

Simplified Temporal Convolutional-based Channel Estimation for a WiFi Vehicular Communication Channel

Simbarashe Aldrin Ngorima^{1,2,3}, Albert Helberg¹, Marelie H. Davel^{1,2,3}

¹Faculty of Engineering, North-West University, South Africa

²Centre for Artificial Intelligence Research, South Africa

³National Institute for Theoretical and Computational Sciences, South Africa

36450057@mynwu.ac.za

albert.helberg@nwu.ac.za, marelie.davel@nwu.ac.za

Abstract—Channel estimation in vehicular communication is a crucial element in the advancement of intelligent transportation systems. However, the use of pilot signals in the IEEE 802.11p standard is insufficient for accurate channel estimation in high-mobility scenarios. Data pilot-aided (DPA) estimation helps address this, but suffers from demapping errors. We propose a simplified Temporal Convolutional Network-based estimator (DPA-TCN) trained on a mixed signal-to-noise ratio dataset to improve estimation performance and reduce computational complexity. Our DPA-TCN estimator achieves a bit error rate comparable to a state-of-the-art long-short-term memory network with DPA and temporal averaging (LSTM-DPA-TA) while reducing the complexity of the model by approximately 65%.

Index Terms—channel estimation, deep learning, IEEE 802.11p, TCN, vehicle-to-vehicle, wireless communications

I. INTRODUCTION

In intelligent transportation systems (ITS), ensuring reliable and efficient vehicular communication is crucial for enhancing overall transportation safety and performance. Accurate and timely data transmission between vehicles and infrastructure is heavily dependent on precise channel estimation. However, this task is challenging due to the dynamic nature of vehicular channels. The channel in these networks is considered doubly selective, indicating that the channels vary in time and frequency. These variations are caused by two main factors: multipath propagation, where signals reflect off various surfaces, and the high mobility of vehicles, leading to rapid changes in the communication environment. Together, these factors complicate the process of accurate channel estimation, making it a significant challenge in the development of ITS. The IEEE 802.11p standard [1], approved and specified for vehicular communication to support ITS, faces limitations in scenarios that have high mobility. Specifically, the number of pilot subcarriers allocated in this standard tends to be insufficient, and relying solely on these pilots has proven to be less effective [2].

Increasing the number of pilots for transmission reduces spectral efficiency [3]. To address the problem of insufficient pilots, a method known as data pilot-aided (DPA) estimation is commonly used [4]. This technique involves using data

subcarriers for channel estimation. However, the presence of noise and multipath effects causes demapping errors which make it difficult to obtain accurate data pilots using only DPA as an estimation method [5].

In recent years, researchers have suggested using estimators that use deep learning (DL) to capture important features in vehicular channels [2], [3], [6], [7]. These estimators are based on the DPA method. Among these DL estimators, LSTM-DPA-TA [8] is one of the state-of-the-art (SOTA) in terms of bit error rate (BER) performance. However, the computational complexity of this estimator restricts the adaptation of this estimator in real-world applications.

In this work, we propose a simplified Temporal Convolutional Network (TCN) estimator that takes DPA estimation as input for the estimation of vehicular channels. Our aim is to address both the computational and BER performance challenges in vehicular channels by leveraging the TCN's parallel nature and one-dimensional convolutional operations. Furthermore, we train the TCN estimator on a mixed SNR dataset that incorporates various channel variations based on results from [9], rather than using a dataset generated solely at high SNR as in previous studies [2], [3], [6], [8], [10]. The DPA process solves the insufficient pilot problem, while the TCN uses the DPA estimates to accurately estimate the channel state. We evaluate our proposed DPA-TCN estimator on a realistic vehicle-to-vehicle same direction with wall (VTV-SDWW) tapped delay line (TDL) vehicular channel model specified in [11]. Performance assessment is based on BER, Normalised Mean Square Error (NMSE) and complexity metrics.

II. SYSTEM MODEL

This section provides an introduction to the IEEE 802.11p standard specification and discusses various channel estimation techniques following [12]. We will cover the widely used least squares (LS) method, which is considered a baseline in our work. Furthermore, we will dive into the DPA method, commonly utilised for initial channel estimation in various estimators, to highlight the importance of DL post-processing.

Lastly, we will compare the performance of these methods with the SOTA, LSTM-DPA-TA.

The IEEE 802.11p standard operates in the frequency band ranging from 5.0 GHz to 5.9 GHz, utilising a 10 MHz bandwidth [11]. It employs Orthogonal Frequency Division Multiplexing (OFDM) with 64 subcarriers for data transmission. Of these 64 subcarriers, 52 are used for data and pilots, while the remaining 12 at indices $\{-32, \dots, -27, 0, 27, \dots, 31\}$ serve as guard bands and DC offsets. Specifically, four pilot subcarriers are located at indices $\{-21, -7, 7, 21\}$, with the remaining 48 subcarriers designated for data transmission.

The IEEE 802.11p frame structure includes a preamble for signal detection, timing synchronisation, and channel estimation, followed by a signal field for transmission parameters. This work assumes perfect synchronisation and considers a frame structure with two extended preambles at the start, followed by I OFDM data symbols.

Given this structure, the signal received on the subcarrier $[k]$ at time i can be expressed as

$$Y_i[k] = H_i[k]X_i[k] + N_i[k], \quad (1)$$

where $X_i[k]$, $H_i[k]$ and $N_i[k]$ are the transmitted symbol, the channel frequency response and the Gaussian noise, respectively.

In this work, we investigate the VTV-SDWW vehicular channel model [11]. This model simulates communication between two vehicles moving in the same direction, separated by a central wall, at a distance of 300-400 metres. In this scenario, vehicles travel at 100 km/h with a Doppler shift of $f_d = 550$ Hz, using 16QAM modulation.

A. Least Squares (LS) Estimation

The LS estimation approach is a basic method commonly used in wireless communications. The LS approach uses two adjacent received symbols, as shown below:

$$\hat{H}_{LS}[k] = \frac{Y_1^{(p)}[k] + Y_2^{(p)}[k]}{2p[k]}, \quad (2)$$

where $p[k]$ is the known preamble, $Y_1^{(p)}[k]$ and $Y_2^{(p)}[k]$ are the preambles received in the $[k]$ subcarrier.

B. DPA

DPA addresses the pilot limitation issue in vehicular communications, without altering the IEEE 802.11p standard frame structure by introducing data pilots. As a result, DPA is recognised as the primary channel estimation method in vehicular communications [2]. First, DPA obtains an initial estimate of the frame using the LS estimate at the preambles (as shown in (2)).

Then, the equalised symbols $Y_i^{\text{eq}}[k]$ are obtained for the subsequent symbols by using the previous estimate to equalise the current symbol as follows:

$$Y_i^{\text{eq}}[k] = \frac{Y_i[k]}{\hat{H}_{i-1}^{\text{DPA}}[k]}, \hat{H}_0^{\text{DPA}}[k] = \hat{H}_{LS}[k]. \quad (3)$$

The equalised data symbol $y_i^{\text{eq}}[k]$ is remapped to the nearest constellation point $d_i[k]$ using the remapping operation $\mathfrak{R}(\cdot)$

$$d_i[k] = \mathfrak{R}\left(\frac{Y_i[k]}{\hat{H}_{i-1}^{\text{DPA}}[k]}\right), \hat{H}_0^{\text{DPA}}[k] = \hat{H}_{LS}[k], \quad (4)$$

The DPA channel estimate is then updated as follows:

$$\hat{H}_i^{\text{DPA}}[k] = \frac{Y_i[k]}{d_i[k]}. \quad (5)$$

C. LSTM-DPA-TA

The LSTM in this estimator uses the received OFDM symbol as input, performs the DPA process on the LSTM output, and then applies a time-average filter. The LSTM-DPA-TA method was developed in [8] and was evaluated on several channels, including the VTV-SDWW channel used in this work. We are using the LSTM-DPA-TA estimator for comparison purposes with our proposed DPA-TCN estimator.

III. PROPOSED DPA-TCN ESTIMATOR

This section outlines the architecture of the DPA-TCN estimator proposed in this work.

A. Temporal Convolutional Networks (TCNs)

TCNs are designed to handle sequential data by leveraging techniques from Convolutional Neural Networks (CNNs), as introduced by LeCun [13]. TCNs use causal and dilated convolutions along with residual connections to capture sequential dependencies. This is done to capture temporal dependencies within the input data using one-dimensional kernels.

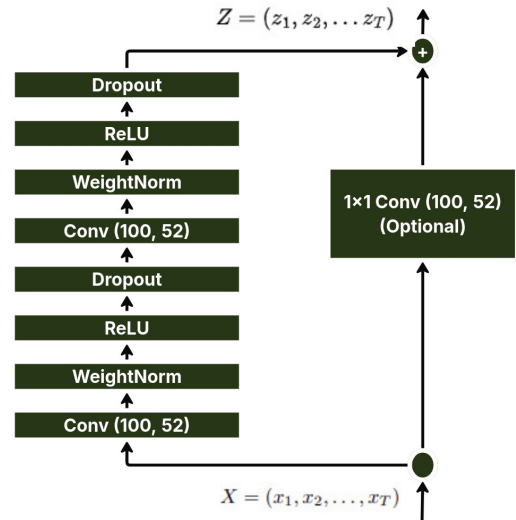


Fig. 1: TCN Residual block where 100 are the interleaved real and imaginary parts considered as features and 52 are complex active subcarriers which are considered as timesteps [14]

The goal is to predict the outputs z_0, \dots, z_T based on the input sequence x_0, \dots, x_T . To predict z_t at time t , only the inputs x_0, \dots, x_t are used. To keep the input and output dimensions consistent, TCNs apply padding of length (kernel

size - 1) at the start of the frame or the input sequence. Causal convolutions avoid future leakage by only convolving an output at time t with elements from time t or earlier. An optional residual connection allows the input to be added to the output if the output has dimensions different from the input [14].

Furthermore, the residual block typically includes dilated causal convolution layers, nonlinearity layers, and dropout layers after each convolutional layer to effectively prevent overfitting and enhance the model's generalisation capabilities.

B. Dataset Generation

The dataset for training the proposed DPA-TCN comprises 18,000 time specific frames with SNR levels ranging from 0 dB to 40 dB in 5 dB steps. We generate 2,000 frames at each SNR level, each containing transmitted and received versions of 50 OFDM symbols. Each OFDM symbol has complex values for 52 active subcarriers (48 data and 4 pilot subcarriers). The 16QAM symbols received are separated and interleaved per subcarrier and time slot, resulting in an input matrix of real values with dimensions 52×100 , and an output matrix (excluding pilot signals) of 48×100 . These samples are batched for processing.

The dataset for training the LSTM-DPA-TA and TCN-DPA estimator is generated as above at a fixed SNR of 40dB.

For all the estimator models, 12,000 frames are used for training, 4,000 frames for validation, and an independent set of 2,000 frames for testing.

Note that in this work, we treat subcarriers as time steps. The input of the TCN is a 3D tensor of shape (N, C, T) , where N is the batch size, C is the number of features (100 features) and T is the number of time steps (52 subcarriers). For a batch size of 128, the input tensor shape is $(128, 100, 52)$. A one-dimensional kernel size of 2 processes two consecutive time steps across the input sequence.

C. TCN estimator

This section describes the proposed DPA-TCN estimator where the input is a DPA-estimated frame $\hat{H}_i^{\text{DPA}}[k]$ which consists of \hat{H}_i^{D} (estimated data subcarriers) and \hat{H}_i^{P} (estimated pilot subcarriers) obtained following (2) to (5).

The proposed DPA-TCN estimator is shown in Fig. 2. The final channel estimate $\hat{H}_i^{\text{TCN}}[k]$ is updated based on (6):

$$\hat{H}_i^{\text{TCN}}[k] = f_{\text{TCN}}(\hat{H}_{i-1}^{\text{D}}, \hat{H}_{i-1}^{\text{P}} \cdot W), \quad (6)$$

Here, f_{TCN} is a TCN function that processes the input estimates \hat{H}_{i-1}^{D} and \hat{H}_{i-1}^{P} to produce the refined channel estimate $\hat{H}_i^{\text{TCN}}[k]$. The function f_{TCN} is parameterised by the weight matrix W , which includes the learnable filters and biases of the TCN's convolutional layers. These weights are optimised during training to minimise the estimation error.

In Fig. 2, the LS block initialises the channel estimation using the preamble, producing $\hat{H}_{\text{LS}}[k]$. The DPA block iteratively refines the estimation by utilising both the received signal $y_i[k]$ and the prior channel estimate $\hat{H}_i^{\text{DPA}}[k]$, resulting in $\hat{H}_i^{\text{DPA}}[k]$. The final TCN block further refines the estimate

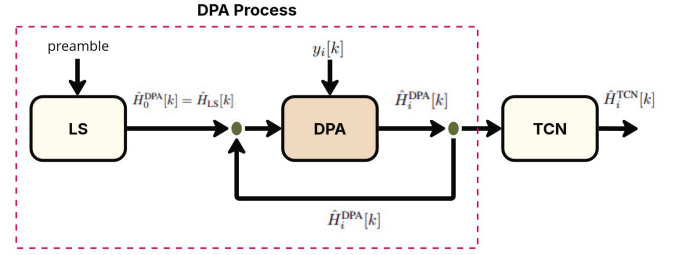


Fig. 2: DPA-TCN estimator process

to obtain the final channel estimate $\hat{H}_i^{\text{TCN}}[k]$. Arrows indicate the flow of data and the iterative feedback loop within the DPA block.

1) *Hyperparameter Tuning:* We use Bayesian optimisation (BO) to find the best hyperparameter combination for model performance. BO is an iterative algorithm that uses a probabilistic model, commonly a Gaussian process, to approximate an unknown objective function. For the TCN estimator, we optimise the learning rate, number of layers, the size of the kernel, the dropout, and the StepLR parameters (step size and gamma) using Weights & Biases (WandB)¹ and Optuna [15], a BO-backed hyperparameter tuning tool. The results of the experiment are recorded and tracked by WandB.

We conduct 50 trials, each running for 200 epochs, to comprehensively explore the hyperparameter space and observe the effects of hyperparameter changes. Convergence is determined using the validation loss. The hyperparameter search space and the best combination are shown in Table I.

TABLE I: Best TCN Hyperparameters

Hyperparameter	Search Space	Best Value
Learning Rate	1×10^{-5} to 1×10^{-2}	0.0006
Number of Layers	1 to 5	4
Kernel Size	2 to 5	2
Dropout	10^{-5} to 0.5	0.17
StepLR Step Size	10 to 50	21
StepLR Gamma	0.5 to 1	0.9
Epochs	0 to 200	156

IV. ANALYSIS AND RESULTS

In this section, the proposed DPA-TCN estimator is compared to other estimators namely the TCN-DPA, LSTM-DPA-TA, LS, DPA estimators as well as the ideal case with perfect channel knowledge. Furthermore, we present a comparative analysis of the computational complexity of these estimators.

Firstly, we perform a hyperparameter search for the LSTM model in the LSTM-DPA-TA estimator on the dataset generated at 40 dB using Optuna and WandB, following the same experimental setup discussed in Section III-C1. The following

¹<https://wandb.ai/>

best parameters are obtained for the LSTM model: a learning rate of 0.003, LSTM size = 128, StepLR step size = 12 and StepLR step gamma = 0.9. Then, this LSTM architecture is combined with DPA and TA as specified in [8] for comparison with the proposed DPA-TCN estimator.

A. BER

Fig. 3a shows the BER performance of various estimators. The LS estimator shows very poor performance across all tested SNRs. A standalone DPA estimator achieves better BER performance than LS but falls short when compared to NN-based estimators, especially at SNRs greater than 20 dB. The TCN-DPA trained at 40 dB achieves average performance at low SNRs up to 15 dB, and an exponential improvement from 20 dB to 40 dB. This improved performance in high SNR environments but not in low SNRs indicates the TCN's ability to excel in the conditions in which it is trained, but less so elsewhere.

Our proposed DPA-TCN estimator, trained using a mixed dataset, demonstrates good performance over a wide SNR range. Furthermore, our proposed DPA-TCN estimator achieves an overall BER performance comparable to that of the LSTM-DPA-TA. At SNR levels of 0 to 10 dB, the performance of the two estimators is similar. From 15 dB to 33 dB, the BER using the LSTM-DPA-TA estimator is lower than that when using the TCN estimator. Beyond 33 dB, the TCN estimator performs slightly better than the LSTM-DPA-TA.

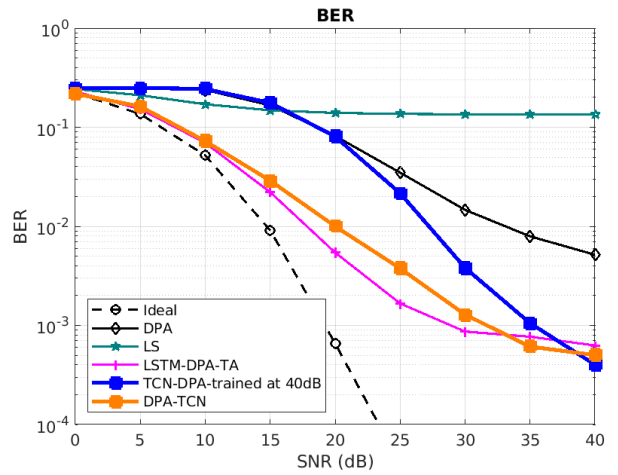
B. NMSE

The proposed DPA-TCN estimator aims to improve channel estimation by enhancing DPA processing. As illustrated in Fig. 3b, the DPA-TCN estimator consistently achieves lower NMSE values compared to the standalone DPA estimator and other methods across all SNR ranges investigated. In contrast, the TCN-DPA trained at 40 dB exhibits performance similar to the initial DPA estimator at low SNRs (0 to 15 dB) but shows exponential improvement starting from 20 dB. The LSTM-DPA-TA estimator outperforms all the models presented with significantly lower NMSE values which are only approached by the other presented models at high SNR levels. Our DPA-TCN outperforms the standalone DPA estimator over most of the SNR range, approaching LSTM-DPA-TA and TCN-DPA performance at very high SNR levels.

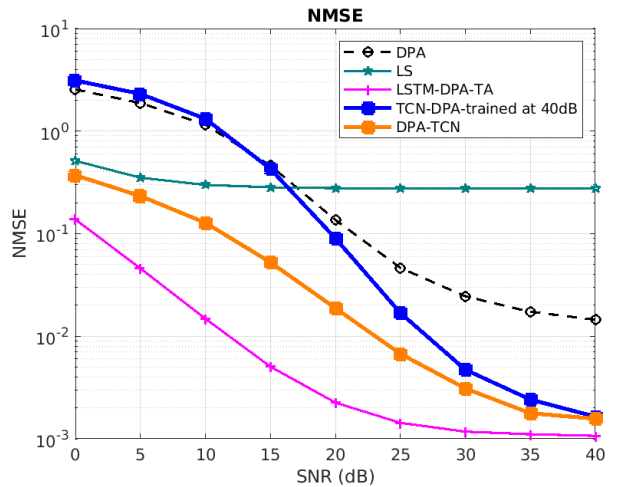
C. Complexity analysis

This section discusses the computational complexity of the estimators presented in this work using real-valued operations as a measure of complexity. Real-valued operations are generally used to analyse the complexity of channel estimation methods [2].

LS estimation requires the addition and division of the received preambles as shown in (2), which is $2X_A$ summations and $2X_A$ divisions where X_A represents the number of active subcarriers. In this case, $X_A = 52$ subcarriers. From equations (3) to (5) DPA requires 2 complex value divisions that require



(a) BER performance of various estimators.



(b) NMSE performance of various estimators.

Fig. 3: BER and NMSE simulation results on VTV-SDWW vehicular channel model.

6 real-valued multiplications, 2 real-valued divisions, 2 real-valued summations, and one real-valued subtraction totaling $26X_A$ operations.

In [8], the authors provide a detailed analysis of the computational complexity of the LSTM-DPA-TA estimator. This analysis includes the real-value operations performed by each LSTM gate and the TA post-processing method. The computational complexity for the LSTM-DPA-TA component is given in [8] as:

$$L_{\text{LSTM}} = 4(L^2) + L(4X_{\text{in}} + 3) + 18X_D + X_{\text{in}} + 13L + 5X_{\text{in}} + 8X_D - 8, \quad (7)$$

where the input size $X_{\text{in}} = 104$, is obtained by stacking the real and imaginary parts of the complex data along the subcarrier dimension, X_D is the input data subcarrier which is also obtained by stacking the real and imaginary parts of the 48 complex data subcarriers, and L represents the unit size of the LSTM, with $L = 128$.

The complexity of a TCN residual block depends primarily on its 1D convolutional layers. Our proposed DPA-TCN has a single residual block and performs a one-dimensional convolution along the subcarrier domain. The residual block has 4 layers and uses a kernel size $K = 2$. Our TCN has the same input (X_I) and output (X_O) sizes, which are equal to the number of active subcarriers $X_A = 52$. The input data contains 100 features, which are also fed into the TCN as X_F .

Each convolutional layer requires ($X_I \times X_F \times K$) real-valued multiplications and X_O real-valued summations. The total complexity for 4 layers is calculated as follows:

- Total multiplications:

$$N_{\text{mul}} = 4 \times (X_A \times K \times X_F) = 8(X_A X_F). \quad (8)$$

- Total summations:

$$N_{\text{sum}} = 4X_A. \quad (9)$$

Additionally, considering the residual connection, when the input and output dimensions are different, the multiplications for the residual connection equal $X_A \times 1$, and the summations for the residual connection are X_A . Thus, the total real-valued multiplications and summations of our TCN estimator are:

$$N_{\text{mul,sum}} = 8X_A X_F + 6X_A. \quad (10)$$

Equations 8 to 10 quantify the computational complexity of the TCN estimator in terms of real-valued multiplications and summations. The total complexity is dominated by the convolutional operations, which scale linearly with the number of active subcarriers X_A and the number of features X_F .

TABLE II: Total computational complexity

Estimator	Total complexity
LS	208
DPA	1,560
LSTM-DPA-TA	123,944
DPA-TCN (proposed)	43,472

Table II shows the total complexity of the estimators compared in this work. The proposed DPA-TCN estimator has a reduced complexity compared to the LSTM-DPA-TA estimator by approximately 65%. In other words, the TCN estimator is around 3 times less complex than the LSTM-DPA-TA. Its reduced complexity makes it ideal for vehicular communications that require faster processing.

V. CONCLUSION

In this paper, we proposed a novel DPA-TCN estimator for channel estimation in vehicular communications, specifically designed for the VTV-SDWW channel model. By integrating the advantages of DPA and a TCN, the proposed estimator effectively addresses the challenges of channel estimation in high-mobility environments. Simulation results indicate that our proposed DPA-TCN estimator achieves BER performance

comparable to the current state-of-the-art estimator (LSTM-DPA-TA) but with significantly reduced computational complexity in comparison. Future work will investigate other vehicular channel models.

ACKNOWLEDGMENT

The authors thank Telkom CoE at the NWU who supported this research.

REFERENCES

- [1] D. Jiang and L. Delgrossi, "IEEE 802.11p: Towards an International Standard for Wireless Access in Vehicular Environments," in *2008 VTC Spring - IEEE Vehicular Technology Conference*, 2008, pp. 2036–2040.
- [2] A. K. Gizzini, M. Chafii, A. Nimr, and G. Fettweis, "Deep Learning Based Channel Estimation Schemes for IEEE 802.11p Standard," *IEEE Access*, vol. 8, pp. 113 751–113 765, 2020.
- [3] J. Pan, H. Shan, R. Li, Y. Wu, W. Wu, and T. Q. Quek, "Channel Estimation Based on Deep Learning in Vehicle-to-Everything Environments," *IEEE Communications Letters*, vol. 25, no. 6, pp. 1891–1895, 2021.
- [4] P. Walk, H. Becker, and P. Jung, "OFDM Channel Estimation via Phase Retrieval," in *2015 49th Asilomar Conference on Signals, Systems and Computers*, 2015, pp. 1161–1168.
- [5] S. Han, Y. Oh, and C. Song, "A Deep Learning Based Channel Estimation Scheme for IEEE 802.11p Systems," in *2019-2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–6.
- [6] A. K. Gizzini, M. Chafii, A. Nimr, and G. Fettweis, "Joint TRFI and Deep Learning for Vehicular Channel Estimation," in *2020 IEEE Global Communications Conference (GLOBECOM) Workshops*. IEEE, 2020, pp. 1–6.
- [7] J. D. Jovane and C.-H. Lee, "Channel Estimation using Temporal Convolutional Networks for V2X Communications," in *ICC 2023 - IEEE International Conference on Communications*, 2023, pp. 565–570.
- [8] A. K. Gizzini, M. Chafii, S. Ehsanfar, and R. M. Shubair, "Temporal Averaging LSTM-based Channel Estimation Scheme for IEEE 802.11p Standard," in *2021 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2021, pp. 01–07.
- [9] S. A. Ngorima, A. S. J. Helberg, and M. H. Davel, "Neural Network-Based Vehicular Channel Estimation Performance: Effect of Noise in the Training Set," in *Artificial Intelligence Research*, A. Gerber, J. Maritz, and A. W. Pillay, Eds. Cham: Springer Nature Switzerland, 2025, pp. 192–206.
- [10] S. A. Ngorima, A. Helberg, and M. H. Davel, "Sequence Based Deep Neural Networks for Channel Estimation in Vehicular Communication Systems," in *Artificial Intelligence Research*, A. Pillay, E. Jembere, and A. J. Gerber, Eds. Cham: Springer Nature Switzerland, 2023, pp. 172–186.
- [11] G. Acosta-Marum and M. A. Ingram, "Six Time- and Frequency-Selective Empirical Channel Models for Vehicular Wireless LANs," in *2007 IEEE 66th Vehicular Technology Conference*, 2007, pp. 2134–2138.
- [12] S. A. Ngorima, A. Helberg, and M. H. Davel, "A Data Pilot-Aided Temporal Convolutional Network for Channel Estimation in IEEE 802.11p Vehicle-to-Vehicle Communications," in *Southern Africa Telecommunication Networks and Applications Conference (SATNAC)*, 2024.
- [13] Y. LeCun, Y. Bengio *et al.*, "Convolutional Networks for Images, Speech, and Time-Series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995.
- [14] S. Bai, J. Z. Kolter, and V. Koltun, "An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling," *arXiv preprint arXiv:1803.01271*, 2018.
- [15] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A Next-generation Hyperparameter Optimization Framework," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 2623–2631.