Convolutional Autoencoder: An Approach for Channel Estimation in 5G networks and Beyond

Abstract

In mobile communications, the channel estimation process is one of the main keys to optimize the communication between the transmitter and the receiver. Knowing the channel response is a challenge because there are multiple phenomena like attenuation, multi-path loss, noise, and delays that affect the transmitted signals. Methods based on pilot insertion such as Least Squares (LS) and Minimal Mean Squared Error (MMSE) are commonly used to estimate the channel. Nevertheless, they have issues related with their performance and complexity in varying scenarios. In this paper, a Convolutional Autoencoder (CAE) technique assists a pilot-based channel estimation for a 5G communication system affected by Doppler shift due to the level of mobility. Through simulation modeling including Line-of-Sight (LoS) and Non-Line-of-Sight (NLoS) environments in a tapped delay line (TDL) model, it is measured the performance based on the Bit Error Rate (BER), Error Vector Magnitude (EVM), estimation time, and Mean Squared Error (MSE). The results prove that CAE estimation outperforms lineal interpolation technique on a SNR range between –10 dB and 15 dB. Furthermore, the proposed estimation technique has a similar performance with Practical Estimator (PE) in low-speed conditions.

Keywords: Convolutional Autoencoder, Channel Estimation, Mobile communications, pilot signals, 5G

1 Introduction

The continuous exponential growth in demand for fast, high performance and reliable wireless communication systems requires a proper use of the spectrum's resources and the processes related with the transmission and reception of data traffic. In this context, the fifth generation (5G) comes up as an important technology that leads the digital transformation through key features such as low latency, high bandwidth, maximum throughput and enhanced capacity [1]. To boost the capabilities of the 5G communication systems and guarantee the users requirements, three major service types: Enhanced Mobile Broadband (eMBB), Massive Machine-Type Communication (mMTC), and Ultra-Reliable and Low-Latency Communications (uRLLC), have been proposed and deployed in currents 5G installations [2]. 5G is based on Orthogonal Frequency Division Multiplexing (OFDM), which is used in the up-link

and down-link. OFDM increases the degree of flexibility and scalability of the 5G systems with different applications scenarios, deployment models and a wide range of frequencies [3]. In this context, multiple numerologies e.g. 15 kHz, 30 kHz, 60 kHz, 120 kHz, 240 kHz, and 480 kHz and diverse options of subcarrier modulation schemes such as Quadrature Phase Shift Keying (QPSK), 16 Quadrature Amplitude Modulation (16QAM), 64QAM, and 256QAM make part of the 5G standard. In addition, an optimal channel characterization enhances an appropriate demodulation of the received signals, considering the attenuation, interference, delay spread, noise, and other effects that affects the signal propagation.

Deep Learning (DL), a branch of Machine Learning (ML) and Artificial Intelligence (AI), has become an important topic in computing, data science and analytics, being considered as a core technology of today's Fourth Industrial Revolution [4]. Due to its high performance with the processing of large amount of data, DL is widely used in various applications such as: image and voice recognition, natural language processing, time series predictions, data classification, among others [5–8]. DL is based in representation learning algorithms known as Artificial Neural Networks (ANNs) that can be grouped and configured in a layer architecture. These models are capable of finding complex patterns and characteristics in challenging prediction or classification tasks. The main advantage of ANNs over traditional ML methods is a better performance with large datasets [9]. The depth of an ANN is related with the number of layers and trainable parameters known as weights. Training a model takes a long time due to substantial number of parameters that need to be adjusted to guarantee the optimization of a cost function (commonly the minimization of a loss function). However, it takes a short amount of time to produce an output during testing compared to other machine learning algorithms [5]. There are multiple DL algorithms, the most common are: Multi-Layer Perceptron (MLP), Convolutional Neural Networks (CNN) or Recurrent Neural Networks (RNN), the choice of a model is based on data type, e.g., it is proven that CNN takes advantage over MLP and RNN when the dataset is formed by images [10] where is important to take into account the dimensional information.

Channel estimation (CE) is a technique used in communications for predicting how a transmitted signal will be affected as it travels through a propagation medium. It is crucial in communication systems to perfect the transmission quality by compensating the effects like attenuation and distortion. The introduction of pilot signals is necessary in order to estimate a channel. 5G uses Demodulation Reference Signals (DM-RS) for such purpose. Common pilot aided methods are Least Squares (LS) and Minimal Mean Squared Error (MMSE). However, their performance is compromised and complexity is increased in changing scenarios. For that reason, the DL techniques are acquiring a significant role in CE, bringing the possibility of having models that could adapt to changing channel conditions. Previous works developed on DL models for CE are described next. The Long Short-Term Memory (LSTM) models, a kind of RNN architecture, have outperformed the LS estimator in scenarios such as varying modulation, channel models, pilot sequences and number of antennas at the expenses

of a higher complexity [11–13]. The problem of channel estimation was addressed with CNN models [14], where the communication channel was handled as a resource grid while the CNN superiority compared with LS and practical estimation across all Tapped Delay Line (TDL) models were proven. Hybrid models like: Convolutional LSTM [15], Autoencoders [16, 17], and Faster Super-Resolution CNN (FSRCNN) [18] highlight the importance of exact CE with the aim of mitigating channel effects in a process similar to image denoising or super resolution in OFDM systems.

The aim of this study is to use a Convolutional Autoencoder (CAE) as a channel estimator in a proposed 5G communication system varying the channel model, the subcarrier modulation, Signal-to-Noise Ratio and the user's speed. The evaluation performance of the model is based on BER, EVM, estimation time, and MSE measurements. The paper contrasts the experimental outcomes with the LS and practical estimators in a SNR range of –10 dB to 20 dB.

This paper is organized as follows: section 2 presents the theoretical foundations, the proposed 5G communication system, the description of CAE architecture, and the hyper-parameters configuration for training using an own generated dataset. Section 3 presents the simulation results, varying the channel model, SNR values, modulation scheme and user's speed. This study concludes with section 4.

2 Materials and Methods

2.1 Communications System Architecture

In this paper, a Single-Input Single-Output OFDM system is built for the transport of the 5G Physical Down-link Shared Channel (PDSCH). Figure 1 illustrates the proposed block diagram.

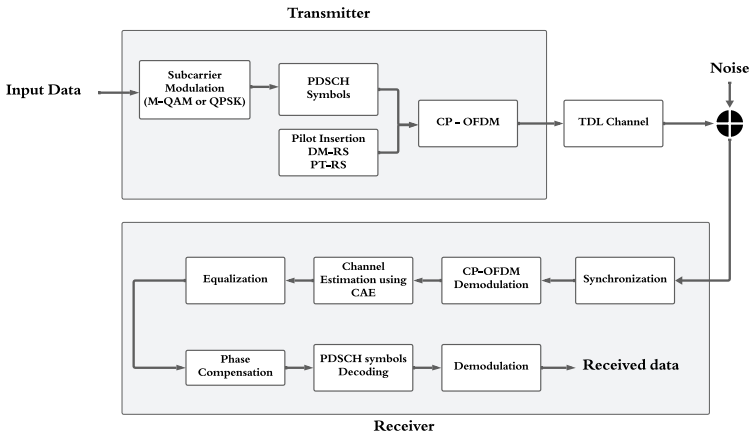


Fig. 1: Proposed block diagram for channel estimation in a 5G network

At the transmitter side, binary data is generated and transferred to the modulation block where it is mapped with the subcarrier modulation formats supported by 5G standard [19]: QPSK, 16QAM, 64QAM, and 256QAM. Next, the generated symbols are mapped to the resource grid, which is a bi-dimensional arrangement of time and frequency domains. In this paper, the channel estimation is pilot aided, for that reason, DM-RS and Phase Tracking Reference Signals (PT-RS) are also allocated in the resource grid. Figure 2 shows the distribution of the pilots and symbols in the resource grid.

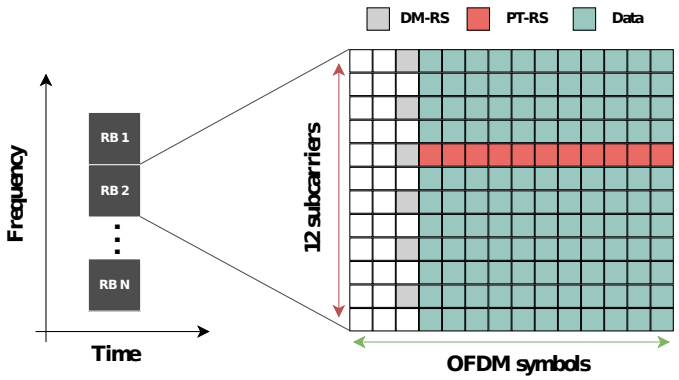


Fig. 2: Pilots and data allocation in the resource grid

The data is transmitted in small units known as subframes; each one has a duration of 1 ms. The structure presented in the previous figure shows that the Resource Block (RB) is composed of 12 subcarriers and the number of RBs in a subframe may vary depending on application scenarios. The number of OFDM symbols N_{sym} in a subframe for 5G is selected depending on the OFDM numerology. Additionally, the slot is another structure in the time domain, it is formed of 14 OFDM symbols. Finally, 10 subframes complete a frame.

The next stage in the transmitter is the CP-OFDM process. First, it is necessary to select a numerology which is the configuration of carrier parameters such as subcarrier spacing (SCS) and Cyclic Prefix (CP). SCS is the distance between two adjacent OFDM subcarriers and it is calculated with the equation 1, where μ is the selected numerology which have values from 0 to 4. CP is a guard interval that helps to mitigate the Inter-Symbol Interference (ISI) caused by multi-path propagation. The process consists of adding a copy of the symbol's last few samples at the beginning of the next ones, ensuring orthogonality between subcarriers. The type of CP can be *extended* (for $\mu = 2$) or *normal* (for the rest of μ).

$$\text{SCS} = 2^{\mu} \cdot 15 \text{ kHz} \tag{1}$$

The output of the CP-OFDM block is a base band signal. The Inverse Fast Fourier Transform (IFFT) is used to transform the modulated data from the resource grid $X(k)$, to $x(n)$, as given by equation 2, where k is the subcarrier index, l is the time index and N is the number of subcarriers.

$$x(n) = \sum_{k=0}^{N-1} X(k)\, e^{j\frac{2\pi\, k\, l}{N}} \tag{2}$$

The base band generated signal is transmitted through the wireless channel. In this paper, the channel model is selected from the 3GPP standard [20]. The channel includes the effects of multi-path and Doppler shifting, which causes frequency and time selective fading, respectively. In particular, the channel is a Tapped Delay Line (TDL) model, which applies signal processing techniques to generate variable time delays in the transmitted signal $x(n)$, each variation has different amplitude and phase in NLoS and LoS scenarios. The power delay profiles TDL-A, TDL-B, and TDL-C are used to represent NLoS; while, TDL-D and TDL-E decribe LoS conditions.

The level of mobility and the carrier frequency are related with the Doppler shift over the wireless channel. The maximum Doppler shift f_d is defined as follows

$$f_d = \frac{v\,\frac{1000}{3600}}{c_0}\, f_c \tag{3}$$

where v is the user's speed in km/h and f_c is the carrier frequency in hertz, with $c_0 \approx 3 \times 10^8$ being the speed of light. Equation 3 shows that in static scenarios, $f_d = 0$, nevertheless, as v and f_c increases, so does the Doppler shifting.

After transmitting across the channel, noise is added, therefore, the received OFDM signal is

$$y(n) = h(n) \circledast x(n) + \eta(n) \tag{4}$$

After time synchronization, Fast Fourier Transform (FFT) is performed at the receiver end obtaining

$$Y(k) = H(k) \circledast X(K) + \eta(K) \tag{5}$$

where $Y(k)$ and $X(k)$ denote the received and transmitted symbols in the kth subcarrier, respectively; $H(k)$ is the channel impulse response in the kth subcarrier and $\eta(K)$ is Additive White Gaussian Noise (AWGN). The pilot signals are extracted from the frequency-time domain signal and then used for channel estimation. After estimating the channel response, the received signal is equalized and compensated in phase. Both processes are fundamental because they mitigate the channel distortion and the phase between transmitted and received symbols. The data is taken from the resource grid and demodulated obtaining at the output a signal that consist of a binary data sequence.

2.2 Channel Estimation using Least Squares

Channel estimation is a key process in OFDM systems in which the communication channel is characterized and mathematically modeled. Some characteristics are the attenuation, propagation delay and impulse response $H(k)$. CE can be classified in: Pilot-Aided Channel Estimation (PACE), Blind Channel Estimation (BCE), and Decision Directed Channel Estimation (DDCE) [13]. This paper focuses in PACE techniques, especially LS estimation, which is widely used in wireless systems due to low complexity. Figure 3 illustrates the process made with PACE.

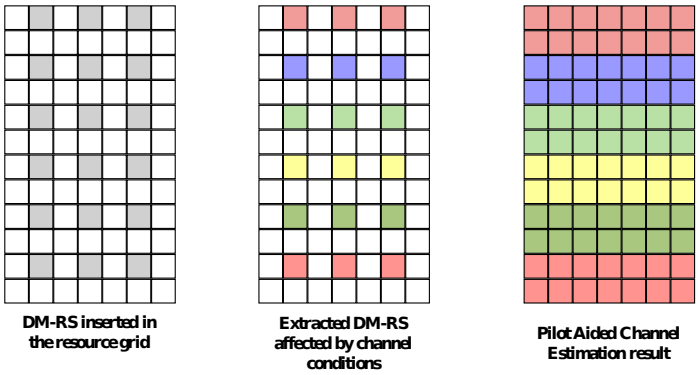


Fig. 3: Pilot Aided Channel Estimation

LS estimator assumes that the changes in the transmitted signals could be expressed with linear equations, hence, the channel coefficients are obtained with linear interpolation. LS algorithm is relatively simple, but has poor accuracy when the levels of noise and mobility are high. The received signals in LS are expressed with equation 5. However, the noise information $\eta(K)$ is not considered. Channel response is calculated by clearing $H(K)$ and extracting the pilots as follows

$$\hat{H}_{LS} = X^{-1}\, Y = \left(\frac{X_p}{Y_p}\right)^T \quad \text{where} \quad p = 0, 1, 2, \cdots, N-1 \tag{6}$$

where Y is the output pilot vector $Y_p = [y_0, y_1, y_2, \cdots, y_{N-1}]^T$, T is the transpose operation, X is the diagonal matrix of transmitted pilots, N is the number of pilots inserted across time domain, and $\hat{H}_{LS}$ is the estimated channel response.

There are other CE methods such as MMSE, which has more accuracy in calculations compared with LS. Also, the noise variance σ_η^2 is used, having a better performance in noisy environments. Nevertheless, the computation is very complex due to the use of second order statistics (auto-covariance matrix and cross covariance matrix with $N \times N$ dimension, being N the length of FFT).

2.3 Convolutional Autoencoder Based Channel Estimation

Convolutional Neural Network (CNN) is a class of Deep Learning algorithm that is widely used in image processing. CNN is composed of several layers that extract features of an input arrangement X, and depending on the application (classification or regression) generates an estimated output $\hat{Y}$. Between input and output layers are the hidden layers, the most common are:

- *Convolutional Layer:* Its main function is the extraction of features through the application of convolution operations $*$. The layer consists of several filters or *kernels*, which are small matrices of weights. The matrices slide over the input or previous feature maps and it does the convolution operation. The size of kernel $h(n)$ is $2k \times 2k$. Assuming an input X in a bi-dimensional scale where each pixel of the input has (x, y) coordinates, the convolution operation is expressed as follows

$$O(x,y) = \sum_{i=-a}^{a} \sum_{j=-b}^{b} h(i,j)\, X(x-i, y-j) \tag{7}$$

  where a and b are the width and height of the input. The size of the filter impacts in the model's complexity, when the size is higher the number of trainable parameters increases. Varying the size of the filter allow the extraction of features in different scales (local or general).
- *Pooling Layer:* The pooling consists of an algorithm that helps in reducing the spatial dimensions of the feature maps while retaining the most important information. As well as in convolutional layer, the pooling has a kernel that slides over the feature map and the selection of the important information is done with operations like: averaging, max, min, and global.
- *Rectified Linear Unit Layer:* This layer uses the Rectified Linear Unit (ReLU) activation function. It introduces non-linearity to the network, allowing to learn complex patterns or relationships in the dataset.

To overcome the drawbacks of the LS estimation approach, in this paper, it is designed a CAE which is a model based on CNN architecture, hence, it shares the same layers. Figure 4 shows the CAE architecture, which consists of two stages: coding and decoding. Coding layers are used to transform the input image into a compressed representation, here the size of filters are varied with the purpose of reduce the spatial dimension. In the output of the coding stage, it is obtained a latent space (a set of main features). The decoding stage is in charge of reconstructing the image based on the latent space. At this point, it is necessary to use up-sampling layers in order to reach the initial dimension. The estimated output of the CAE is compared with the real label, measuring the MSE.

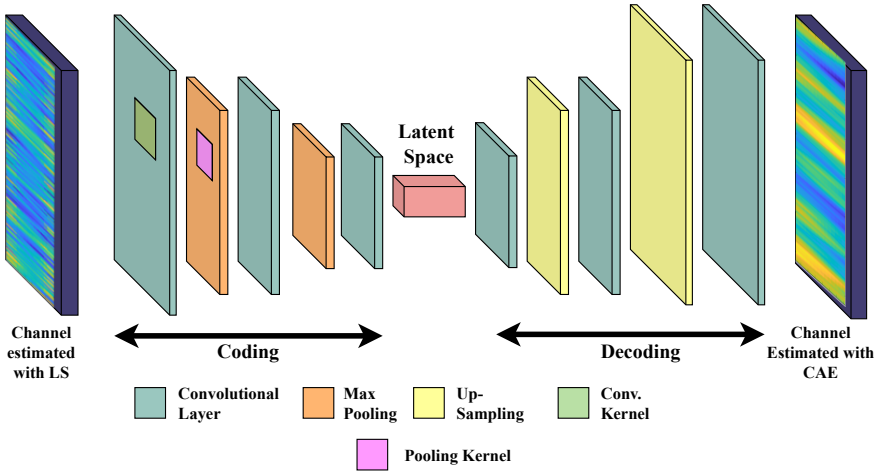


Fig. 4: Layers of the Convolutional Autoencoder

Some advantages of the CAE are listed below

- The CE can be addressed as denoising or super resolution situations, where the input X is the channel estimated with LS, while the output is a corrected image, closer to the real label (ideal channel estimation).
- The convolutional layers allow the use of multi-dimensional arrangements in the input. In this case, the resource grid has a $K \times L \times 2$ dimension, where K is the number of subcarriers, L is the number of OFDM symbols and the third dimension corresponds to the real and imaginary values of the channel coefficients.
- Using max operation in the pooling layer reduces the computational complexity of the model.
- The extraction of essential features contributes to remove redundant information like noise or repeated patterns in the dataset.

2.4 Model Training

The generation of the dataset is a key part to ensure the performance of the CAE estimation with various channel conditions. In order to have all the possible scenarios, the dataset is generated by randomly varying the TDL channel characteristics: power delay profile, delay spread, maximum Doppler shift and the level of noise. The parameters of the configured channel conditions are:

- *Power Delay Profile:* TDL-A, TDL-B, TDL-C, TDL-D, and TDL-E.
- *Delay spread:* From 1 ns to 300 ns.
- *Maximum Doppler shift:* From 1 Hz to 500 Hz. Using equation 3, the user's speed is between 0.21 km/h and 107.92 km/h.
- *Level of noise:* It is fixed using the SNR in dB, then, the range is between 0 dB to 20 dB.

The input X is composed of the multiple samples of the LS estimation, while the output Y is the ideal estimated channel. The dimension of the resource grid is given by the number of OFDM symbols and resource blocks, which are 14 and 52, respectively. Then, the dimension of the dataset is $(RB \times 12, 14, 2, N)$, where N is the number of samples.

Data pre-processing plays a key role before training any algorithm. The purpose of this process is transform the raw data into a more appropriate and understandable format. In the CAE it helps the weights' iterative adjustments, avoiding abrupt changes during back propagation stage. In the current dataset, it is necessary to split the channel coefficients in the real and imaginary values, hence, the length N is doubled. In addition, it is used zero-center normalization, guaranteeing the data mean in 0 and the standard deviation in 1. A detailed description of the proposed model training is found in pseudocode 1.

Algorithm 1 CAE based channel estimation

Require: LS estimated channel and ideal estimated channel.
Ensure: Convolutional Autoencoder.

1: Generate the dataset fixing the length N. During generation, the channel's characteristics are changed randomly. The dataset dimension is $(624, 14, 2, N)$.
2: Load the dataset. X and Y are the LS and ideal estimated channels, respectively.
3: Double the dataset length N, splitting the channel coefficients into the real and imaginary values. The new dimension is $(624, 14, 1, 2\,N)$.
4: Data segmented as training and test data. 75% in training and 25% in test. $\dim(X_{train}, Y_{train}) = (624, 14, 1, 1.5\,N)$ and $\dim(X_{test}, Y_{test}) = (624, 14, 1, 0.5\,N)$.
5: Define training hyper-parameters of CAE. Shown in table 1
6: Form CAE layers.
7: Train the CAE until reach the minimum of loss function (MSE) between estimated output $\hat{Y}$ and ideal estimated channel Y.
8: Save the CAE model.

3 Simulation Results

This section presents the performance of the proposed CAE over the 5G modeled channel and compare it with the LS and practical estimation. First, the settings for the simulation are described and then the results for the evaluation scenarios are presented and analyzed.

3.1 Simulation Configuration

The parameters of the 5G wireless environment are listed in table 2. To evaluate the performance of the CAE in the proposed OFDM block diagram shown in Fig. 1, the power delay profiles are TDL-B and TDL-E, which are modeled as NLoS and LoS

Table 1: Hyper-parameters for training CAE

Hyper-parameter	**Value**
Length of dataset N	20000 [1]
Size of training data	15000
Size of test dataset	5000
Learning rate	3×10^{-4}
Number of epochs	10
Iterations per epoch	46
Loss function	Mean Squared Error
Optimizer	ADAM

[1] After applying pre-processing

environments, respectively. The down-link carrier frequency is 5 GHz. The number of subcarriers is 624 and the subcarrier spacing is set in 15 kHz ($\mu = 0$). The data is transmitted during 100 ms (10 frames) and it is modulated with QPSK and 256QAM. Finally, the user's speed has values of 5 km/h and 100 km/h.

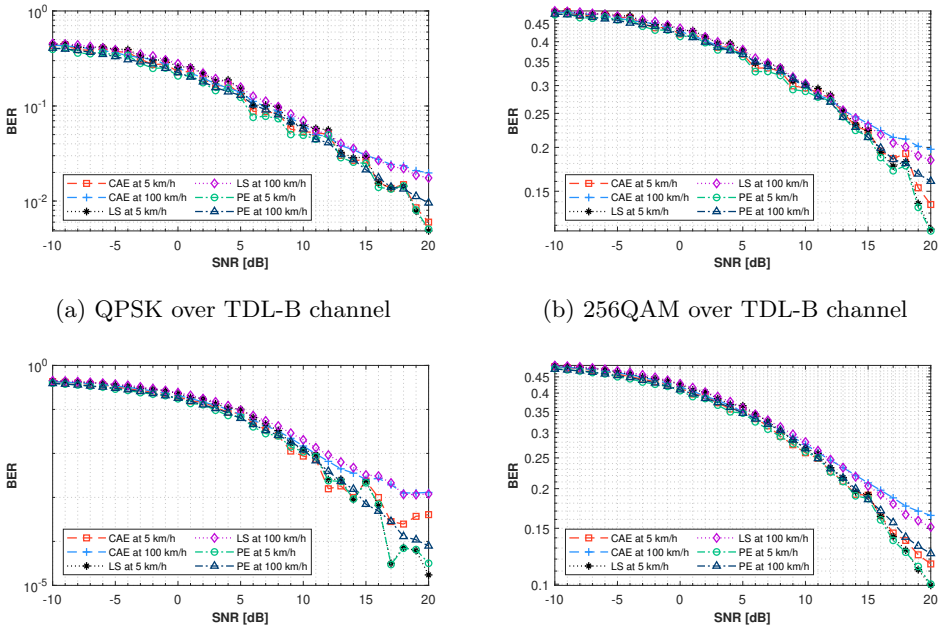
Table 2: Simulation parameters

Parameter	**Value**
Downlink carrier frequency	5 GHz
Resource blocks	52
Number of subcarriers	624
Subcarrier Spacing	15 kHz
OFDM symbols	14 with normal CP
Power Delay Profile	TDL-B (NLoS) and TDL-E (LoS)
SNR	-10 dB to 20 dB
User's speed	5 km/h and 100 km/h
Maximum Doppler Shift	23.16 Hz and 463.28 Hz
Delay Spread	300 ns
Physical Channel	PDSCH mapping A
Modulation scheme	QPSK and 256 QAM
Transmitted data	10 frames

3.2 Evaluation

To evaluate the performance of the proposed CAE estimator, the simulation was carried out considering four different scenarios: combining NLoS and LoS environments with the modulations listed in table 2. In each scenario, the performance of the CAE estimator, LS estimator, and Practical Estimator (PE) was compared by utilizing BER, EVM, MSE, and estimation time versus SNR in dB. Additionally, the comparison includes the responses when the relative movement between transmitter and receiver is at low or high-speed.

Figure 5 shows the BER of the aforementioned estimators in the four scenarios. The estimation of each model led to the BER declining gradually as the SNR increased. The modulation format used for transmission affects the received bits, this occurred due to the robustness to noise, whereas the symbols in QPSK constellation diagram are widely spaced apart, the 256QAM are closer, generating a worst response with noisy conditions. The change in the power delay profile shows that in a LoS scenario the performance is slightly better, the main reason is because the signal propagates directly from the transmitter to the receiver, avoiding multi path losses or delays generated by natural or artificial obstacles, this improves the number of correct received bits. The effect of speed in each estimator is produces a gap when the speed increases, this occurs from roughly a SNR higher than 12 dB. Before this value, the curves of low and high-speed estimations are overlapped.



(a) QPSK over TDL-B channel

(b) 256QAM over TDL-B channel

(c) QPSK over TDL-E channel

(d) 256QAM over TDL-E channel

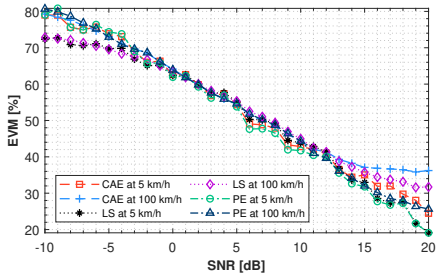
Fig. 5: BER with proposed channel estimators vs. SNR level

From the range between -10 dB and 15 dB the LS estimator yielded the highest BER in both low-speed and high-speed variations. After that limit, the LS estimation improves gradually, even, at 20 dB, it has a better performance compared with CAE, indicating an accurate estimation in low noise conditions. On the contrary, the PE has the best performance, having the lowest BER values in the four scenarios. The advantage of PE is that it exploits the received grid, the original resource grid, and the carrier parameters to generate the estimated channel, while LS only uses

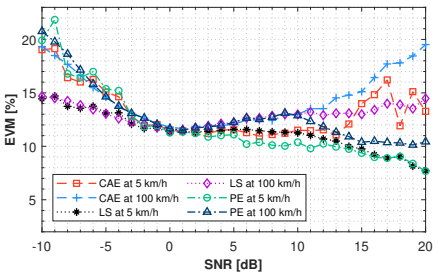
the indices and symbols of the pilot signals. On the other hand, the proposed CAE outperforms LS estimator in the SNR range of -10 dB and 15 dB. This is an important fact because the generated dataset did not have negative SNR values, showing the generalization of a deep learning algorithm. Both PE and CAE have remarked differences from the SNR value of 15 dB as seen in Fig. 5c. For the previous levels of noise and the other scenarios with low-speed and QPSK configurations, both have similar values. Finally, the BER metric shows that PE has a better estimation with a higher maximum Doppler shift.

In Fig. 6 is presented the EVM performance of each estimator. The trend of EVM with QPSK is to decrease gradually when the level of noise reduces, indeed, this trend is similar in both TDL-B and TDL-B power delay profiles. Nevertheless, such trend does not continue when the data is transmitted in 256QAM as can be seen through the peaks around 8 dB in Fig. 6b and 9 dB in Fig. 6d. This is a consequence of the higher density of points in the constellation and the noise sensitivity, where small changes in the channel conditions produce interference between received symbols. The analysis of EVM plots could be done in three ranges: from -10 dB to -3 dB the LS estimator has better performance independently of the channel condition and the deployed modulation; from -3 dB to 3 dB all the proposed estimators overlapped and the changes are not significant; the third range comprises the values between 3 dB and 20 dB, there the trend of the PE continues while it starts to deteriorate the performance in the other estimators. This effect is not evident in QPSK, but in 256QAM it does, specially from 12 dB, where the CAE has the worst performance, followed by the LS estimator. This can be explained by the fact that the loss function has been determined to minimize the errors in the estimation instead of the EVM metric.

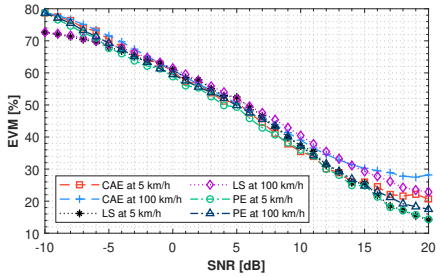
Figure 7 presents the MSE metric with the aforementioned scenarios. All the four scenarios share the same trend of MSE decreasing while SNR increases, all estimations tend to zero. The effect of speed does not affect the quality of estimation, for each model both low and high-speed curves do not have significant changes, indicating that MSE is independent of the maximum Doppler shift. Once again, the LS performed worst until 5 dB, being approximately 6 and 12 times higher compared with CAE and PE, respectively. The CAE outperformed LS estimation in all the simulation range, this probes that the loss function used during training was the MSE, even, when the negative values were not included during the learning stage. The difference between CAE and PE is evident from -10 dB to 0 dB, but after this limit, both are similar.



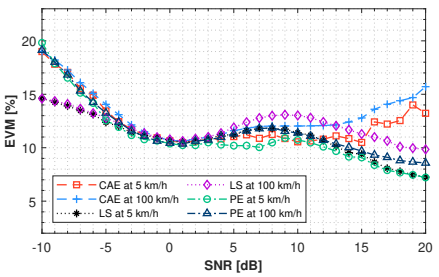
(a) QPSK over TDL-B channel



(b) 256QAM over TDL-B channel

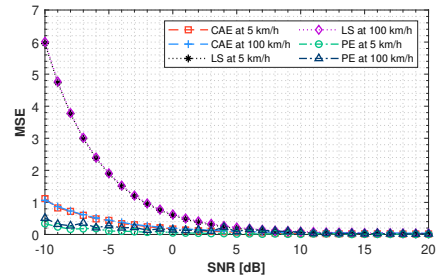


(c) QPSK over TDL-E channel

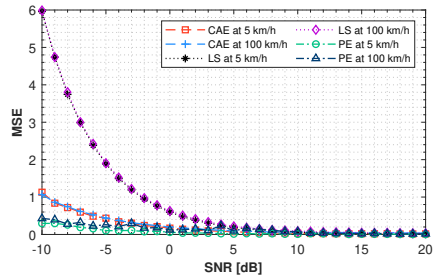


(d) 256QAM over TDL-E channel

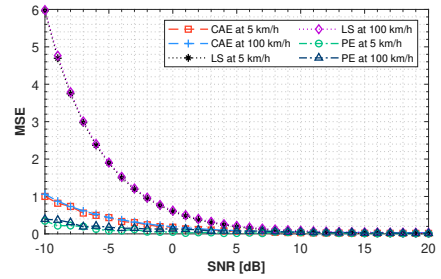
Fig. 6: EVM with proposed channel estimators vs. SNR level



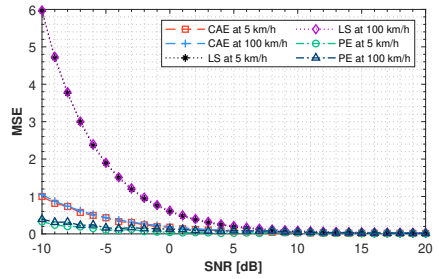
(a) QPSK over TDL-B channel



(b) 256QAM over TDL-B channel



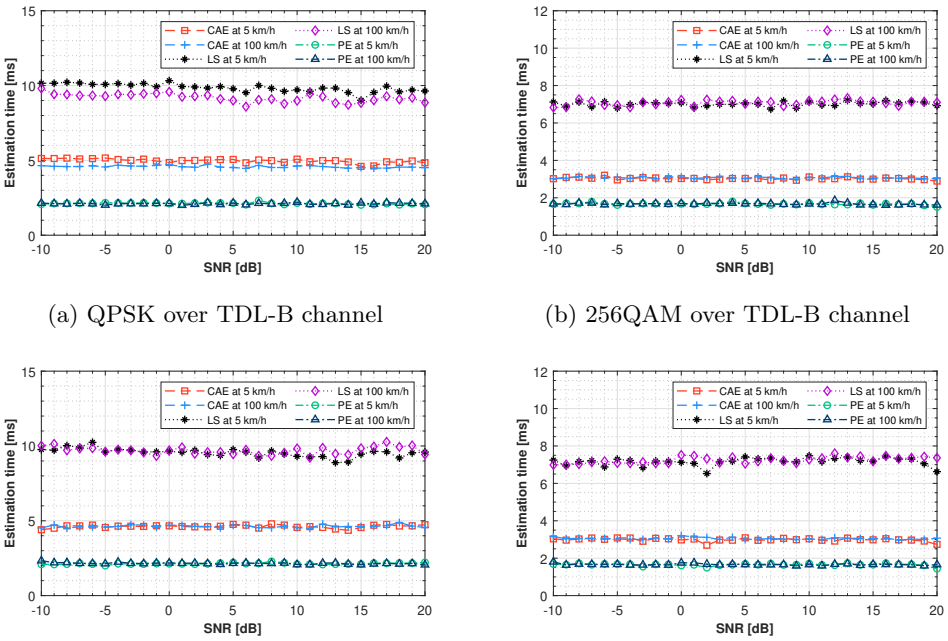
(c) QPSK over TDL-E channel



(d) 256QAM over TDL-E channel

Fig. 7: MSE of channel estimates vs. SNR level

Finally, Fig. 8 illustrates the time that took to each model to generate an estimated channel. The estimation time is independent of SNR, it remains almost constant during the simulations. The changes between the times occurs due to the modulation used for frames transmission, hence, for 256QAM it took less time compared with QPSK. This can be explained by the spectral efficiency, the same data is transmitted faster with high order modulations, because the symbols represent more bits. As it happened with MSE, the maximum Doppler shift does not affect the estimation time. LS estimator requires more time to generate an output, while PE is faster. The CAE estimates the channel with QPSK and 256QAM in an average time of 4.76 ms and 3.54 ms, respectively. The estimation time in CAE is less compared with LS, this shows the importance of max pooling layers, which reduce the complexity in the model.



(a) QPSK over TDL-B channel

(b) 256QAM over TDL-B channel

(c) QPSK over TDL-E channel

(d) 256QAM over TDL-E channel

Fig. 8: Estimation time of channel estimates vs. SNR level

4 Conclusion

In this paper, the use of a CAE as a technique for pilot-aided channel estimation process in a SISO OFDM system with different scenarios deployed over TDL-B and TDL-E modeled channels defined in the 5G standard was presented. The proposed CAE was trained with a dataset varying the channel conditions randomly, the input

was the LS estimation and the target was the ideal channel estimation. The performance of the proposed technique was compared with LS and PE estimations in terms of BER, EVM, MSE, and estimation time as a function of the SNR level by changing user's speed and subcarrier modulation. The results show that CAE outperforms the LS estimation in the BER and MSE metrics in a SNR range between -10 dB and 15 dB. Furthermore, the time required to estimate the channel is lower than LS in all the scenarios caused by the addition of max pooling layers. When the relative movement of the receiver is lower and the modulation used is QPSK, the BER and EVM of the CAE are similar to PE values, despite using only the information of the pilot signals and not the reference resource grid and carrier parameters. Finally, the MSE performance have not significant changes compared to PE from 0 dB to 20 dB. These results support that deep learning models have a high generalization ability and robustness with noisy conditions, being able to estimate a channel with high accuracy and low latency in 5G networks and beyond.

Some suggestions for future related works include the use of regularization methods to reduce the complexity of the model and improve the performance across a wider range increasing the number of samples and the SNR limits.

Acknowledgments. The authors wish to acknowledge and thanks the Universidad Distrital Francisco José de Caldas for supporting the development of this paper.

Conflicts of interest. The authors declare no conflicts of interest.

References

[1] Sudhamani, C., Roslee, M., Tiang, J.J., Rehman, A.U.: A survey on 5g coverage improvement techniques: Issues and future challenges. Sensors **23**(4) (2023) https://doi.org/10.3390/s23042356

[2] Pana, V.S., Babalola, O.P., Balyan, V.: 5g radio access networks: A survey. Array **14**, 100170 (2022) https://doi.org/10.1016/j.array.2022.100170

[3] Valgas, J., Monserrat, J.F., Arslan, H.: Flexible numerology in 5g nr: Interference quantification and proper selection depending on the scenario. Mobile Information Systems **2021**, 6651326 (2021) https://doi.org/10.1155/2021/6651326

[4] Sarker, I.H.: Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions. SN Computer Science **2**(6), 420 (2021) https://doi.org/10.1007/s42979-021-00815-1

[5] Jiang, X., Hu, Z., Wang, S., Zhang, Y.: Deep learning for medical image-based cancer diagnosis. Cancers **15**(14) (2023) https://doi.org/10.3390/cancers15143608

[6] Dellal-Hedjazi, B., Alimazighi, Z.: Deep learning for recommendation systems. In: 2020 6th IEEE Congress on Information Science and Technology (CiSt), pp.

90–97 (2020). https://doi.org/10.1109/CiSt49399.2021.9357241

[7] Alkhudaydi, O.A., Krichen, M., Alghamdi, A.D.: A deep learning methodology for predicting cybersecurity attacks on the internet of things. Information **14**(10) (2023) https://doi.org/10.3390/info14100550

[8] Bhalla, A., Nikhila, M.S., Singh, P.: Simulation of self-driving car using deep learning. In: 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), pp. 519–525 (2020). https://doi.org/10.1109/ICISS49785.2020.9315968

[9] Sarker, I.H.: Machine learning: Algorithms, real-world applications and research directions. SN Computer Science **2**(3), 160 (2021) https://doi.org/10.1007/s42979-021-00592-x

[10] Dovbnych, M., Plechawska–Wójcik, M.: A comparison of conventional and deep learning methods of image classification. Journal of Computer Sciences Institute **21**, 303–308 (2021) https://doi.org/10.35784/jcsi.2727

[11] Wong, K.J., Juwono, F.H., Reine, R.: Deep learning for channel estimation and signal detection in ofdm-based communication systems. ELKHA **14**(1), 52–59 (2022)

[12] Mohammed, A.S.M., Taman, A.I.A., Hassan, A.M., Zekry, A.: Deep learning channel estimation for ofdm 5g systems with different channel models. Wireless Personal Communications **128**(4), 2891–2912 (2023) https://doi.org/10.1007/s11277-022-10077-6

[13] Dash, L., Thampy, A.S.: Channel estimation using hybrid optimizer based recurrent neural network long short term memory for mimo communications in 5g network. SN Applied Sciences **5**(2), 60 (2023) https://doi.org/10.1007/s42452-022-05253-z

[14] Gupta, R., Gupta, J.: Channel estimation of 5g new radio using convolutional neural network. In: 2022 8th International Conference on Signal Processing and Communication (ICSC), pp. 58–62 (2022). https://doi.org/10.1109/ICSC56524.2022.10009096

[15] Wang, Y., Chang, J., Lu, Z., Yu, F., Wei, J., Xu, Y.: Channel estimation of 5g ofdm system based on convlstm network. In: 2022 7th International Conference on Communication, Image and Signal Processing (CCISP), pp. 62–66 (2022). https://doi.org/10.1109/CCISP55629.2022.9974588

[16] Yamada, Y., Ohtsuki, T.: Autoencoder-based pilot pattern design for cdl channels. In: 2022 27th Asia Pacific Conference on Communications (APCC), pp. 76–80 (2022). https://doi.org/10.1109/APCC55198.2022.9943621

[17] Thakkar, K., Goyal, A., Bhattacharyya, B.: Deep learning and channel estimation. In: 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 745–751 (2020). https://doi.org/10.1109/ICACCS48705.2020.9074414

[18] Wang, M., Wang, A., Liu, Z., Chai, J.: Deep learning based channel estimation method for mine ofdm system. Scientific Reports **13**(1), 17105 (2023) https://doi.org/10.1038/s41598-023-43971-5

[19] 3GPP: User equipment (ue) conformance specification; radio transmission and reception; part 1: Range 1 standalone. In: 3GPP TS 38.521-1 Version 15.1.0 Release 15 (2019)

[20] 3GPP: Study on channel model for frequencies from 0.5 to 100 ghz. In: TR 38.901 Version 16.1.0 Release 16 (2020)