JOURNAL OF BASIC SCIENCE AND ENGINEERING

Channel Estimation for MIMO-OFDM Communication Systems Using Machine learning for 5G applications

Avinash S Joshi[1], Dr. Gopal A. Bidkar[2]

[1] Asst. Professor, Dept. of ECE, KLS's Vishwanathrao Deshpande Rural Institute of Technology, Haliyal, INDIA.

[2] Professor, Dept. of ECE, Shri Dharmastala Manjunatheshwar College of Engineering and Technology, Dharwad, INDIA.

DOI: https://doie.org/10.0612/Jbse.2024946272

Abstract

The system performance of wireless networks heavily relies on channel estimation. Moreover, the application of deep learning has shown substantial advancements in improving communication reliability and decreasing the computational complexity of 5G and beyond networks. While least squares (LS) estimation is widely employed for obtaining channel estimates due to its cost-effectiveness and independence from prior statistical information about the channel, it is associated with a relatively high estimation error. The presented paper suggests a novel channel estimation framework leveraging deep learning to enhance the accuracy of channel estimates obtained through the least squares (LS) approach. The objective is realized by employing a MIMO (multiple-input multiple-output) system with a multi-path channel profile, simulating scenarios in 5G and beyond networks, considering the level of mobility indicated by Doppler effects. The construction of the system model is applicable to any number of transceiver antennas, and the machine learning module is designed to be versatile, allowing the utilization of various neural network architectures. The numerical findings illustrate the effectiveness of the newly introduced deep learning-based channel estimation framework compared to conventional methods widely employed in prior research. Furthermore, among the examined artificial neural network architectures, bidirectional long short-term memory exhibits the highest quality in channel estimation and the lowest bit error ratio.
Keywords: Machine learning; channel estimation; MIMO-OFDM; frequency selective channels

1. Introduction

In contemporary communication technology, the utilization of Multiple Input Multiple Output (MIMO) for wireless communication represents a significant technical breakthrough. MIMO systems are characterized by equipping both the transmitting and receiving ends with multiple antenna elements. The fundamental idea behind MIMO is the combination of signals transmitted from all transmit antennas at each receive antenna element, aiming to enhance the Bit Error Rate (BER) performance or data rate for communication, ultimately improving the overall communication quality for each MIMO user. Leveraging this advantage can lead to a substantial increase in both the Quality of Service (QoS) of the networks and the revenues for the operator. Wireless systems employing multiple antennas offer a promising foundation for achieving high data rates due to the enhanced bit per symbol capacity when compared to Single Input Single Output (SISO), Multiple Input Single Output (MISO), and Single Input Multiple Output (SIMO) systems. The demand for high-data-rate wireless access is prevalent across various applications. Traditionally, achieving higher data rates in transmission has necessitated the allocation of additional bandwidth.

In rich multipath environments, wireless communication systems have the potential to attain substantial channel capacities by capitalizing on the additional spatial dimension. The attainable capacity and performance are contingent on both channel conditions and the structure of the transmit signal. Achieving this objective is influenced by the design of the MIMO system architecture, which impacts the complexity of both the transmitter and, notably, the receiver.

The MIMO coding technique can be categorized into three groups: Space–time coding (STC), space division multiplexing (SDM), and beamforming. At the heart of MIMO systems lies the fundamental concept of Space–Time Processing. Time serves as the inherent dimension for digital communication data, while space pertains to the spatial dimension involved in employing multiple spatially distributed antennas. Based on the literature survey, existing channel estimation techniques have not yielded satisfactory results. Therefore, we present an innovative approach that utilizes a hybrid of machine learning techniques (CNN, RNN) for channel estimation with the goal of enhancing the Mean Squared Error (MSE) and reducing the bit error rate (BER) in the MIMO-OFDM system. This approach aims to improve the overall performance and adaptability of the system in real-world scenarios.

Objectives

- To design and implement a hybrid of hybrid of machine learning (CNN, RNN) or artificial intelligence-based channel estimation method. This approach aims to significantly improve the accuracy of channel estimation and reduce both improve Mean Squared Error (MSE) and bit error rate (BER) metrics, thereby enhancing the overall performance and reliability of the MIMO -OFDM system.
- The second objective is to conduct a comprehensive comparative analysis between the proposed machine learning or artificial intelligence-based channel estimation method

2. Methodology

To have introduced an innovative approach that capitalizes on a hybrid deep learning framework, combining Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), to enhance the precision of channel estimation. To focus lies in augmenting the Mean Squared Error (MSE) and minimizing the Bit Error Rate (BER) within MIMO-OFDM systems. This is achieved through a systematic three-step process:

- **Data Generation and MIMO-OFDM Setup:** In the initial phase, we construct training data by implementing a MIMO-OFDM system encompassing both the transmitter and receiver components. This holistic setup serves as the foundation for subsequent stages.
- **Hybrid Deep Learning Training:** The second step revolves around training a hybrid deep learning architecture, which seamlessly integrates both CNN and RNN components. This hybrid approach is adept at channel estimation tasks. The CNN aspect excels in extracting spatial features from the data, while the RNN aspect excels in capturing

Vol. 21, No. 1, (2024)
ISSN: 1005-0930

temporal dependencies. By combining their strengths, we achieve an optimized channel estimation model.

- **Integrated System Implementation:** The final stage entails the integration of the MIMO-OFDM transmitter and receiver with the hybrid CNN-RNN channel estimation model. This comprehensive system brings together the advancements made in the first two steps. As a result, it incorporates enhanced channel estimation capabilities, thereby leading to improved MSE and BER metrics within the MIMO-OFDM framework.

Through this innovative three-step methodology, we effectively address the challenges of accurate channel estimation in MIMO-OFDM systems. The integration of hybrid deep learning not only bolsters the accuracy of estimation but also aligns with the overarching goal of minimizing MSE and BER for optimized communication performance.

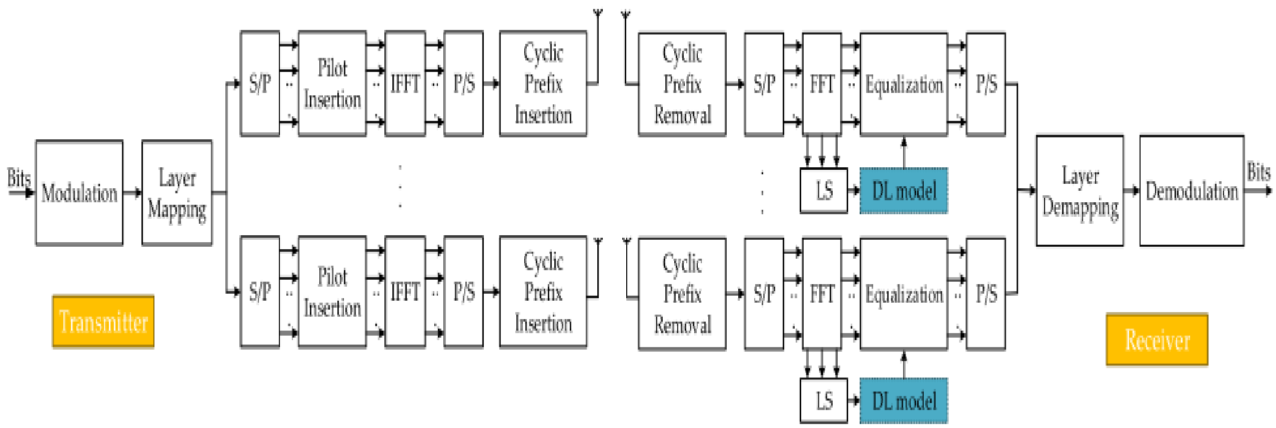

Figure 1: Block diagram

3. Implementation

Deep learning architecture using Long Short-Term Memory (LSTM) networks for sequence data classification. Let's break down the architecture details:

Input Layer:
Sequence Input Layer(256): This layer defines the input for sequences of data, where each sequence has a length of 256. It's suitable for handling sequences of data points, such as time-series data or natural language sequences.

LSTM Layer:
lstm Layer(100, 'Output Mode', 'last'): This is an LSTM layer with 100 hidden units. LSTMs are a type of recurrent neural network (RNN) designed to handle sequence data by capturing long-range dependencies. The 'Output Mode', 'last' parameter setting means that only the output corresponding to the last time step of the sequence will be used as the LSTM's output.

Fully Connected Layer:
Fully Connected Layer(4): This layer is a fully connected (dense) layer with 4 neurons. It takes the output from the LSTM layer and processes it using standard feedforward

Vol. 21, No. 1, (2024)
ISSN: 1005-0930

connections. This layer can perform transformations on the learned features before passing them to the final output layer.

Softmax Layer:

Softmax Layer: The softmax layer performs the final transformation on the output of the previous layer. It converts the raw scores into a probability distribution over the classes. In this case, it will provide the probabilities for the 4 classes of the classification task.

Classification Layer:

classification Layer: This is the final layer of the network and is specific to the classification task. It applies a softmax activation function to the output of the previous layer and computes the loss for the classification task.
the architecture processes sequences of length 256 through an LSTM layer with 100 hidden units. The output of the LSTM layer is then passed through a fully connected layer, followed by a softmax layer for probability computation, and finally, the classification layer for loss computation in a classification task with 4 classes.

Convolutional Neural Network (CNN) architecture for image classification. Let's break down the architecture details:

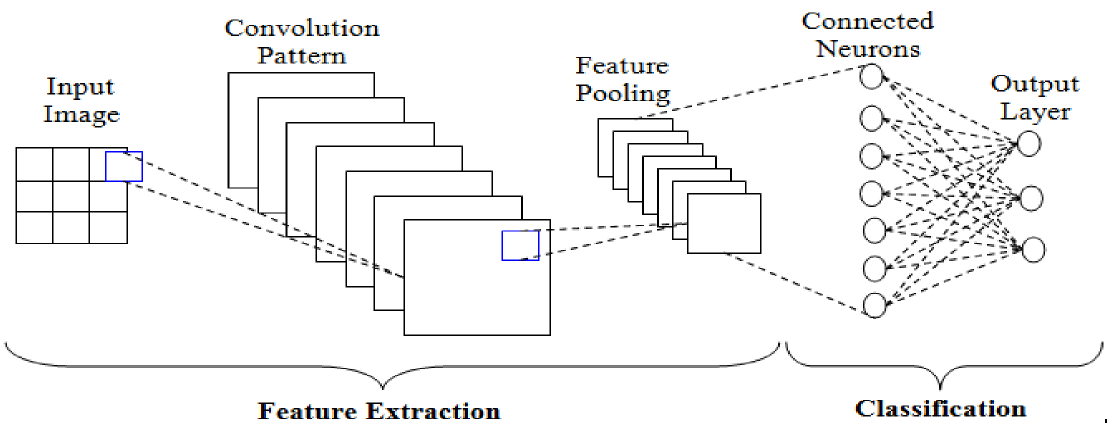

Figure 1: Basic CNN Architecture Diagram

Input Layer:
Image Input Layer([16 16 1]): This layer defines the input for images. Each image is expected to have a size of 16x16 pixels with a single channel (grayscale). The input layer sets the dimensions for the images that will be fed into the network.

Convolutional Layer:
convolution2d Layer(5, 20): This is a 2D convolutional layer with 20 filters of size 5x5. Convolutional layers apply filters to the input image to extract features. The 20 filters in this layer will detect various patterns and features in the input images.
Rectified Linear Unit (ReLU) Layer:

Vol. 21, No. 1, (2024)
ISSN: 1005-0930

relu Layer: ReLU activation functions introduce non-linearity into the network. They apply an element-wise function that sets all negative values to zero and leaves positive values unchanged. This helps the network learn complex relationships in the data.
Max Pooling Layer:
maxPooling2d Layer(2, 'Stride', 2): Max pooling layers downsample the spatial dimensions of the feature maps while retaining the most important information. This layer performs max pooling with a 2x2 window and a stride of 2, effectively reducing the spatial dimensions by half.
Fully Connected Layer:
fully Connected Layer(4): This is a fully connected (dense) layer with 4 neurons. It takes the output from the previous layers and processes it using standard feedforward connections. This layer can perform transformations on the learned features before passing them to the final output layer.
Softmax Layer:
softmax Layer: Similar to the previous explanation, this layer computes the softmax activation to produce a probability distribution over the classes.
Classification Layer:
Classification Layer: This is the final layer of the network, specifically designed for classification tasks. It applies the softmax activation function to the output of the previous layer and computes the loss for the classification task.

In summary, the architecture takes 16x16 grayscale images as input and processes them through a sequence of layers. The convolutional layer extracts features, followed by ReLU activation for non-linearity. Max pooling reduces spatial dimensions, and a fully connected layer processes the features before classification. The softmax layer produces class probabilities, and the classification layer computes the loss for training.

4. Results and Discussions

In order to train and test the FDNN model, a set of data with 255, and 860 realizations was gathered. We used 70% of the data for training, 15% as the validation set, and 15% of the data for testing. For the CNN model and bi-LSTM model, we used a data set of 11,000 realizations with the same proportions for the training set, validation set, and test set as the FDNN. The parameters for training those models are shown in Table 1.

Table 1: Parameters for training deep learning models.

Training Option	Value
Optimizer	adam
Epochs	100
Mini Batch size	1000
Learning Rate	0.01
Learn rate Drop Factor	0.1

Vol. 21, No. 1, (2024)
ISSN: 1005-0930

To evaluate the performance of the proposed estimators, the simulation was carried out and the results compared with the conventional LS estimation and LMMSE estimation by utilizing the bit error rate (BER) and mean square error (MSE) versus signal to noise ratio (SNR).

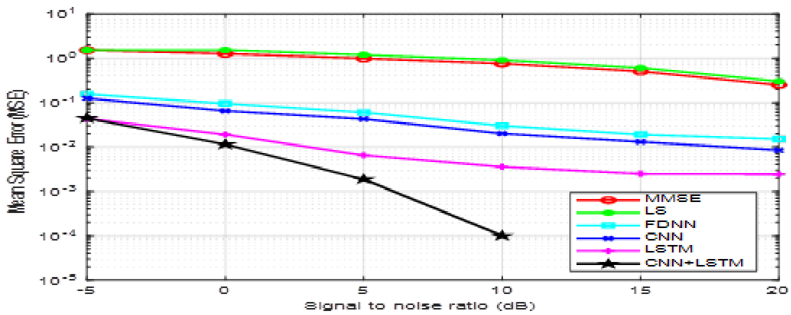

Figure 2: SNR to MSE

Figures 2 show the MSE of different channel estimations in the first and second scenarios, respectively. The 16-QAM (quadrature amplitude modulation) method was deployed to modulate the transmitted data in the simulation. The channel estimation methods led to the MSE declining gradually as the SNR increased. In both the scenarios, LS estimation yielded the worst MSE performance, which was because it does not take the statistical channel information into account when performing the channel estimation. On the contrary, LMMSE estimation exploits the mean and covariance matrices, which resulted in better MSE performance than its LS counterpart. Our proposed deep learning estimators yielded the best MSE performance compared to the conventional methods.

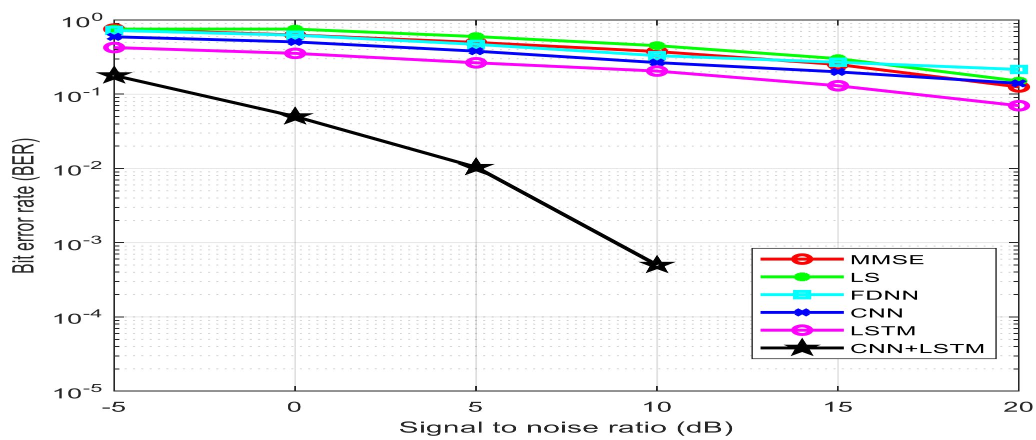

Figure 3: SNR to BER

it is also provide the BER performance of the considered scenarios in Figure 3 with the different channel estimation methods, respectively. The trend of the BER performance for the examined estimators is similar to that of MSE performance. However, in both scenarios, the BER performance of the FDNN model is slightly worse than the CNN+LSTM method at SNR = 10 dB. This can be explained by the fact that the loss function has been defined to minimize the channel estimation errors instead of the BER metric.

5. Conclusions

In this paper, the presented various DNN structures, including a fully-connected DNN, CNN, and bi-LSTM, to aid the channel estimation process in a MIMO-OFDM system under different fading multi-path channel scenarios modeled by the TDL-C model defined in 5G networks. The proposed CNN+LSTM-based channel estimation framework was trained using channel estimates from least squares estimation alongside corresponding perfect channels to determine the parameters as weights and biases. Utilizing the QAM modulation scheme, we compared the performance of the proposed estimations with conventional LSTM and CNN estimations in terms of channel estimation error and bit error ratio as functions of SNR levels. As the channel properties were effectively learned, we observed that the proposed hybrid deep learning-aided estimations significantly reduced the channel estimation error and bit error ratio. Among the proposed deep learning-hybrid approaches, the LSTM and CNN models showed the greatest reduction in channel estimation error due to their ability to exploit the time and frequency correlation among the channels. Furthermore, the proposed deep learning-based channel estimation methods demonstrated robust performance across varying pilot densities and changes in Doppler frequency.

References

1. Le HA, Van Chien T, Nguyen TH, Choo H, Nguyen VD. Machine Learning-Based 5G-and-Beyond Channel Estimation for MIMO-OFDM Communication Systems. Sensors. 2021; 21(14):4861. https://doi.org/10.3390/s21144861
2. Le Ha, A.; Van Chien, T.; Nguyen, T.H.; Choi, W. Deep Learning-Aided 5G Channel Estimation. In Proceedings of the 2021 15th International Conference on Ubiquitous Information Management and Communication (IMCOM), Seoul, Korea, 4–6 January 2021; pp. 1–7.
3. Van Chien, T.; Canh, T.N.; Björnson, E.; Larsson, E.G. Power Control in Cellular Massive MIMO with Varying User Activity: A Deep Learning Solution. IEEE Trans. Wirel. Commun. 2019, 19, 5732–5748.
4. Abdul Karim Gizzini, M.C.; Ahmad Nimr, G.F. Deep Learning Based Channel Estimation Schemes for IEEE 802.11p Standard. IEEE Access 2020, 8, 113751–113765.