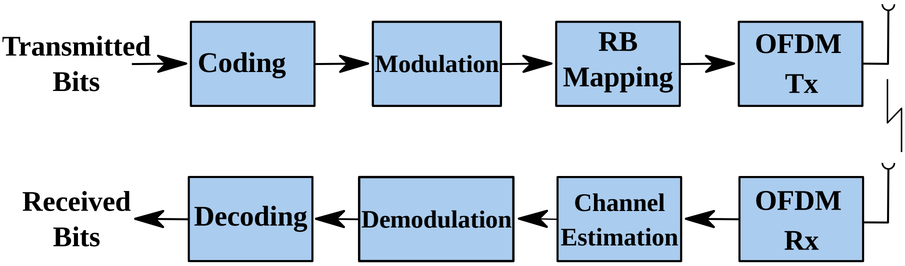

Fig. 5. Block diagram of LTE Transmitter and Receiver

To transmit data to or from the UE, the transmitted bits should be processed first as follows [10] [11]:

1. The sequence of bits will be coded to increase its robustness using a coding sequence.
2. Modulation is then used to map the coded bits to complex symbols and transform the data from digital to analog signal.
3. The number of resource blocks is assigned by the scheduler, and the analog signal is mapped to the allocated resources. The main focus of this thesis is this part of the transmitter, called scheduling.
4. In LTE, an OFDM transmitter is used, where orthogonal carrier frequencies are used for transmission. One of the main advantages of OFDM is spectral efficiency, as more symbols can be sent on a given bandwidth when compared to other techniques. This main block combines several sub-blocks like Precoding and Antenna Mapping that deals with transmission using the antennas, and frequency conversion which up-converts the frequency of the signal to the frequencies used in LTE [11].

After transmitting the signal over the wireless channel, introduced in section 2.6, the receiver processes the received signal using the reversed path of the transmitter. Using Fig. 5, the steps are as follows [10] [11]:

1. The OFDM receiver is used to down-convert the frequency of the signal and detect the information symbols of different carrier frequencies.
2. The wireless channel has a big effect on the signal that cannot be ignored for correct detection. Therefore, the variations in amplitude, phase and/or frequency caused by the channel are estimated and included in the demodulation.
3. Demodulation converts the received analog signal, including the channel estimation, and maps the detected symbols to the constellation diagram to extract the sequence of received coded bits.
4. The transmitter's coding sequence is used at the receiver to decode the received coded bits. Several error correction codes, as Turbo code, enable the receiver to find errors and correct them.

After transmitting a signal, the BS waits for a feedback from the UE included in the Hybrid Automatic Repeat Request (HARQ). The HARQ signal is sent in the UL, and it can either be [11]:

1. Acknowledgment (ACK), when the signal is correctly decoded. This indicates that the packet is received, and the next packet can be sent.
2. Negative acknowledgment (NACK), when signal is incorrectly decoded due to a high number of errors. A retransmission of the same packet is then requested by the UE.

2.6. Wireless Channel

The medium connecting the transmitter to the receiver in a wireless communication system is called wireless channel. The properties of this channel directly affect the propagating signal and should be taken into consideration at both the BS and UE to improve the transmission. These properties include, but are not limited to, noise level and interference level [13][18].

Many wireless systems, including LTE, require a high signal strength over noise and interference to provide a good reception quality. This factor is called Signal to Interference and Noise Ratio (SINR) and is directly related to the following factors:

1. The signal strength is related to transmission power allocated by the BS or UE. Some gains and losses are added at the transmitter due to the properties of the blocks, as the antenna gain and the filter loss.
2. The noise level is modelled as an Additive White Gaussian Noise, as it includes the random processes which are occurring naturally.
3. The interference level is caused by the neighboring BSs and UEs. The level varies in different scenarios as the number of close UEs and the strength of close BSs increases the interference on the received signal.

Moreover, the distance separating the UE from the BS, as well as the mobility of the UE in cellular systems, cause variation in the propagation mechanisms. Some propagation mechanisms are explained below, and they increase interference and loss in received signal strength [18]:

1. A signal propagating loses strength over distance due to its physical properties. This mechanism is called fading. When the UE and BS have a line of sight, the distance separating them will be the main loss factor, especially when the distance exceeds a certain range.
2. Multipath propagation is caused by surrounding interfering objects. Reflection, diffraction and scattering are the effects of this mechanism, and they result in time delays, frequency shifts, changing of the direction and creating multiple copies of the signal at different phases.

In this thesis, the UEs are considered stationary, and the wireless channel includes gaussian noise generated by Ericsson’s laboratory, interference from BSs and UEs, and fading caused by the distance between the BS and the nodes.

3. Machine Learning

This chapter introduces the main concept behind machine learning, starting from a definition to how it operates. Also, some important algorithms will be introduced to highlight the differences between the different ML types.

3.1. Concept

Alan Turing, mathematician and computer scientist, wrote one of the most important papers in the computer science field: Computing Machinery and Intelligence. Written in 1950, this paper highlighted the potentials of the machine called computer and Artificial Intelligence (AI). With the huge advancement in technology and the integration of several inventions in the modern computer, AI can now mimic cognitive functions of human and became very broad while including different fields like medical research, mathematics, statistics and engineering [19].

Another interesting definition from an engineering perspective has been given to ML by Tom Mitchell in 1997 [20]:

“A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E”.

This software algorithm is mainly used to predict an outcome based on input data. It iteratively learns from the data and does not follow a set of rules for detection. Additionally, ML uses statistical methods to enable machines to learn with experience from different scenarios in a supervised or unsupervised way. Deep learning is a major subset of ML, where the algorithms can learn by experience without any external interaction, i.e. an unsupervised learning method which enables the computers to learn on their own. In opposite, a supervised learning method requires teaching the model with the usage of training data, called samples, from a labelled data set. The relation between AI, ML and deep learning is depicted in Fig. 6.

In other words, under the supervised method, ML can be used to predict future events based on the set of data collected in real-time. An overview of the ML process is shown in Fig. 7 [21].

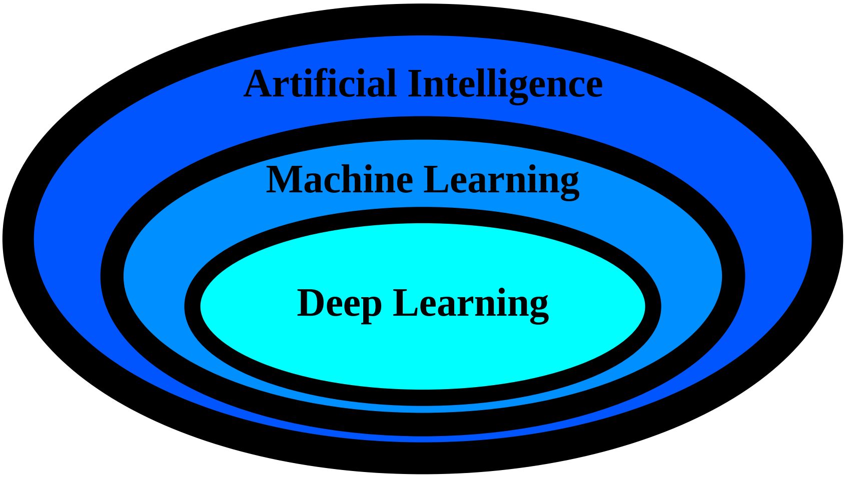

Fig. 6. Relation between AI, ML and Deep learning, from [21].

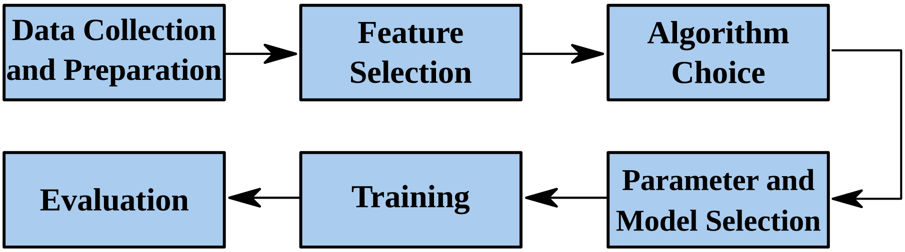

Fig. 7. Process of selection and evaluation of ML algorithms.

ML requires significant amount of collected data, preferably without noise, stored in a dataset that has a label for each column. The labels used to make the prediction are chosen in the Feature Selection, as every application requires different suitable attributes. To finalize the model, the algorithm is chosen so the parameters can be modified, as some algorithms require a specific set of parameters.

For the training procedure, the same dataset can be used to train and predict the result. This is done by splitting the dataset into two parts based on the size of the training set, as shown in Fig. 8.

The test size is an important parameter for every ML algorithm as well as it is controlling which samples are meant to be used for training and testing. Aiming to compensate in the best way the effects of overfitting and out-of-sample accuracies.

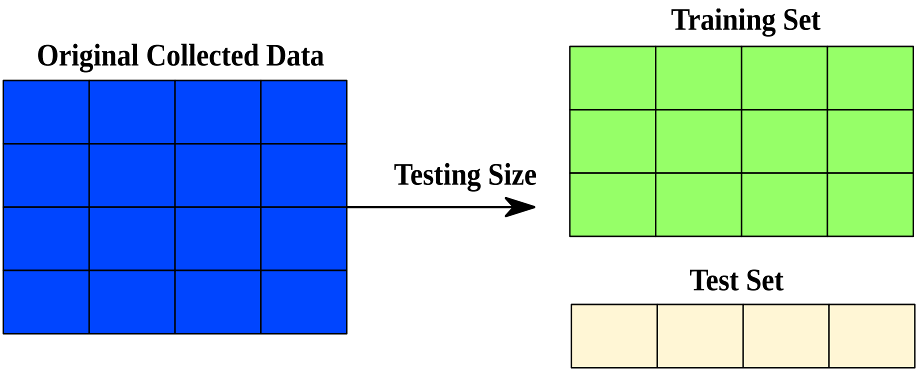

Fig. 8. Process of dataset splitting into training and testing sets.

This approach is realistic in real-world models, where no knowledge of the results increases the out of sample accuracy, which represent the prediction for an observation that it was not part of the testing data set. Also, the test set is highly dependent on the training set, so the test part should be added to the training part again to increase accuracy in a beneficial way.

In the evaluation phase, the predicted values coming from the output of the algorithm are compared to the test set equivalent, resulting in two parameters:

- The training accuracy is the percentage of correct assumption using the same dataset, and a high training accuracy results in an over-fit model, which does not correspond to general conditions.
- The out of sample accuracy rate is the percentage of correct prediction from outside the dataset. This rate should be high enough to generalize the model for real-time scenarios.

3.2. Algorithms

3.2.1. Supervise Learning

Several algorithms for different applications are available in the libraries of ML, accordingly the algorithms should be chosen carefully to get the finest results with minimum complexity. Commonly used algorithms include:

3.2.1.1. Regression

This supervised algorithm predicts continuous values based on the historical dataset by fitting the dataset into a curve with two sets of variables: the dependent and independent variables.

Based on the relation between the two sets, the regression type is chosen. Simple regression is based on one linear or non-linear variable and its predicted dependent variable by the independent one. While multiple regression is the extension of the simple type and is based on multiple variables, where the capability of the algorithm is increasing proportionally with the complexity of the model.

Simple regression is a fast and simple algorithm and can be used to predict continuous outputs based on a two-dimensional curve, although with several values, regression will become complicated, specifically with a multi-dimensional curve.

Moreover, the regression models are divided at two major subcategories in relation with their linearity:

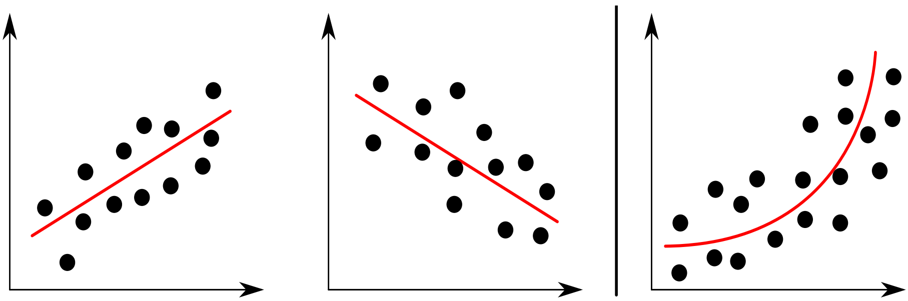

Fig. 9. Linear (first two) and non-linear fit of samples, from [21].

Linear regression is when the dependency of the output-input relation can be defined by a linear function. On the other hand, non-linear regression is defined by non-linear functions as exponentials, polynomials and quadratics.

3.2.1.2. Classification

In machine learning, the classification method is another supervised learning approach, which can be characterized as a means of categorizing, i.e. classifying, some unknown variables into a discrete set of classes. Classification attempts to learn the relationship between a set of feature variables and a target variable of interest [22].

In a classification problem, there exist two major constraints, which the fulfillment of both is sometimes unrealistic. At first, the set of classes are covering the whole possible output space, and secondly, the most important point about this set of ML algorithms is that the output value is discrete, with each example corresponding to precisely one class. There are occasions which examples might correspond to different classes or alternatively, examples that could not be classified in a particular input value, problem that the “fuzzy” classifiers are trying to solve. [21] [23]

Data classification has several applications in the modern industrial categories. Most importantly, many of the problems that developers are trying to solve can be expressed as association of a feature and a target variable. Moreover, this relation provides and extends the applicability for classification in a vast range of different scenarios. The most commonly used types of classifications algorithms in machine learning are: [24]

- Naïve Bayes: ML algorithms which take into consideration the principle of Maximum A Posteriori (MAP) for the classification of their problem.
- Decision trees: approach that splits the training set into distinct nodes, where one node can include one, most of or all of the different categories that the database can be divided.
- k-Nearest Neighbors (k-NN): a non-parametric classification algorithm that measures the distance of the unknown input from every other training example. [6]
- Logistic regression: a statistical ML technique which classifies the dataset records, based on the values of the input fields.
- Support Vector Machines (SVM): are among the most robust algorithms which are based in the maximization of the minimum distance from the decision line, i.e. separator, to the nearest example.

- Neural Networks (NN): are based upon the log likelihood function with respect to the network parameters, extending to the multiclass problem and ensuing as output a binary or N-ary result.

Less popular algorithms, as Extreme Learning Machines (ELM) and Linear Discriminant Analysis (LDA) are showing poor capabilities in evaluation [25], and high rate of misjudgment [26], respectively.

3.2.2. Unsupervised Learning

The most important difference between supervised and unsupervised learning is that the training data is unlabeled in a way that the algorithm does not have any specific information about the nature or the class of the inputs. The aim of this method focuses into finding clusters or similar inputs in the data, and consequently categorizing the unlabeled values in related groups. Moreover, the unsupervised approach is looking to determine the distribution of data within the input space, known as density estimation, or to project the data from a high-dimensional space down to lower and much simplistic systems for visualization. [22] Some of the most important ML unsupervised algorithms are shown in table 1 and in fig. 10 [23].

3.2.3. Semi Supervise Learning

Semi supervised learning, as the title describes, is the mixed version of the two different methods previously presented in sections 3.2.1 and 3.2.2. These algorithms can operate with partially labeled data and the rest of its data, the majority in most of the cases, with unlabeled ones. A good example of semi supervised algorithms is the reinforcement learning, which is a method called an agent in this context [23] and it observes the environment, selecting its best action according to a policy defined by previous situations. These actions are resulting to rewards or penalties, whether there is a positive or negative outcome, respectively. In [7] [8], semi supervised techniques are used to explore the capability of implementing ML in the mobile network of LTE systems.

Table 1. ML unsupervised algorithms categorized.

Clustering	**Visualization and dimensionality reduction**	**Association rule learning**
○ k-Means ○ Hierarchical Cluster Analysis (HCA) ○ Expectation Maximization	○ Principal Component Analysis (PCA) ○ Kernel PCA ○ Locally-Linear Embedding (LLE) ○ t-distributed Stochastic Neighbor Embedding (t-SNE)	○ Apriori ○ Eclat

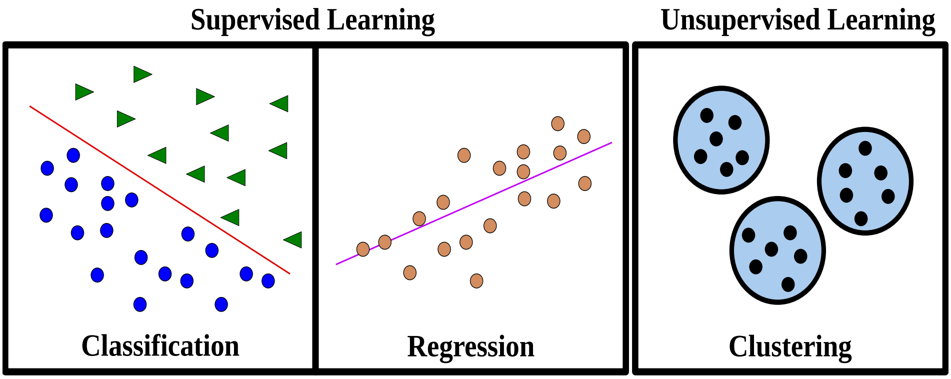

Fig. 10. Comparison between supervised and unsupervised learning [27].

3.3. Restraints and Drawbacks

As useful and more efficient ML is becoming in our lives, as dangerous it is if the drawbacks of this method are not taken into consideration. Following, the most significant restraints are presented and briefly explained. [21][23]

1. Massive Data Acquisition:
   ML algorithms require enormous amounts of data for training to perform their complex tasks that the developers designed them for. Also, a good quality restriction must be designated to get the best results.
2. Resources:
   Need of extended resources in time for the algorithm training to fulfill the high standards of accuracy and relevancy. Computational power for the operation of these large amount of data must be considered likewise.
3. Result Comprehension:
   The understanding of the ML outcomes can be tricky. Accordingly, the selection of the right algorithm should be done carefully to optimally interpret these results.
4. Error Sensitive:
   ML is hypersensitive to errors, for this reason, the training sets have to secure that the samples are unbiased and exclusive. However, it is difficult to achieve a good result and even more complicated to recognize the issue and correct it if the dataset includes a big amount of erroneous data.

4. Dynamic Resource Allocation

Mobile communication systems exhibit significant variation in the number of UEs in a given cell, and in their time-varying channel conditions. Moreover, with the expansion of mobile applications, different UE have different requirements. Thus, resource allocation became more complex as flexibility and dynamicity is required, while maintaining a resource and energy efficient system. This is done using the scheduler, implemented in the eNodeB in LTE, which allocates the available resource blocks according to the UE's need and condition. [10] [11] [13]

This chapter introduces the frame structure behind scheduling, and how the current scheduler manages the resources between all active users. Finally, MCS selection in LTE/NR 3GPP is explained in detail.

4.1. Frame Structure

In LTE, like several communication systems, data is exchanged in frames. A radio frame, depicted in Fig. 11, is composed of 10 subframes each of 1 ms. This composition enables the synchronization of data between the eNodeB and the UE, where each transmitted packet contains the frame and subframe number.

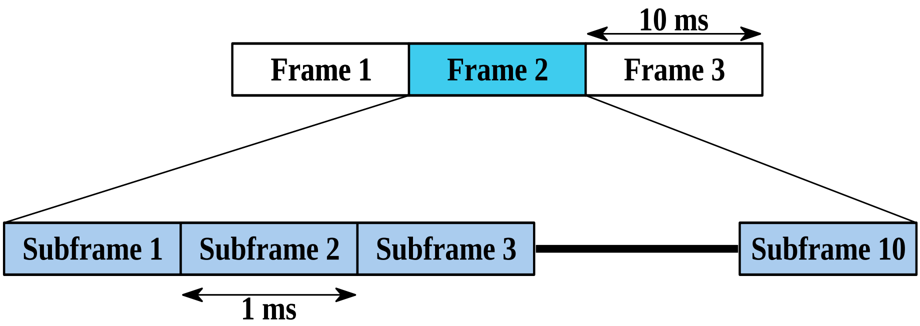

Fig. 11. Radio frame structure.

The fundamental transmission schemes are the same for DL and UL in LTE, given that OFDM is used in the DL and SC-FDMA in the UL, with 15 kHz subcarrier separation in both links.

Although both FDD and TDD operations are supported in LTE, two frame structures, with the same frame length, are available due to the differences between the TDD and FDD versions.

4.1.1. Type 1: FDD Version

In FDD, DL and UL uses different carrier frequencies, so a device that supports HD-FDD must switch between the links for transmission and reception, while another device can use both links simultaneously if it supports FD-FDD.

Every subframe in type 1 has 2 slots with 6 or 7 OFDM each, depending on the cyclic prefix. Fig. 12 shows an overview of the frame structure, where each OFDM symbol is used as reference or transmission signal, throughout the UL or DL transmission, like synchronization, broadcast, reference, control, ARQ, etc. [10] [11]

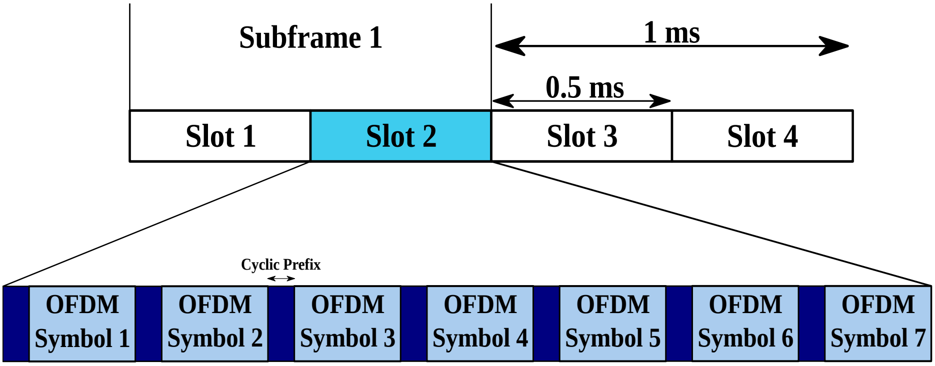

Fig. 12. Frame Structure in LTE - FDD [11].

4.1.2. Type 2: TDD Version

In TDD, the DL and UL transmission uses the same subcarrier frequency, so like HD-FDD, a switch between the links is required for reception and transmission. This is done in type 2 using special subframes, as depicted in Fig. 13.

The TDD version uses two types of subframes: normal and special, where special subframes replace subframes 2 and 6 [11].

A normal subframe is used either for DL or UL. While a special subframe is composed as follows: Downlink Time Slot, Guard Period and Uplink Time Slot. It provides a dynamic structure for changes from DL to UL using the GP.

The possible DL-UL configurations are cell specific as shown in the Table 2 [9].

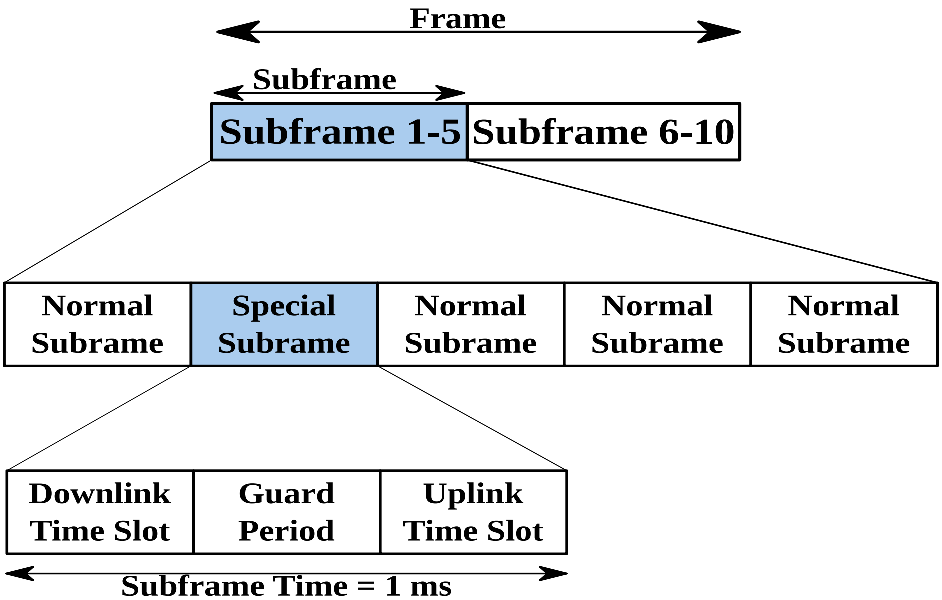

Fig. 13. Frame Structure in LTE - TDD [11].

Even though LTE devices can support any combination of the duplex modes while increasing both complexity and performance, type 2 frame structure has several advantages over type 1, and one of the them is the dynamic switch between DL and UL provided by the special subframe structure using the same carrier frequency. Based on the previous section, one RB is part of a frame, and the dynamic resource allocation per subframe per user is done in the scheduling part at the eNodeB, as follows [10] [11].

Table 2. UL-DL configuration for LTE-TDD.

UL – DL Configuration	Subframe Number									
	0	1	2	3	4	5	6	7	8	9
0	DL	S	UL	UL	UL	DL	S	UL	UL	UL
1	DL	S	UL	UL	DL	DL	S	UL	UL	DL
2	DL	S	UL	DL	DL	DL	S	UL	DL	DL
3	DL	S	UL	UL	UL	DL	DL	DL	DL	DL
4	DL	S	UL	UL	DL	DL	DL	DL	DL	DL
5	DL	S	UL	DL	DL	DL	DL	DL	DL	DL
6	DL	S	UL	UL	UL	DL	S	UL	UL	DL

4.2. Scheduler

With the increasing number of active UEs, scheduling plays an essential role in the system, more specifically in the eNodeB, by allocating the available yet limited RBs to the correct UEs, thus increasing the efficiency of the system. Although the scheduling strategy is not standardized, different strategies are used by different vendors to provide the required QoS for the UEs. [11]

When compared to older allocation methods like round robin or proportional queuing, the scheduler significantly increases the throughput inside the cell, even though it adds complexity and doesn't has a big advantage at the cell's edges. [10]

In each eNodeB, there is a DL and an UL scheduler to schedule both transmissions separately. Both schedulers are responsible for dynamically controlling the UEs in their respective transmissions, and their tasks are as follows:

1. Downlink Scheduler:

The DL scheduler uses CSI for channel dependent scheduling, and its task is to determine the UE(s) to transmit to and the set of RBs for each of these UEs. When multiple UEs are to be scheduled, the DL scheduler controls the instantaneous bit rate of each UE, multiplexing of channels for simultaneous transmission, and MCS selection. Moreover, the interference from neighboring cells can affect the transmission power on specific RBs, thus the coordination between different BS can increase the efficiency for UEs at cell edges. [11]

2. Uplink Scheduler:

The UL scheduler is similar to the DL one as it determines which UE(s) can transmit UL information on each subframe, also called scheduling grant. The main task of this scheduler is to ensure the transmission of UL information from different UE on a subframe basis. Channel information and inter-cell interference coordination are also exploited in the UL scheduler, similarly to the DL one. [11]

A key requirement for scheduling at the eNodeB is the instantaneous DL channel knowledge gathered in the CSI report. Every active UE is then allocated RBs on a subframe basis using specific parameters such as SINR, HARQ, etc. [13]

Fig. 14 depicts the work done by the scheduler in the eNodeB. By taking several parameters into consideration, RBs can be allocated to provide the best QoS based on the UE's need and link condition, with the help of MCS as explained in the following section.

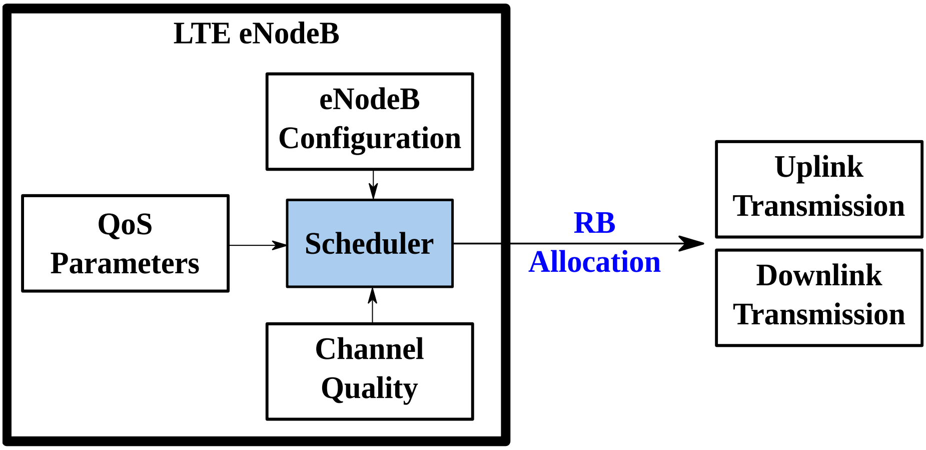

Fig. 14. Scheduler in eNodeB of LTE.

Thus, improving the scheduler can lead to simpler yet more effective method to improve the link between the eNodeB and the UE, and this can be done using machine learning.

4.3. Modulation and Coding Scheme

Other than RBs allocation, the scheduler uses the available information to choose the UE's MCS, which directly affects the QoS. MCS is an index based on the channel quality indicator (CQI) sent by the UE. The UE's UL report is used at subframe N to determine the MCS index for subframe N+2 using a lookup table.

Over the different LTE releases, the lookup tables for MCS selection have been upgraded for different modulation schemes. The number of MCS indexes was 15 at first, as shown in table 3 [15], and it reached 31 in the current version in table 4.

Using several parameters like CQI, HARQ and User's data size, the MCS is chosen, and thus, the modulation scheme and coding rate are determined. This process should be accurate and fast, as an inaccurate MCS can result in a retransmission in case of bad channel condition, or inefficiency in case of good channel condition. Also, the MCS selection is done between subframes, so the decision should be fast, especially if several UEs are connected to the eNodeB.

Table 3. 3GPP CQI lookup table [28].

CQI index	**Modulation**	**Code Rate x1024**	**Efficiency (bits/s)/Hz**
0	Out of range		
1	QPSK	78	0.1523
2	QPSK	193	0.377
3	QPSK	449	0.877
4	16QAM	378	1.4766
5	16QAM	490	1.9141
6	16QAM	616	2.4063
7	64QAM	466	2.7305
8	64QAM	567	3.3223
9	64QAM	666	3.9023
10	64QAM	772	4.5234
11	64QAM	873	5.1152
12	256QAM	711	5.5547
13	256QAM	797	6.2266
14	256QAM	885	6.9141
15	256QAM	948	7.4063

Table 4. 3GPP MCS lookup table [28].

MCS Index	Modulation Order	MCS Index	Modulation Order
0	2	16	4
1	2	17	6
2	2	18	6
3	2	19	6
4	2	20	6
5	2	21	6
6	2	22	6
7	2	23	6
8	2	24	6
9	2	25	6
10	4	26	6
11	4	27	6
12	4	28	6
13	4	29	2
14	4	30	4
15	4	31	6

4.4. Scheduling Strategies

Based on section 4.3, the MCS is selected based on the UE’s CQI report, which is directly related to SINR and link quality. The scheduler chooses the UE’s MCS by comparing the UE’s CQI to the lookup tables 3 and 4.

However, as mentioned in 4.2, the scheduling strategy is not standardized and some of strategies used by the vendors are as follows:

1. Normal Strategy:

Using the estimated CQI, the MCS is selected based on the table’s CQI equal to the estimated one. This strategy follows the lookup table and the MCS index is increased when an ACK is received from the UE, while it’s decreased when a NACK is received.

2. Conservative Strategy:

The conservative strategy targets a correct reception of the transmitted data even if the efficiency is decreased. This strategy selects a lower MCS index than the estimated one, until it reaches the maximum value in the lookup table. When a NACK is received, the index is decreased. However, even if an ACK is received, an increase in the MCS index is done when the CQI increases by two indexes.

3. Aggressive Strategy:

Opposite to the conservative strategy, the aggressive strategy targets a higher efficiency in terms of bit rate over the possibility of incorrect reception. The selected MCS is higher than the estimated one until the SINR reaches its minimum values, where MCS = {0}. When an ACK is received, the index is increased, and the MCS index decreases only when a NACK is received and the CQI index drops by two indexes.

Based on the vendor and the applications used the UEs, the MCS strategy is selected to either increase the data rate by compromising a higher error rate, decrease the error rate by compromising a higher efficiency, or choose a balanced mix between of both efficiency and correct reception.

In this thesis, the normal strategy is used as the MCS indexes are selected using the CQI report and the lookup tables in section 4.3.

5. Model Design

The current scheduler, implemented in LTE/NR 3GPP, selects the MCS per subframe per user using the CQI provided in the uplink by the UE, as explained in chapter 3. [15][28]

However, the current scheduling process has the following limitations:

- MCS selection is limited for the next subframe for a given user. Entering to 5G NR, the processing complexity will increase, and the resources can become very limited. Hence, it is always a significant advantage for the resource planning of the independent UEs to have the expected MCS which will be used for the future subframes.

- The MCS selection becomes very complicated in the MU-MIMO case as the MCS for the user cannot only depend on the CQI, but also depends on the other UEs in the MU-MIMO group (group of users scheduled at the same time-frequency resource). The spatial relation between all the users in the group needs to be analyzed and this takes lots of critical processing resources, resulting in a bottleneck for the NR.

To take care of the aforementioned limitations, we propose the usage of ML at the BS aiming to predict future MCS. In summary, we propose the following:

- MCS prediction for the future subframes/slots using ML.
- Extend the MCS prediction for the candidate users of MU-MIMO.

To diminish the process at the scheduler, one can provide it with information for future subframes using ML. Based on chapter 4, ML can provide a good accuracy given the constant flow of data from the UE to the scheduler. Therefore, given the decision is an integer between 0 to 31, for the LTE-advanced Pro (Release 14) [28], classification algorithm is to be used instead of regression for the following reasons:

- Output is an integer with predefined number of classes and neighbors.
- High complexity of regression as several parameters are taken.

5.1. Scheme

First, every active UE sends its SINR value, which is included in the CSI report, at the start of each frame in the UL transmission. The CSI report is sent periodically at the start of each frame, so the information can be used over all subframes for the MCS selection.

Meanwhile, many parameters; as HARQ feedback, current MCS and data size, can also be collected on a subframe basis from the UE and the eNodeB.

The parameters used for the MCS prediction, over the course of one frame, are shown in Fig. 15. As mentioned above, the SINR is collected once every 10ms while the rest of the parameters are constantly updated.

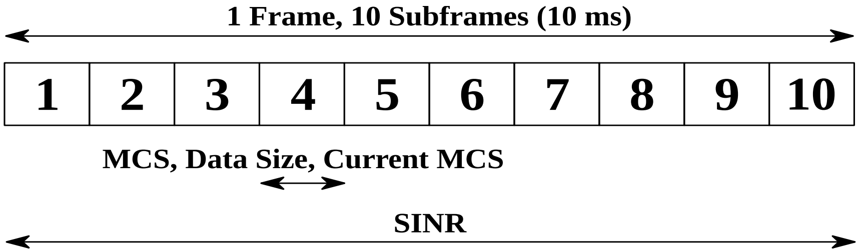

Fig. 15. Parameters used for MCS prediction in a frame.

The overall system model is depicted in the following block diagram of Fig. 16. Based on section 5, scheduling is done at eNodeB using the UL information of the user. To increase the efficiency, ML will provide information as well to the scheduler based on a training database, as mentioned in section 4. Thus, the scheme includes three main parts: UE, Channel and Base Station.

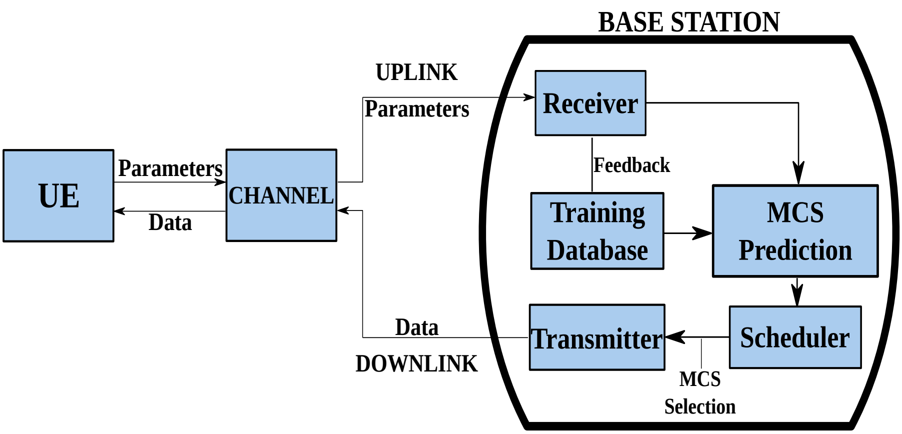

Fig. 16. Block diagram of the suggested system model.

5.2. Parameters

To make the system practical, a set of the parameters used for current MCS selection, based on section 4.2.1, are also used in the ML prediction algorithm. It includes SINR, HARQ, current MCS (k) and data size.

Moreover, to improve the accuracy of the ML algorithm, the training database includes new features:

5.2.1. Current MCS Accuracy

A flag indicating the accuracy of the MCS selected. In case of a retransmission at the next subframe, the MCS was not optimal and the flag will be indicated as zero. On the other hand, if the transmission was successful, then the flag will be marked as one.

5.2.2. UE Scenarios

The training dataset includes data collected from the laboratory logs, where several UEs, in different channel conditions were tested. To keep track of the different behaviors, each UE was assigned a UserID, which will be used in the ML prediction algorithm. The dataset includes UEs in alternative scenarios, and later a prediction is made for the current UE. Furthermore, it is useful to investigate the closest historical UE, so the algorithm can track the stored behavior and assign it to the current one.

6. Implementation

Based on the suggested scheme in chapter 5, the prediction method used in this thesis can be implemented, with some add-ons, at the base station to increase the efficiency of the scheduler. These add-ons are the training database and the ML algorithm.

In this chapter, a step-by-step overview of the work done will be presented in detail, from the selected programming language to the implementation of the ML algorithm.

6.1. Programming Language

Nowadays, several platforms and programming languages include AI and ML in their libraries like Microsoft, IBM, MATLAB, etc. In this thesis, Python was chosen for scripting and programming due to the free and open sourced extensive libraries [29].

Additionally, Python is a dynamic and adaptive programming language, and it can be used for several purposes as it is easy to implement. One of them is scripting, which is implemented in section 6.3 to gather useful data from large files [29].

Finally, since 2007, Python is continuously updating its ML libraries such as Tensor Flow, SciKit Learn, NumPy, etc. They were used to write the ML algorithm for MCS prediction in section 6.5.

6.2. Trace Logs

Trace logs were collected using Ericsson's system and equipment. Using an eNodeB and several stationary UEs in the laboratory, various scenarios were created by adding intercell interference and by varying the BS's transmission power. Data from several UE scenarios with very different channel conditions were included in the trace logs, and even though the UEs weren't moving, the MCS is varying for different scenarios given the interference level from the neighboring cells. The trace logs collection was done as shown in the block diagram in Fig.17.

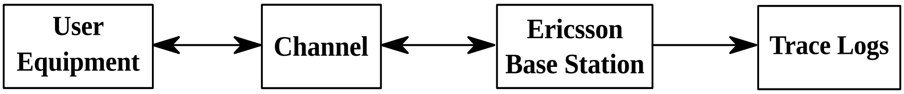

Fig. 17. Model used for trace logs collection.

One of the scenarios used in the laboratory is shown in Fig. 18, where using two UEs connected to two different BSs, their positions were similar to create interference from both cells. Other scenarios included UEs close to the BS, at the edge and outside their cells.

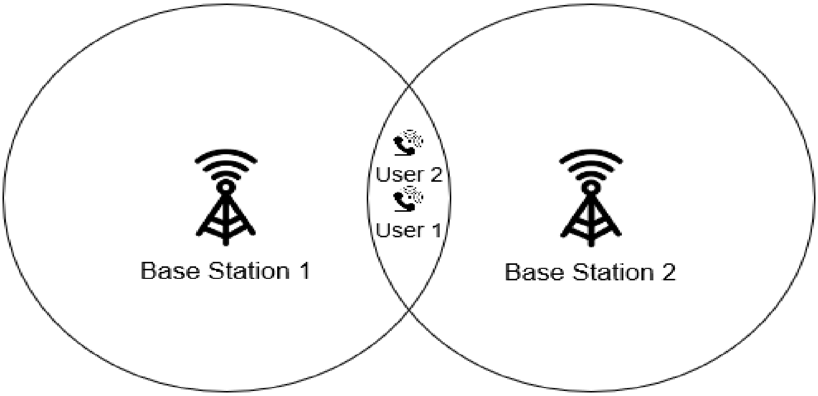

Fig. 18. One UE scenario used in the prediction.

The main purpose of the UE scenarios is creating a variety of channel conditions, where for the same SINR, the MCS can vary due to the interference from neighboring cells and eNodeB.

6.3. Data Collection

After running the simulation in section 5.1, the trace logs need filtering to gather useful data for the ML algorithm. The parameters, listed in section 5.2, form the columns in the dataset, and the remaining data, after filtering, form the entries. Table 5 represents the useful parameters, including their types and description, collected from the trace logs.

In section 5.1, it was mentioned that the SINR is updated every frame while the rest of parameters is updated every subframe.

Table 5. Parameters used in the training dataset

Parameter	Variable Type	Description
userID	integer	Indicates user’s scenario
SINR	double	Included in the CQI report
MCS	integer	MCS of current subframe
AccMCS	flag (Bool)	Accuracy of the selected MCS
next MCS	integer	MCS for the next subframe
NDF	flag (Bool)	Based on HARQ feedback
Bytes	integer	Data size to be transmitted

To predict the future MCS, a sample must be included in the training dataset. This sample is “next MCS” and it was collected by processing the whole frame of each UE and appending the MCSs in one column. For example, the “next MCS” for subframe N is the actual “MCS” for subframe N+2.

6.4. Assumption

The data collected in section 6.3 created a large dataset, but it also includes a lot of erroneous entries where, for example, a high MCS is selected for a very low SINR. For simplicity and to increase accuracy, some assumptions were taken into consideration by adding some constraints to the dataset,

- MCS range between 1 and 28

Although the MCS indexes range between 0 and 31, the prediction will be limited to the transmission phase, with MCS between 1 and 28. First, MCS = {0} corresponds to no transmission either because no data is available, or the user is inactive, which can be predicted in the current algorithm. Second, MCS = {29, 30, 31} corresponds to re-transmission, which isn’t useful for the transmitter as it can’t be accurately predicted due to circumstances like UE’s inaccurate channel estimation, erroneous feedback, misdetection, etc.

- Laboratory Logs, not live logs

Live logs were technically and logistically hard to collect, so lab logs were used to test the system.

- SINR >= 0

In the laboratory, the BS can always provide the UEs with positive SINR due to the small environment, for path loss, and limited to no interference inside.

- Bytes > 0

Back to the first assumption, if the data size = 0, it can result in MCS = {0}, which isn't included in this prediction. Also, the UE is assumed to be active, so each UE has to be assigned data.

- Stationary UE, Section 5.1

6.5. ML Algorithm

Following the steps in the previous sections, the final dataset was used to train and test the algorithm. In fig. 19, the training sequence and test data are both part of the collected data and they contribute together to the ML prediction, and they are divided based on the test size, according to fig. 8 in section 3.1.

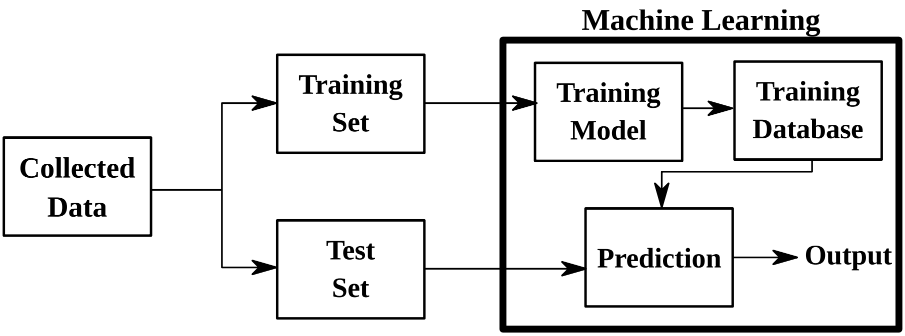

Fig. 19. ML algorithm training and testing

In this thesis, the prediction was made using the neural network classification algorithm, presented in section 3.2.1.2. The main reason behind choosing classification is the integer output using multiple variables, and the neural network is in accord with the prediction steps, as shown in fig. 20. The accuracy of the prediction over the varying test size is being shown in chapter 7.

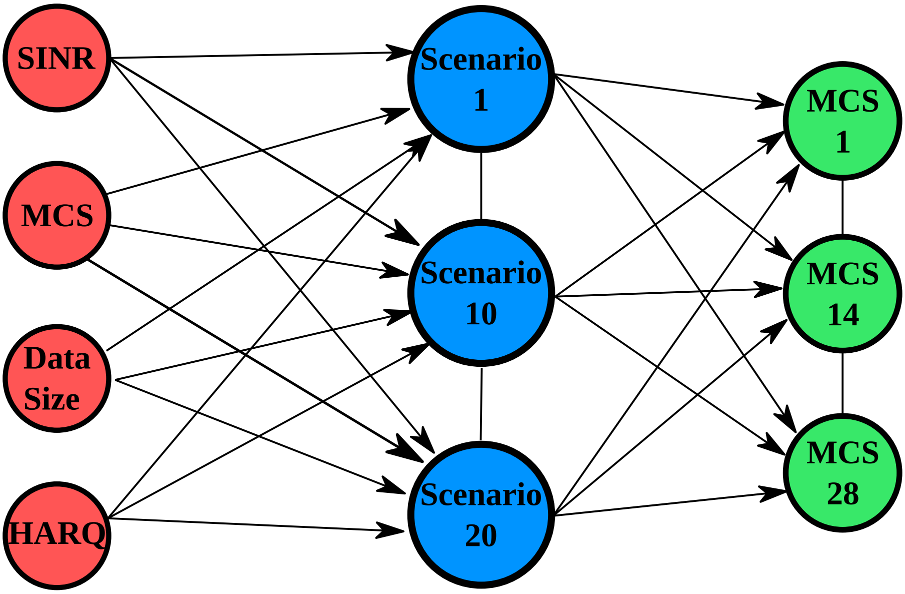

Fig. 20. Neural network algorithm for MCS prediction.

In details, the future MCS prediction, output, is as follows:

- Predict the scenario (hidden node) based on the input data
- Predict the MCS based on output of the hidden node

By choosing the closest UE scenario from the training dataset, using the parameters mentioned in table 3 as inputs, the future MCS can be predicted as a smaller yet more accurate and relative dataset will be used.

The flowchart in Fig. 21 shows the detailed steps taken to make an MCS prediction based on the input data from the UE.

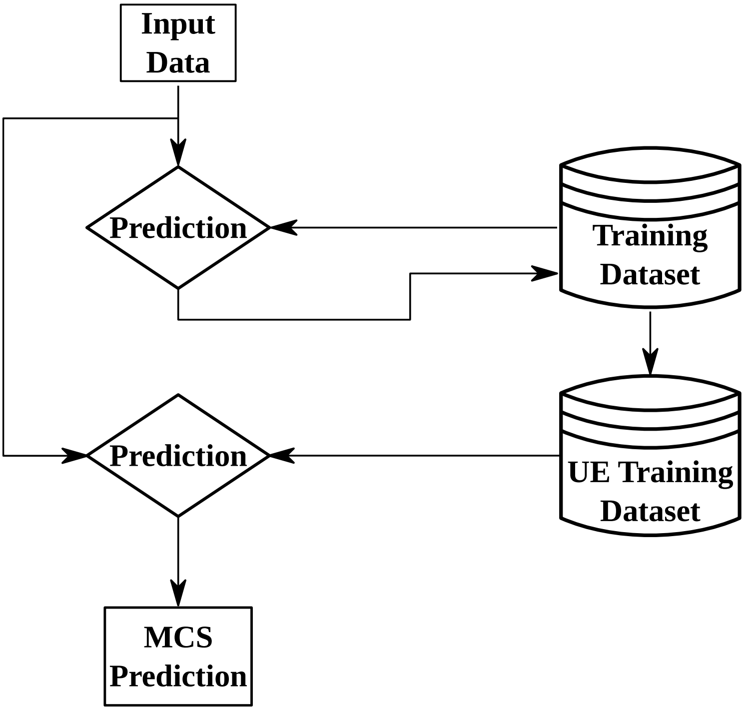

Fig. 21. Flow chart of the MCS prediction.

The UE training dataset is a sub-dataset of the main training dataset, and it is selected using the predicted UE scenario (hidden node in the neural network). The second prediction follows using the same input data and the selected sub-dataset, so no additional data is added to the overall prediction.

7. Results

After implementing the designed system, based on chapters 5 and 6, several simulations were made to test the accuracy of the prediction. As neural network was used, the sub-predictions were also simulated separately to test the efficiency of all parts using 10% of the dataset as testing size. The overall dataset size of collected samples reached approximately the 400.000 entries.

The results, shown in this chapter, will be divided by prediction type, where the scenario and MCS predictions will be simulated separately, before simulating the whole algorithm.

Moreover, as the k-NN is implemented in both cases, the number of neighbors will also be tested to find the highest accuracy.

7.1. Accuracy

The accuracy of an ML algorithm is the ratio of correct predictions over total number of predictions, and it ranges between 0 and 1:

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ number\ of\ Predictions\ made}$$

As explained in section 3.1, in the test phase, the output of the algorithm is compared to the test set, which includes the correct data. For classification algorithms, a correct prediction is made when the output is the correct class.

7.2. Application of the Assumptions

To prove the usefulness of the assumptions made in section 6.4, the predictions made in the following sections will include the simulation results on datasets with and without the assumptions.

7.3. Relation between MCS and SINR

First, based on the assumptions made in chapter 6, the MCS distribution over the SINR range is shown in Fig. 22. This is due to the various scenarios taken into consideration while collecting the MCS indexes for all UEs.

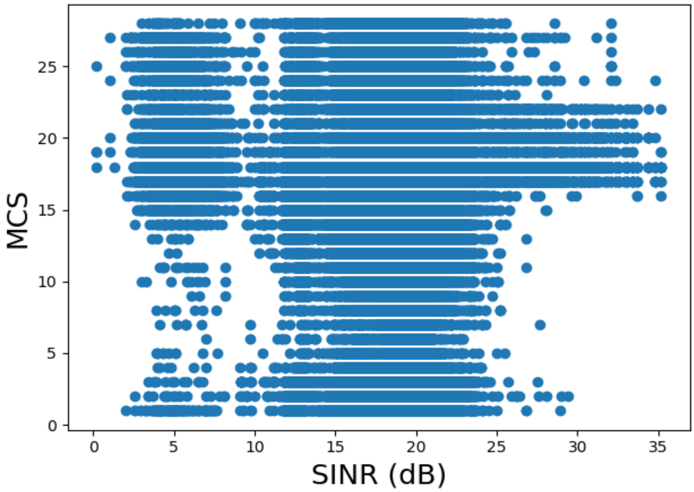

Fig. 22. MCS over SINR plot for all UE.

The MCS is distributed all over the SINR range. For high SINR, over 25 dB, the MCS is relatively high as the channel condition is good enough to only use high modulation schemes. The following plots show the distribution of MCS over SINR for two different UEs. The first UE, on the left side, has an even distribution of low MCS indexes over the varying SINR. This is due to several reasons, one of them is the low data size used by this UE while, for example, making a phone call. On the other side, the second UE has high MCS indexes over the majority of SINRs for several reasons, mainly the high data size and low interference from other cells.

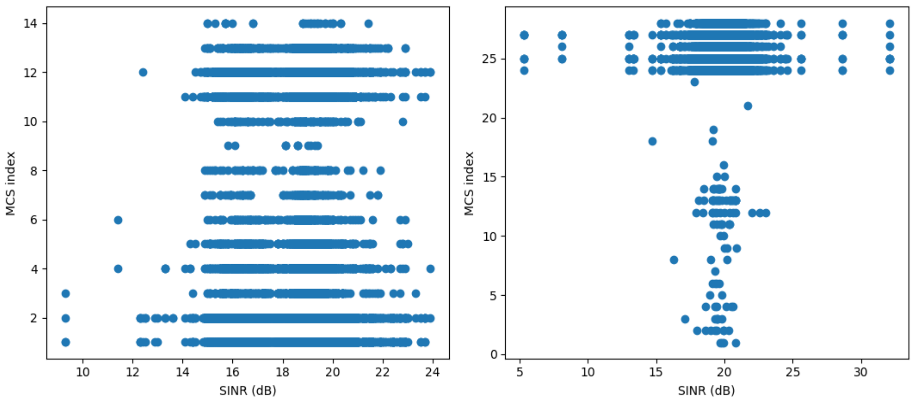

Fig. 23. MCS over SINR plots for two different UEs

7.4. UE Scenario Prediction

The UE scenario prediction accuracy is shown in fig. 24. The output results of the prediction, in the sub-dataset selection, used in section 7.3. Two curves are shown in the plot, where the blue and orange curves represent the datasets used with and without assumptions respectively. The assumptions are listed in section 6.4.

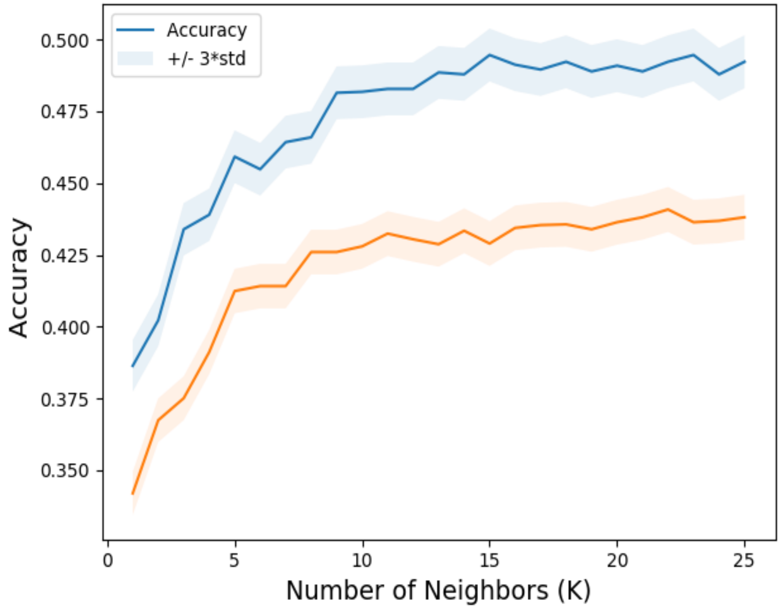

Fig. 24. UE scenario prediction accuracy.

Using the assumptions to make the UE scenario is accurate over the range of neighbors, and the highest accuracy is obtained using 23 neighbors. This result is in accordance with the available dataset, as 24 different scenarios were included.

7.5. MCS Prediction

To show the accuracy of the MCS prediction, the datasets were used to predict the future MCS directly. The output of the predictions were plotted and shown in Fig. 25.

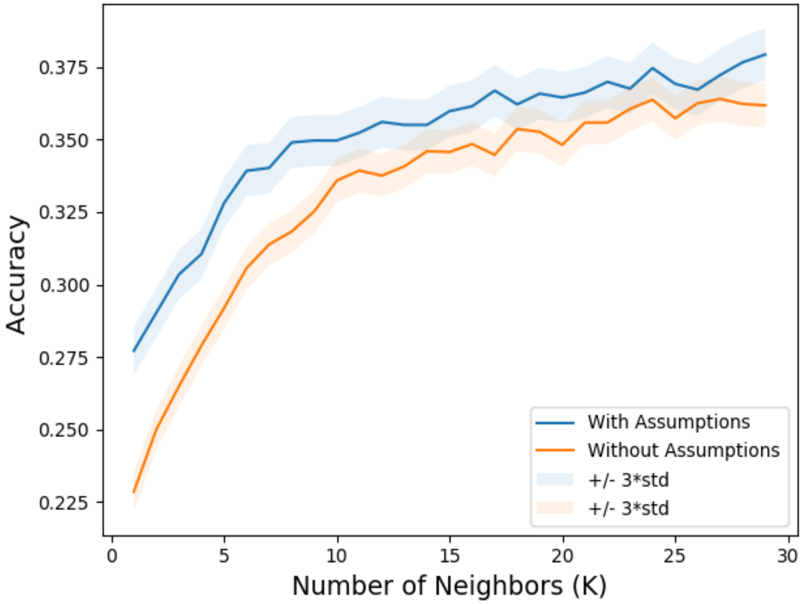

Fig. 25. MCS prediction accuracy.

For different neighbors, the accuracy of the prediction using the assumptions (blue curve) is better than without them (orange curve). This result is expected given that by deleting some less predictable results, like the MCS = {0, 29, 30, 31} the output will be useful and more related to the provided dataset.

7.6. Neural Network Prediction

In section 6.5, it was mentioned that the final output is the combination of two predictions: predicting the MCS, example based on the sub-dataset related to the predicted UE scenario. Section 7.4 and 7.5 give respective examples using the full datasets.

Fig. 26 shows the MCS prediction using single UE scenarios only, because of the scenario prediction in the first place.

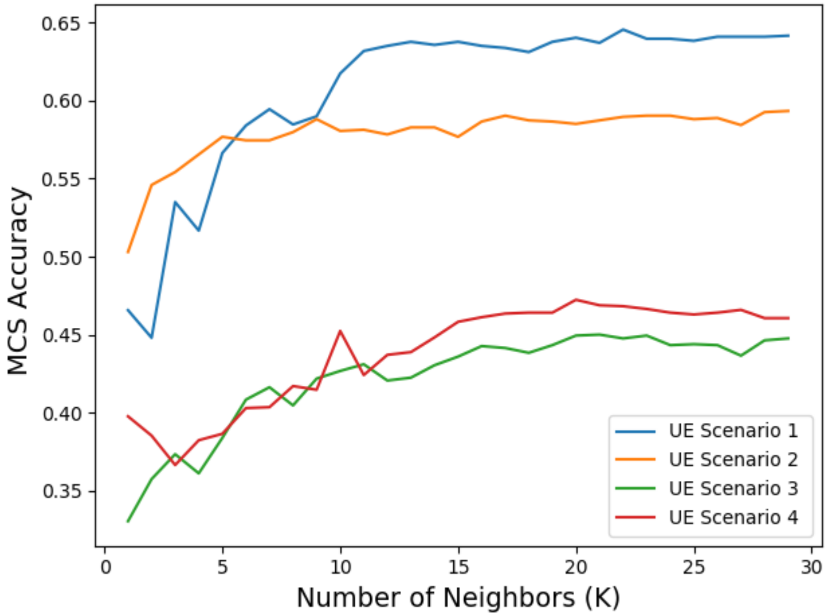

Fig. 26. MCS prediction accuracy for different UE scenarios.

The variation in accuracy for different scenarios and different number of neighbors is due to several reasons:

- The variation of the parameters in each scenario, and a bad channel condition can give erroneous MCS resulting in a less accurate prediction.
- The variation of MCS in one scenario, where the indexes can be concentrated or diverse, as previously shown in fig. 23. The varying number of MCS indexes used in a scenario will change the peak accuracy according to the number of neighbors, and for each scenario, a specific number of neighbors should be taken to get the best possible prediction.

Moreover, Fig. 27 shows the average accuracy of the neural network compared to the direct MCS prediction in section 7.5. A prediction using the neural network algorithm is better for all number of neighbors with a margin of 5% (0.05 difference).

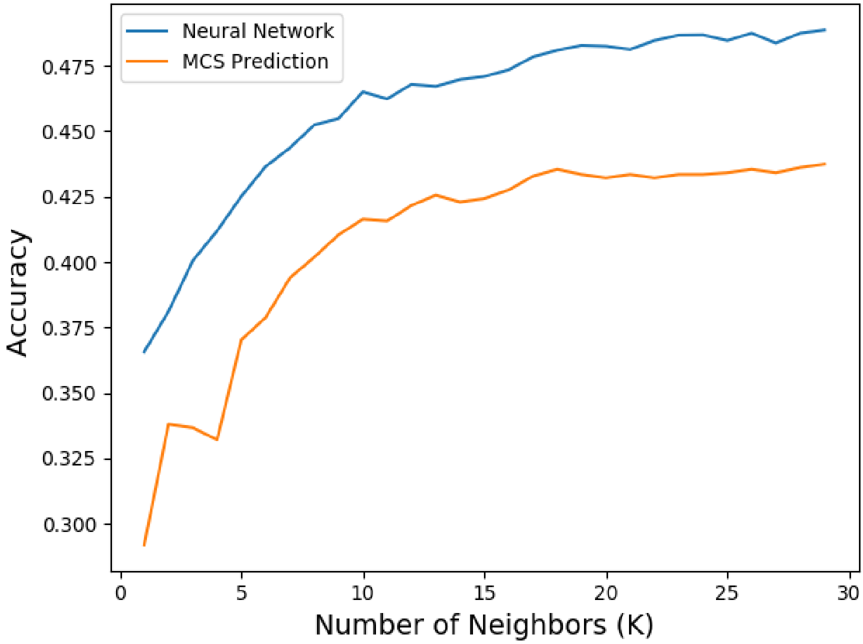

Fig. 27. Neural network accuracy.

This is due to the specific sub-datasets used in the neural network, where the prediction outputs are limited yet more relative to the input parameters.

8. Conclusion and Future Work

8.1. Conclusion

With the upcoming arrival of 5G and the essential adoption of MIMO technology which is included in it; complexity and energy consumption are increasing in a “worrying” way. To solve this problem, we propose a multi-class ML model based on neural networks which is going to predict the MCS index for the expected subframe of an independent UE. By exploiting the classification learning, we can reduce the aforementioned “worrying” parameters and enhance the performance of link adaptation. However, some of the challenges that we faced restricted the results and the possible outcomes of our project, as most of our measurements, simulations and collected data are narrowed at the laboratory environment. The reasons for this were, primarily, the limited time of our research in comparison with the considerable capabilities of investigation on ML topic, and secondary, the lack of substantial real-time data logs which would expand the exploration of our thesis in realistic channel conditions. Furthermore, the prediction accuracy of the MCS index selection algorithm achieved its peak rate, which is fifty percent, giving thusly promising rates for further exploration. Hence, this prediction is aiming to benefit in an optimum way the BS, as it is always a significant advantage for the resource allocation if MCS for the users are already known for the future subframes.

8.2. Future Work

Even though the prior challenges were significant in many ways, future work and much more research can be done to expand, while reaching the true capabilities of machine learning and Massive MIMO technology which is expected world widely. A critical extension of this work concerns the investigation of the ML model for the MU-MIMO MCS prediction scenarios. This can be done by training the ML model for MU grouping, where a group could be defined as several users which are spatially separated; although they fall in similar average MCS regions. However, the previous extension requires the expansion of the training parameters and increasing even more the database and the training time of the ML algorithm. Thus, the proposed suggestion can develop greatly the scope of our thesis, while helping to reach the high requirements of 5G NR.

Reference

[1] Global Machine Learning Market Trend, Analysis and Forecasts to 2025,https://orbisresearch.com/reports/index/machine-learning-market-to-2025-global-analysis-and-forecasts-by-services-and-vertical

[2] A. Samuel, *"Some Studies in Machine Learning using the Game of Checkers"*, IBM Journal of Research and Development, vol. 3, no.3, pp. 210-229, July 1959

[3] Ericsson 5G expected number of users, press release, https://www.ericsson.com/en/press-releases/2017/11/ericsson-predicts-1-billion-5g-subscriptions-in-2023

[4] Y. Lu, G. Xu, *"Channel and Modulation Selection Based on Support Vector Machines for Cognitive radio"*, International Conference on Wireless Communications, Wuhan, China, April 2007

[5] R. Daniels, R.W. Heath, *"Online Adaptive Modulation and Coding with Support Vector Machines"*, European Wireless Conference, Lucca, Italy, June 2010

[6] R.C. Daniels, C.M. Caramanis, R.W. Heath, *"Adaptation in Convolutionally Coded MIMO-OFDM Wireless Systems Through Supervised Learning and SNR Ordering"*, IEEE Transactions on Vehicular Technology, vol. 59, no. 1, January 2010

[7] C.B. Ha, Y.H. You, H.K. Song, *"Machine Learning Model for Adaptive Modulation of Multi-Stream in MIMO-OFDM System"*, IEEE Access, vol. 7, pp. 5141-5152, December 2018

[8] R. Bruno, A. Masaracchia, A. Pasarella, *"Robust Adaptive Modulation and Coding (AMC) Selection in LTE Systems Using Reinforcement Learning"*, IEEE 80th Vehicular Technology Conference (VTC-Fall), 978-1-4799-4449-1, Vancouver, Canada, December 2014

[9] 3GPP Workshop Release 8, June 2010, http://www.3gpp.org/Information/presentations/presentations_2010/

[10] O. Liberg, M. Sundberg, and E. Wang, "*Cellular Internet of Things: Technologies, Standards and Performance*", London, Elsevier, 9780128124581, 2018

[11] E. Dahlam, S. Parkvall and J. Skol, "*4G, LTE-Advanced Pro and the road to 5G, 3rd edition*", Academic Press, 9780128045756, 2016

[12] G. Lindell, *"Introduction to Digital Communications"*, Sweden, Lund University, August 2006

[13] A. Goldsmith, "*Wireless Communications, 2nd edition*", UK, Cambridge University Press, 9780521837163, 2005

[14] S. Lin, D.J. Costello, *"Error Control Coding"*, 2nd edition, USA, Pearson Prentice Hall, 0-13-042672-5, 2004

[15] European Telecommunication Standardization Institute (ETSI), 3GPP TS 36.213 version 12.3.0 Release 12, https://www.etsi.org/deliver/etsi_ts/136200_136299/136213/12.03.00_60/ts_136213v120300p.pdf

[16] C. Liu, L. Tian, Y. Xiang, W. Hong, *"Development of LTE Wideband Receiver"*, 3rd Asia-Pacific Conference on Antennas and Propagation, 978-1-4799-4354-8, December 2014

[17] M. Jiang, L. Tian, J. Yu, W. Hong, *"Design of LTE wideband Transmitter"*, 3rd Asia-Pacific Conference on Antennas and Propagation, 978-1-4799-4354-8, December 2014

[18] A.F. Molisch, *"Wireless Communications"*, 2nd Edition, USA, Wiley & Sons Ltd, 978-0-470-74187-0, 2011

[19] A. M. Turing, "*Computing Machinery and Intelligence, Mind 49*",pp.433-460, 1950

[20] T. Mitchell, *"Machine Learning" 1st Edition*, USA, McGraw-Hill, 978-0-0704-2807-2, 1997

[21] S. Marsland, "*Machine Learning: An Algorithmic Perspective, Second Edition*", US, CRC Press, 978-1-4665-8328-3, 2015.

[22] C.M. Bishop, *"Pattern Recognition and Machine Learning"*, USA, Springer Science and Business Media LLC, 978-0-3873-1073-2, 2006

[23] A. Géron, *"Hands-On Machine Learning with Scikit-Learn & TensorFlow"*, USA, O'Reilly Media Inc., 978-1-491-96229-9, 2017.

[24] M. Aly, "Survey on Multiclass Classification Methods", malaa@caltech.edu, November 2005.

[25] Overview of ELM, Nanyang Technological University of Singapore, http://www.ntu.edu.sg/home/egbhuang/index.html

[26] Y. Dai, H. Yan, *"The Comparison of Five Discriminant Methods"*, International Conference of Management and Service Science, 978-1-4244-6581-1, Wuhan, China, August 2011.

[27] M. Al-Rubaie, J.M. Chang, *"Privacy-Preserving Machine Learning: Threats and Solutions"*, IEEE Security and Privacy, vol. 17, no. 2, pp. 49-58, March 2019

[28] European Telecommunication Standardization Institute (ETSI), 3GPP TS 36.211 version 14.2.0 Release 14, https://www.etsi.org/deliver/etsi_ts/136200_136299/136211/14.02.00_60/ts_136211v140200p.pdf

[29] Python official website, https://www.python.org/