

Received November 19, 2020, accepted December 7, 2020, date of publication December 11, 2020, date of current version December 29, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3044097

Deep Learning-Based Wireless Channel Estimation for MIMO Uncoded Space-Time Labeling Diversity

BHEKISIZWE MTHETHWA^{ID}, (Member, IEEE), AND HONGJUN XU^{ID}, (Member, IEEE)

School of Engineering, University of KwaZulu-Natal, Howard College Campus, 4001 Durban, South Africa

Corresponding author: Bhekisizwe Mthethwa (tshomie2020@yahoo.com)

This work was supported by the UKZN CRART Centre.

ABSTRACT Uncoded space-time labeling diversity (USTLD) is a space-time block coded (STBC) system with labeling diversity applied to it to increase wireless link reliability without compromising the spectral efficiency. USTLD achieves higher link reliability relative to the traditional Alamouti STBC system. This work aims to design a bandwidth-efficient and blind wireless channel estimator for the USTLD system. Traditional channel estimation techniques like the least-squares (LS) and the minimum mean squared error (MMSE) methods are generally inefficient in using the channel bandwidth. The LS and MMSE channel estimation schemes require the prior knowledge of transmitted pilot symbols and/or channel statistics, together with the receiver noise variance, for channel estimation. A neural network machine learning (NN-ML) channel estimator with transmit power-sharing is proposed to facilitate blind channel estimation for the USTLD system and to minimize the required channel estimation bandwidth utilization. We mathematically model the equivalent noise power and derive the optimal transmit power fraction that minimizes the channel estimation bandwidth utilization. The blind NN-ML channel estimator with transmit power-sharing is shown to utilize 20% of the LS and MMSE wireless channel estimators' bandwidth to achieve the same bit error rate (BER) performance for the USTLD system in the case of 16-QAM and 16-PSK modulation.

INDEX TERMS Blind channel estimation, deep learning, imperfect channel estimation, space-time codes, space-time labeling diversity, transmit power-sharing, wireless MIMO.

I. INTRODUCTION

Uncoded Space-Time Labeling Diversity (USTLD) is a technique developed recently by [1] to increase the link reliability of space-time block coded (STBC) systems in a multiple-input multiple-output (MIMO) environment. It uses two distinct symbol constellation mapper designs to map bitstreams to symbols. The first STBC timeslot sends information symbols from a gray coded symbol constellation mapper. The second timeslot sends the same information symbols picked from the second constellation mapper designed using the labeling technique defined in [1]. This scheme outperforms the traditional Alamouti STBC [2] system in terms of bit error rate (BER) performance as it has coding gain over the Alamouti system. In [3], the authors develop a genetic algorithm-based mapper labeling design technique for non-symmetric constellations since the USTLD mapper design in [1] is limited to symmetric constellations. In [4], the authors develop a generic analytical framework to evaluate the BER performance of USTLD in Rician, and Rayleigh fading wireless channels for a three transmit antenna MIMO system. In [5], the authors show that applying media-based modulation with radio frequency (RF) mirrors enhances the wireless link reliability of USTLD STBC schemes. The authors in [6] apply signal space diversity (SSD) to USTLD STBC in order to improve the error rate performance of USTLD. They show that the SSD USTLD scheme outperforms the USTLD BER performance. Trellis code aided high rate space-time labeling diversity (TC-STLD) is proposed in [7] to deliver superior spectral efficiency whilst maintaining the BER performance, relative to that of USTLD. In [8], the authors investigate USTLD in a three transmit antenna MIMO configuration as the other USTLD research has only been carried out in a two transmit antenna MIMO configuration. They develop the second and third labeling mappers using a heuristic method and observe that the three transmit antenna USTLD scheme

The associate editor coordinating the review of this manuscript and approving it for publication was Jiankang Zhang^{ID}.

has superior BER performance relative to the two transmit USTLD scheme presented in [1]. Rectangular quadrature amplitude modulation (QAM) for USTLD is investigated in Nakagami- m fading channels in [9]. A heuristic algorithm to design the optimal labeling mapper for the rectangular QAM USTLD scheme is proposed. In [10], the authors develop a high-density M-QAM labeling mapper using a heuristic algorithm for a three transmit antenna USTLD STBC scheme. They design the second and third labeling mappers using the heuristic algorithm for 256-QAM and 1024-QAM constellations since most research has developed mapper designs for lower modulation orders.

The works discussed so far for the USTLD scheme have assumed a perfect wireless channel estimate, which motivates the investigation of USTLD under an imperfect channel estimate. Recently, deep learning has been proposed to address challenges associated with wireless channel estimators.

As stated earlier, deep learning has been proposed in the field of wireless channel estimation. In [11], the authors propose deep learning in predicting the time-frequency response of a fast-fading wireless MIMO channel. They show that the proposed deep learning algorithm has a competitive mean squared error performance relative to the traditional MMSE channel estimator. In [12], the authors prove that their proposed deep learning channel estimator outperforms the traditional compressed sensing-based algorithms for massive MIMO wireless channel estimation. The authors in [13] propose a deep learning channel estimation algorithm for doubly selective wireless fading channels. Deep learning is applied in [14] to estimate the uplink wireless channels for massive MIMO systems at the base station with some antennas using high-resolution analog-to-digital converters (ADC) and others using low-resolution ADCs. The proposed deep learning algorithm uses the high-resolution ADCs to predict the channels of the antennas using low-resolution ADCs. In [15], a deep learning-based channel estimation technique is proposed for wireless energy transfer. Based on the energy received by the energy receiver, the energy transmitter channel state information (CSI) is learned using the proposed deep learning autoencoder. The authors in [12] propose a learned denoising-based approximate message passing (LDAMP) channel estimator for beamspace millimeter-wave massive MIMO channels with limited RF chains. The deep learning-based LDAMP algorithm outperforms the compressed sensing-based algorithms. In [16], a deep learning-based channel estimator is proposed for a time-varying Rayleigh fading channel. Its mean squared error performance is shown to outperform that of the traditional channel estimation algorithms. A deep learning algorithm is proposed in [17] to handle the end-to-end wireless orthogonal frequency division multiplexing (OFDM) channels. It implicitly estimates the CSI and directly decodes the transmitted symbols. It shows robustness relative to the conventional channel estimation techniques when fewer training pilot symbols are used. In [18], deep learning-based channel estimation and equalization scheme (DL-CE) for filter bank multicarrier

(FBMC) modulation is proposed. It is shown in [18] that this DL-CE scheme achieves state-of-the-art performances in channel estimation and equalization. In [19], the authors propose a deep learning-based downlink channel estimator for fast time-varying and non-stationary wireless fading channels present in high-speed mobile scenarios. The proposed deep learning channel estimator proves to have better performance relative to the traditional channel estimators whilst offering lower computational complexity.

Research on bandwidth-efficient channel estimation has been performed in literature largely in MIMO-OFDM systems. In the works presented in [20]–[23], the authors develop bandwidth-efficient channel estimators for the MIMO-OFDM environment. In [24], the authors propose a bandwidth-efficient channel estimator for a single carrier MIMO system with frequency domain equalization. The channel estimator iteratively uses a series of fast-Fourier transforms (FFT) and inverse FFT operations to reconstruct the CSI fully. A bandwidth-efficient blind channel estimator is proposed in [25] for a full-duplex (FD) point-to-point wireless communication system. The blind channel estimator simultaneously estimates the channel parameters of the FD system without requiring time division duplex (TDD).

In summary, it is evident from the literature that LS [26] and MMSE [27] require prior knowledge of the transmitted pilot symbols and/or the wireless channel statistics to perform channel estimation. The other general observation is that the channel estimator's mean squared error (MSE) drops as the number of transmitted pilot symbols is increased. It will be challenging to perform channel estimation using the traditional LS and MMSE channel estimators in environments where the transmitted pilot symbols and channel wireless statistics are unknown. With the high cost of licensed wireless channel bandwidth, service providers are pressured to utilize bandwidth efficiently. Hence, large numbers of pilot symbols sent over a wireless channel for channel estimation may not be desirable. Therefore, we propose a blind NN-ML channel estimator with transmit power-sharing that minimizes the channel bandwidth usage whilst delivering a competitive MSE and BER performance compared to the traditional LS and MMSE schemes. We choose NN-ML because it does not require prior knowledge of the transmitted pilot symbols, wireless channel statistics, and receiver noise variance to perform channel estimation.

The idea of transmit power-sharing is taken from [28], where an optimal power fraction is derived that facilitates the optimal sharing of transmit power between the information symbols and a single reference symbol to improve BER performance. In our case, we apply this power-sharing technique between information symbols and multiple reference/pilot symbols to improve the MSE and BER performance of the NN-ML channel estimator relative to the traditional LS and MMSE based channel estimators. This translates to lower usage of channel bandwidth in order to deliver the same BER performance.

The main contributions of this paper are as follows:

- We propose a novel deep learning-based bandwidth-efficient blind channel estimator for the USTLD MIMO system by employing optimal transmit power-sharing between information symbols and pilot symbols. To our knowledge, no literature has developed a bandwidth-efficient channel estimator using transmit power-sharing between pilot symbols and information symbols. Over and above that, the literature in [20]–[25] develops bandwidth-efficient MIMO channel estimators for very different system models compared to ours. The differences in environmental context or system model affect the method of channel estimation bandwidth optimization. We, therefore, cannot, for example, directly use a MIMO-OFDM optimized channel estimator in our system model.
- We mathematically derive a multiple pilot/reference symbol equivalent noise power upper bound for USTLD MIMO, unlike in [28], where the equivalent noise power is only for a single reference symbol. No work in the literature has derived the equivalent noise power for the USTLD MIMO system.
- We apply differential calculus to determine the optimal power fraction that minimizes the equivalent noise power. The minimized equivalent noise power is shown to minimize the MSE and BER of the USTLD MIMO system. This minimization of the MSE and BER implies a minimization of channel estimation bandwidth utilization to achieve the same BER performance as the traditional channel estimation methods.

The remainder of the paper is organized as follows: In Section II, we present the system model for the proposed blind NN-ML channel estimator with transmit power-sharing for USTLD MIMO and the background theory of LS and MMSE channel estimation. In Section III, we introduce the proposed blind NN-ML channel estimator’s theory with transmit power-sharing. Section III also presents the equivalent noise power upper bound’s derivation and the optimal transmit power fraction. Section IV discusses the MSE and BER simulation results, and Section V concludes the paper.

Notation: Bold lowercase (**a**) and uppercase letters (**A**) denote vectors and matrices, respectively. $(\cdot)^H$ and $\|\cdot\|_F$ are the Hermitian and the Frobenius norm of a vector or matrix, respectively. $tr(\cdot)$ is a trace function which takes the sum total of the major diagonal of a matrix. The symbol $\forall x$ means for all values of x . The operator $E(\cdot)$ is the statistical expectation or mean of a random variable. The functions $Re(\cdot)$ and $Im(\cdot)$ return the real and imaginary components of a complex number, respectively.

II. SYSTEM MODEL AND CHANNEL ESTIMATION BACKGROUND

A. SYSTEM MODEL

A $2 \times N_r$ USTLD system is used to evaluate the channel estimation algorithms’ BER performance, where N_r is the number of receive antennas, and 2 represents the number of

transmit antennas. The USTLD system is a modification of the conventional $2 \times N_r$ Alamouti system [2]. The fundamental idea is to transmit a mapped symbol pair in the second time slot instead of the complex conjugates. The USTLD system generates the 2×2 STBC codeword matrix based on two mappers; ω_1^W and ω_2^W as in [1]. For example, the two mappers for 16-QAM signal constellations are the Gray-coded labeling map ω_1^{16} and the optimized labeling map ω_2^{16} as per [1]. The labeling maps and their design criterion are detailed in [1].

A bitstream consisting of $2 \log_2 W$ random bits, where W is the W-QAM/W-PSK modulation order, $\mathbf{d} = [\mathbf{b}_{t_1}, \mathbf{b}_{t_2}]$ is fed into the USTLD modulator to produce the modulated symbol pairs $\mathbf{x}_1 = [x_{t_1} x_{t_2}]^T$ and $\mathbf{x}_2 = [\tilde{x}_{t_2} \tilde{x}_{t_1}]^T$ for the first and second time slot, respectively. Let $(\mathbf{x}_1, \mathbf{x}_2)$ be an ordered pair, and let χ_{LD} be a set that contains all the possible modulated symbol pairs $(\mathbf{x}_1, \mathbf{x}_2)$, such that $(\mathbf{x}_1, \mathbf{x}_2) \in \chi_{LD}$. Based on \mathbf{b}_{t_1} and \mathbf{b}_{t_2} , where $\mathbf{b}_{t_1} = [b_1^1 b_2^1 \dots b_m^1]$, $\mathbf{b}_{t_2} = [b_1^2 b_2^2 \dots b_m^2]$ with $m = \log_2 W$, the labeling map ω_1^W produces x_{t_1} and x_{t_2} , while ω_2^W produces \tilde{x}_{t_1} and \tilde{x}_{t_2} , respectively, where x_{t_i} and \tilde{x}_{t_i} belong to an W-QAM/W-PSK constellation set χ , with $t_i \in [1 : W]$, $E \left\{ |x_{t_i}|^2 \right\} = E \left\{ |\tilde{x}_{t_i}|^2 \right\} = 1$ and $i \in [1 : 2]$.

The transmission of the 2×2 STBC symbols happens over a quasi-static fading wireless channel with a constant channel gain over one message frame, including $M = 200$ W-QAM/W-PSK information symbols and N channel estimation pilot symbols transmitted per frame per transmit antenna. The pilot symbols are generated using the Zadoff-Chu sequence [29] since it can generate orthogonal complex sequences of constant amplitude and varying phase. This is important to avoid creating a singular square matrix $\mathbf{X}_r \mathbf{X}_r^H$ since the LS, and MMSE channel estimation methods rely on matrix inversion. The Zadoff-Chu sequence pilot symbols are generated using Equation (1)

$$P(n) = \begin{cases} e^{-\frac{j\pi Qn(n+2q)}{N}}, & \forall N = 2\theta, \theta \in [1, \infty) \\ e^{\frac{-j\pi Qn(n+2q+1)}{N}}, & \forall N = 2\theta + 1, \theta \in [1, \infty) \end{cases} \quad (1)$$

where $n \in [0 : N - 1]$, $P(n) \in \mathbb{C}$ is the complex pilot symbol at position n of the N -dimensional pilot symbol vector, N is the number of pilot symbols transmitted per pilot symbol vector, j is a complex number, $q \in \mathbb{N}$ and Q is a relative prime number to N and obeys the equation

$$\varphi(N, Q) = 1 \quad (2)$$

where φ is the greatest common divisor function.

The wireless channel is Rayleigh frequency-flat fading. The received pilot symbols and information/message symbols at the receiver are mathematically modeled as per Equations (3) and (4):

$$\mathbf{Y}_r = \mathbf{H}^T \mathbf{X}_r + \mathbf{N}_r \quad (3)$$

$$\mathbf{Y}_i = \mathbf{H}^{\tau+1} \mathbf{X}_i + \mathbf{N}_i \quad (4)$$

where $\mathbf{H}^{\tau+1} = \mathbf{H}^\tau = \mathbf{H} \in \mathbb{C}^{N_r \times N_t}$ is the constant wireless channel matrix over one transmission frame because the

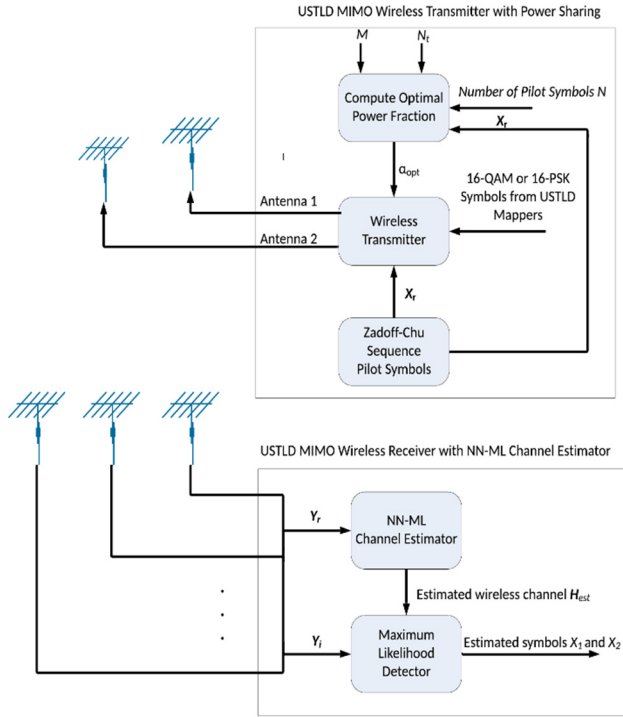


FIGURE 1. Shows the USTLD system with blind NN-ML channel estimator with transmit power-sharing.

wireless channel is quasi-static fading. The channel matrix H has complex channel gains, which are independent and identically distributed (i.i.d) according to $\mathcal{CN}(0, 1)$. $X_r \in \mathbb{C}^{N_t \times N}$ is the transmitted pilot symbol matrix, $X_i \in \mathbb{C}^{N_t \times 2}$ is the transmitted information symbol matrix over $N_t = 2$, transmit antennas, and two timeslots. The information symbol matrix X_i has transmitted W-QAM/W-PSK symbols. $Y_r \in \mathbb{C}^{N_r \times N}$ is the received/observed pilot symbol matrix over N_r receive antennas, $Y_i \in \mathbb{C}^{N_r \times 2}$ is the received information symbol matrix over N_r receive antennas and two timeslots for USTLD. The additive white Gaussian noise (AWGN) matrix $N_r \in \mathbb{C}^{N_r \times N}$ is observed at the wireless receiver over the received $N_r \times N$ pilot symbols. The AWGN matrix $N_i \in \mathbb{C}^{N_r \times 2}$ is observed at the wireless receiver when receiving the information symbols over two timeslots. The reference/pilot noise matrix N_r and the information noise matrix N_i have i.i.d entries that follow the complex Gaussian distribution as follows:

$$n_i^{zx} \sim \mathcal{CN}(0, \sigma_i^2) \quad \text{and} \quad n_r^{wy} \sim \mathcal{CN}(0, \sigma_r^2) \quad (5)$$

where n_i^{zx} is the information noise matrix z^{th} row and x^{th} column entry, σ_i^2 is the average noise power for the information receiver white noise, n_r^{wy} is the reference/pilot noise matrix w^{th} row and y^{th} column entry with an average noise power of σ_r^2 .

As shown in Fig. 1, the proposed system takes a fraction of the transmit power from the information symbol transmission from the wireless transmitter side and donates this

transmit power fraction to the reference/pilot symbol transmission [28]. Knowing that we have M information symbols transmitted per frame per transmit antenna means we donate $M\alpha$ transmit power to N pilot symbols. This implies that each pilot symbol gets $\frac{M}{N}\alpha$ extra transmit power, and each information symbol loses α transmit power. Mathematically this is denoted as follows:

$$\sigma_i^2 = \frac{2}{(1 - \alpha)\gamma} \quad (6.1)$$

$$\sigma_r^2 = \frac{2}{(1 + \frac{M}{N}\alpha)\gamma} \quad (6.2)$$

where α is the transmit power fraction, M is the number of information symbols transmitted per frame per transmit antenna, N is the number of pilot symbols sent per frame per transmit antenna, and γ is the average received signal-to-noise ratio (SNR) per receive antenna. The total power for the transmission of $M + N$ information and pilot symbols must be constant. Equations (6.1) and (6.2) obey this conservation of transmit power constraint. The optimal power fraction that ensures the optimal BER and MSE performance is computed using the $\alpha_{opt} = f(X_r, N, M, N_t)$ function that needs to be derived.

On the wireless receiver side, the NN-ML channel estimator is fed the received pilot symbol matrix Y_r . The NN-ML channel estimator then predicts the wireless channel and feeds it into the maximum likelihood (ML) detector and this channel estimate is done once per received frame. The ML detector then uses the channel estimate \hat{H}_{est} to detect the transmitted symbols based on the received Y_i symbol matrix.

B. BACKGROUND OF TRADITIONAL CHANNEL ESTIMATION METHODS

The LS [26] channel estimation method is the least complex channel estimation method relative to the MMSE [27] method and the approximate linear minimum mean square error (ALMMSE) [30] method but is generally the least performing of the channel estimation methods. The LS method works by generating a closed-form channel estimation formula, which estimates a wireless channel that minimizes the square of the Euclidean distance between the observed/received pilot symbol matrix and the product of the estimated wireless channel and the transmitted pilot symbol matrix. The formula for estimating the wireless channel based on the observed/received pilot vectors and the transmitted known pilot symbol matrix is as shown [26]

$$\hat{H}_{LS} = Y_r X_r^H (X_r X_r^H)^{-1} \quad (7)$$

where \hat{H}_{LS} is the LS estimated wireless channel matrix, Y_r is the observed/received pilot symbol matrix, and X_r is the transmitted pilot symbol matrix. From Equation (7) it is clear that $X_r X_r^H$ must be invertible and hence non-singular in nature, which motivates the selection of orthogonal pilot symbol vectors as entries in the transmitted pilot symbol matrix. Equation (7) also shows that the LS channel estimator

requires the full knowledge of the transmitted pilot symbol matrix. The MMSE channel estimation method works by estimating the wireless channel using Equation (8) [27]

$$\hat{\mathbf{H}}_{mmse} = \left(\sigma_i^2 \mathbf{R}_{HH}^{-1} + \mathbf{X}_{mmse}^H \mathbf{X}_{mmse} \right)^{-1} \mathbf{X}_{mmse}^H \mathbf{Y}_{mmse} \quad (8)$$

where σ_i^2 is the receiver noise variance, $\mathbf{R}_{HH} = E(\mathbf{H}_{mmse} \mathbf{H}_{mmse}^H)$ is the wireless channel autocorrelation matrix, $\mathbf{X}_{mmse} = \mathbf{X}_r^H$ is the MMSE pilot symbol matrix, $\mathbf{Y}_{mmse} = \mathbf{X}_{mmse} \mathbf{H}_{mmse} + \mathbf{N}_{mmse}$ is the observed/received MMSE pilot symbol matrix where $\mathbf{H}_{mmse} = \mathbf{H}^H$. As can be seen from Equation (8), the MMSE channel estimator requires the full knowledge of the pilot symbol matrix, wireless channel autocorrelation statistics, and the noise variance at the receiver side. These are assumed to be known without any estimation errors.

III. PROPOSED CHANNEL ESTIMATION METHOD FOR MIMO USTLD

The channel estimation method proposed here facilitates blind channel estimation when the transmitted pilot symbols, wireless channel second-order statistics, and the noise variance are unknown at the receiver side. The NN-ML channel estimator with transmit power-sharing method is a blind machine learning channel estimator. It also reduces the required bandwidth to achieve a good MSE and BER performance relative to the traditional channel estimation methods.

Recently, researchers within the communications research space have taken a keen interest in applying machine learning to solve communications-related research problems. The problems are primarily related to wireless symbol detectors' design using machine learning. The order of computational complexity of current expert wireless receivers is high for higher-order modulation W-QAM/W-PSK. Machine learning comes with the benefit of training a mathematical function to predict an output based on a noisy input. Once trained, there is no need for the machine learning algorithm to search iteratively, in a large search space for higher-order modulation orders, for an estimated transmitted symbol in the case of a wireless receiver symbol detector. The function will, in a much shorter convergence time, with similar BER performance, estimate the transmitted symbol compared to a ML detector that takes a longer time to converge to a solution. This is critical for real-time communication environments as link latency needs to be minimal to achieve a good quality of service (QoS).

In the case of channel estimation, we evaluate if we can train a machine-learning algorithm to predict the wireless channel based on a noisy received pilot symbol matrix with similar or better MSE and BER performance relative to the expert method of using LS or MMSE to carry out channel estimation. The advantage of using machine learning over LS or MMSE is that we can blindly estimate the wireless channel without knowing the pilot symbol matrix, channel autocorrelation matrix, and receiver noise variance.

The work in this section is organized as follows: Section III-A concentrates on Hyperparameter tuning of the neural network model, Section III-B ventilates the training phase of the neural network model and Section III-C exhibits the derivation of the optimal transmit power fraction that minimizes the channel estimation bandwidth utilization.

A. PROPOSED NN-ML CHANNEL ESTIMATION HYPERPARAMETER TUNING

For a supervised machine learning algorithm to be useful, it needs to be first trained using appropriate data. Two machine learning algorithms are being trained for, the first being the NN-ML-Channel-Estimation without power-sharing where $\alpha = 0$. The second is the NN-ML-Channel-Estimation with power-sharing, where $\alpha = \alpha_{opt}$. We then generated 10 000 samples for the received pilot symbols based on Equation (3) for $\alpha = 0$ and $\alpha = \alpha_{opt}$. We also used the Zadoff-Chu sequences in Equation (1) to generate the transmitted pilot symbol matrix over a range of N pilot symbols. The 10 000 samples were generated for each SNR value in the range [0dB to 16dB, step 2dB]. Since this section is responsible for hyperparameter tuning, we only tuned the hyperparameters using the 10dB SNR samples in order to reduce the tuning time. Table 1 shows the machine learning architecture and the training hyperparameters found using a genetic algorithm for the cases when $\alpha = 0$ and $\alpha = \alpha_{opt}$. The hyperparameters found through genetic algorithm [31] optimization are the pseudo-random number generator seed value, learning rate and training batch size. The objective function which the genetic algorithm optimized was the validation MSE at 10dB SNR.

The NN-ML-Channel-Estimation architecture in Table 1 was invariant to the changes in the number of pilot symbols N . However, the architecture was found to be sensitive to the MIMO receive antenna configuration N_r . This is because changing the value of N_r explicitly alters the MIMO channel matrix dimensions, which also alters the number of required neurons at the output layer of the architecture. Therefore, we declare that the architecture in Table 1 is only valid for the 2×4 MIMO configuration discussed in our simulation results. Any other MIMO configuration may require a new architecture and re-training of the NN-ML-Channel-Estimation model.

The input data dimension, of the architecture in Table 1, is based on the number of transmitted pilot symbols (N) per symbol vector and the number of receive antennas N_r . At the bottom of Table 1 are the tuneable hyperparameter values, which we derived via a genetic algorithm.

Algorithm 1 works by generating P random population members with randomized seed-values, batch sizes and learning rates as an initial Genetic algorithm population of possible hyperparameters. The population members, Ω_k , hyperparameters are randomly assigned values subject to the constraints stated in Step 1 of Algorithm 1. The algorithm then iterates through each population member per generation. It uses each population member's hyperparameters to train the neural

TABLE 1. NN-ML-channel-estimation architecture and hyper-parameters.

Layer	Parameter Description
Flatten	$1 \times 2N_r \times 2N$ input data dimension
(Input Layer)	256 neurons. LeakyReLU
Dense+LeakyReLU	slope of 0.3
BatchNormalization	void
Dropout	Probability of 0.5
(Hidden Layer)	256 neurons. LeakyReLU
Dense+LeakyReLU	slope of 0.3
BatchNormalization	void
Dropout	Probability of 0.5
(Output Layer)	16 neurons.
Dense+Linear	
Batch Size=2189, Learning Rate=0.00104, Seed Value=433	
The above hyperparameters are for when $\alpha = \alpha_{opt}$	
Batch Size=1300, Learning Rate= 0.00583, Seed Value=549	
The above hyperparameters are for when $\alpha = 0$	

network architecture in Table 1, represented as $\Phi(\cdot)$ in the Algorithm. The neural network architecture is trained with a training dataset of 80% of the imported dataset stored in χ . The other 20% is used as a test dataset to produce the validation MSE at 10dB SNR. The dataset stored in χ is produced from Equation (3), at 10dB SNR, for the power fraction values $\alpha = 0$ and $\alpha = \alpha_{opt}$. The training dataset is collected at a single SNR value of 10dB because at this stage, we are merely pre-training the neural network to select hyperparameters that produce the lowest validation MSE at 10dB SNR. At this stage, we are not fully training the neural network for channel estimation but for tuning or selecting optimal hyperparameters.

The population members in the current generation are selected one after the other, and their hyperparameters, together with the training and test datasets, are used to train and test the neural network in Table 1 in order to evaluate the validation MSE at 10dB SNR. This validation MSE is saved in a list that stores the mapping between the population member's number k and the validation MSE associated with the population member's hyperparameters. When all the population members have been used to determine the validation MSE, at 10dB SNR, for the neural network in Table 1, the algorithm uses the stored validation MSEs to create the next generation of population members. It does this by computing the fitness value associated with each population member. The fitness value is calculated based on the validation MSE of the neural network in Table 1 and

Algorithm 1 Genetic Algorithm

Result: Gene sequence with pseudo-random seed-value, batch size and learning rate for the population member that produces the lowest validation MSE at 10dB SNR.

Initialization:

$N_r = \epsilon$,
 $N = \delta$,
 $\mu_l = 0.35$
 $\mu_g = 0.15$
 $P = 10$
 $\omega[1:P] = 0$
 $G = 100$
 $\pi = 0.20$
 $E = 1000$

Step 1: Generate P random population members with random hyperparameter values.

Let each population member be denoted by Ω_k where $k \in [1:P]$. Ω_k contains the set $\{\lambda_k, \eta_k, \zeta_k\}$ where λ_k is the pseudo-random seed value, η_k is the learning rate and ζ_k is the batch size for the k^{th} population member. The random values are set within the following constraints: $\lambda_k \in [2, 1001] \in \mathbb{Z}$, $\eta_k \in [0.0001, 0.01] \in \mathbb{R}$, and $\zeta_k \in [1000, 4000] \in \mathbb{Z}$.

Step 2: Set $i = 1$

Step 3: IF $i \leq G$ THEN

- a) Set $k = 1$
- b) IF $k \leq P$ THEN
 - I. $parent_k = \Omega_k$
 - II. $\lambda_k = parent_k.getSeedValue()$
 - III. $\eta_k = parent_k.getLearnRate()$
 - IV. $\zeta_k = parent_k.getBatchSize()$
 - V. $\chi = Import(Dataset)$
 - VI. $\omega[k] = \Phi(\chi, \pi, E, \lambda_k, \eta_k, \zeta_k, N_r, N)$
 - VII. $k = k + 1$
 - VIII. GOTO Step 3b
- c) ELSE
 - I. Set $seq = 1$
 - II. IF $seq \leq P$ THEN
 - $\tau[1:P] = getFitnessValues(\omega[1:P])$
 - $\rho[1:P] = getSelectionProbabilities(\tau[1:P])$
 - $v[1:2] = selectParents(\rho[1:P])$
 - $c = produceChild(v[1:2])$
 - $\Omega_{seq} = c$
 - $seq = seq + 1$
 - GOTO Step 3c number II.
 - III. $i = i + 1$
 - IV. GOTO Step 3

Step 4: En

the formula $\tau[k] = \frac{10^6}{10^{(1000\omega[k]-45)}}$. $\omega[k]$ is the k^{th} population member's validation MSE at 10dB SNR. $\tau[k]$ is the k^{th} population member's fitness value. The formula for computing the fitness values is found empirically by maximizing the

TABLE 2. Definition of genetic algorithm parameters.

Parameter	Description
N_r	number of receive antennas in MIMO configuration
N	Number of pilot symbols
μ_l	Local mutation rate
μ_g	Global mutation rate
P	Population size per generation
$\omega[1:P]$	A list that stores the MSE validation loss values at 10dB SNR for each Ω_k .
G	Maximum number of evolutionary generations
π	The fraction of the dataset which will be used as the test data.
E	Number of training Epochs
$\tau[1:P]$	A list that stores the fitness value for each Ω_k .
$\rho[1:P]$	A list that stores the selection probabilities for each Ω_k .
$v[1:2]$	A list that stores the two parents that are selected for mating
c	The child produced from the mating of the two selected parents.
χ	Dataset

small differences between validation MSE values at 10dB SNR. If we do not non-linearly amplify the small differences between MSE values, then the selection probabilities will be almost identical for all population members since their fitness values will be very close to each other. For example, if 5 population members A, B, C, D, and F have validation MSE values of 0.04797, 0.0478, 0.048, 0.04632, and 0.04921, respectively, then it is obvious to see that the selection probabilities for population members A, B, C, D, and F will be almost identical if the fitness function does not non-linearly amplify the validation MSE values. However, if we use our empirical formula to calculate the fitness values, we get the following fitness values for population members A, B, C, D, and F: 1071.51, 1584.89, 1000, 47863.01, and 61.66, respectively. The population member F has the worst validation MSE and the lowest fitness value of 61.66. The population member D has the best validation MSE and the highest fitness value of 47863.01. The difference between the validation MSE for population members F and D is only 0.0029. However, the fitness values have a huge difference to avoid equal selection probabilities being calculated in scenarios when the validation MSE values are very close to each other.

After the fitness values are computed, the selection probability per population member is calculated based on each population member's fitness value. The selection probabilities

are computed based on the formula $\rho[k] = \frac{\tau[k]}{\sum_{l=1}^P \tau[l]}$ [31].

$\rho[k]$ is the selection probability of the k^{th} population member. The selection probability is the likelihood of a population member being randomly selected to mate and produce a child for future generations. Only two parents are randomly selected to produce one child.

When the two parents are selected randomly, a genetic crossover is performed to produce a child. The crossover is done by inheriting the pseudo-random seed value from the first parent and then inheriting the batch size and the learning rate from the second parent. Random genetic mutation may occur with a probability of 0.35 for local mutation and 0.15 for global mutation. Local mutation involves altering a single inherited hyperparameter value of the child. Global mutation involves the altering of all three inherited hyperparameter values of the child. The genetic mutation involves randomly assigning values to the hyperparameters subject to the constraints stated in Step 1 of Algorithm 1.

The child produced from the selected parents' mating is then added as a new population member for the next generation. Selecting mating parents and producing children is repeated until the new population size is P . Only then do we destroy the old population and move on to the next generation of population members.

The process of searching for a global minimum validation MSE is repeated until the number of iterations is equal to the maximum number of generations G . Only then do we search for the population member Ω_k , together with its hyperparameters, that produces the lowest validation MSE evaluated from generation 1 to generation G . The optimal population member's hyperparameters are then used to fully train the neural network in Table 1 from 16dB to 0dB SNR.

B. TRAINING THE NN-ML CHANNEL ESTIMATOR

There are two sets of data used in the machine learning algorithm training and testing in Table 1. The first dataset \tilde{A} , is the training dataset, which is made up of 80% of the 10 000 samples generated using the received pilot symbol matrix data from Equation (3), and the second dataset M is the test dataset, which is the remaining unseen 20% of the 10 000 samples. The datasets are collected for SNR values from 0dB to 16dB. The datasets are generated for both scenarios when $\alpha = 0$ and $\alpha = \alpha_{opt}$. We trained the machine learning algorithm over a wide range of SNR values 0dB to 16dB. The algorithm was trained from 16dB down to 0dB as it was noticed that training it the other way from 0dB to 16dB yielded a poor validation mean squared error per SNR training cycle.

For the training to happen, the received complex pilot symbol matrix from Equation (3) is converted into a 2-dimensional data structure, with real numbers, with a single channel. A matrix representation that converts a complex matrix into a 2-dimensional real-valued data structure is used as per [32]. The resulting 2-dimensional data structure is stored in a single element array to create a single channel.

Since the received pilot symbol matrix is $\mathbb{C}^{N_r \times N}$, the real-valued matrix is $\mathbb{R}^{2N_r \times 2N}$ as shown in Equation (9) [32]

$$\mathbf{Z}_r = \begin{pmatrix} \text{Re}(\mathbf{Y}_r) & -\text{Im}(\mathbf{Y}_r) \\ \text{Im}(\mathbf{Y}_r) & \text{Re}(\mathbf{Y}_r) \end{pmatrix} \in \mathbb{R}^{2N_r \times 2N} \quad (9)$$

where in Equation (9):

$$\mathbf{Y}_r = \begin{pmatrix} y_{11} & \cdots & y_{1N} \\ \vdots & \ddots & \vdots \\ y_{N_r,1} & \cdots & y_{N_r,N} \end{pmatrix} \in \mathbb{C}^{N_r \times N}.$$

The set of 10 000 samples is made up of the 2-dimensional data \mathbf{Z}_r with a single channel. To train the machine learning algorithm, using supervised learning, we need the output label data that corresponds to this input training set $\hat{\mathbf{A}}$. The output training data is the actual wireless channel matrix $\mathbf{H} \in \mathbb{C}^{N_r \times N_r}$ as per the method used in [11]. We then convert this wireless channel matrix into a real-valued vector $\mathbf{h} \in \mathbb{R}^{1 \times 2N_r N_r}$, which in our case we have $N_r = 4$ and $N_r = 2$, which means $\mathbf{h} \in \mathbb{R}^{1 \times 16}$, which makes the output layer of the neural network a 16 neuron layer as per Table 1. The real-valued entries of the vector \mathbf{h} are determined by taking the real and imaginary values of the complex entries of the channel matrix \mathbf{H} .

During training, the set $\hat{\mathbf{A}}$ is fed into the machine learning algorithm function $F(\hat{\mathbf{A}}, K)$ with hyper-parameters (K). This function's output is compared to the output labeled data in vector \mathbf{h} that corresponds to the actual wireless channel as per [11]. The function $F(\hat{\mathbf{A}}, K)$ hyper-parameters (K) are tuned using the Adam optimizer with an objective to minimize the validation MSE based on the test data from set \mathcal{M} . The optimization objective function or loss function is shown in Equation (10)

$$\text{MSE Loss} = \frac{1}{\|\hat{\mathbf{A}}\|} \sum_{\mathbf{x} \in \hat{\mathbf{A}}} \|F(\mathbf{x}, K) - \mathbf{h}\|_F^2 \quad (10)$$

where $\|\hat{\mathbf{A}}\|$ is the batch size of the training set. This loss function is used in back-propagation by the Adam optimizer [33] to determine the neural network function $F(\hat{\mathbf{A}}, K)$ hyper-parameters (K), or synaptic weights. During training, the training process looks for the weights that produce the lowest validation MSE, which is determined by evaluating the MSE loss function in Equation (10) after each training epoch using the unseen test data set \mathcal{M} .

C. OPTIMAL TRANSMIT POWER SHARING

Based on Section II's system model, we derive the optimal power fraction function $\alpha_{opt} = f(\mathbf{X}_r, N, M, N_r)$. This subsection is dedicated to deriving the optimal power fraction function and finding the optimal number of pilot symbols to be transmitted for optimal MSE and BER performance.

Inspired by a generalized differential scheme for spatial modulation systems [28], the following derivation is performed to determine the optimal power fraction that minimizes the NN-ML wireless channel estimator's MSE and BER performance. Based on Equations (3) and (4), we can

see that the channel matrix \mathbf{H} is the common link between Equations (3) and (4). Manipulating Equation (3), we get the following Equation (11):

$$(\mathbf{Y}_r - \mathbf{N}_r) = \mathbf{H}\mathbf{X}_r, \text{ further manipulation yields}$$

$(\mathbf{Y}_r - \mathbf{N}_r)\mathbf{X}_r^H = \mathbf{H}\mathbf{X}_r\mathbf{X}_r^H$, where $\mathbf{X}_r\mathbf{X}_r^H$ is an invertible square matrix, thus

$$\mathbf{H} = (\mathbf{Y}_r - \mathbf{N}_r)\mathbf{X}_r^H (\mathbf{X}_r\mathbf{X}_r^H)^{-1} \quad (11)$$

Substitute Equation (11) into Equation (4) we get the following:

$\mathbf{Y}_i = ((\mathbf{Y}_r - \mathbf{N}_r)\mathbf{X}_r^H (\mathbf{X}_r\mathbf{X}_r^H)^{-1})\mathbf{X}_i + \mathbf{N}_i$, and simplifying yields Equation (12)

$$\mathbf{Y}_i = \mathbf{Y}_r\mathbf{X}_r^H (\mathbf{X}_r\mathbf{X}_r^H)^{-1} \mathbf{X}_i - \mathbf{N}_r\mathbf{X}_r^H (\mathbf{X}_r\mathbf{X}_r^H)^{-1} \mathbf{X}_i + \mathbf{N}_i \quad (12)$$

We can see from Equation (12) that the first coefficient of \mathbf{X}_i is $\mathbf{Y}_r\mathbf{X}_r^H (\mathbf{X}_r\mathbf{X}_r^H)^{-1}$ and is identical to Equation (7), which is the least-squares wireless channel estimate. We can then replace the first coefficient of \mathbf{X}_i with the generic placeholder for the wireless channel estimate, which we will call $\hat{\mathbf{H}}_{est}$. This changes Equation (12) to be represented mathematically as $\mathbf{Y}_i = \hat{\mathbf{H}}_{est}\mathbf{X}_i - \mathbf{N}_r\mathbf{X}_r^H (\mathbf{X}_r\mathbf{X}_r^H)^{-1} \mathbf{X}_i + \mathbf{N}_i$.

To find the optimal power fraction that minimizes the channel estimate MSE and BER, we need to derive the equivalent noise power based on a similar method used to derive the equivalent noise power for a generalized differential scheme for spatial modulation systems [28]. From Equation (12) it is clear that the equivalent noise power is actually dependent on the noise term $-\mathbf{N}_r\mathbf{X}_r^H (\mathbf{X}_r\mathbf{X}_r^H)^{-1} \mathbf{X}_i + \mathbf{N}_i$, for the MIMO pilot assisted wireless channel estimation methods. If $\hat{\mathbf{H}}_{est}$ is the estimated wireless channel at an instant, then the average signal mean squared error expression, $\text{argmin}_{\hat{\mathbf{H}}_{est}} E(\|\mathbf{Y}_i - \hat{\mathbf{H}}_{est}\mathbf{X}_i\|_F^2)$, needs to be minimized for a good channel estimation accuracy where the operator $E(\cdot)$ is the statistical expectation given that \mathbf{X}_i is known at the receiver. In order to practically evaluate the accuracy or MSE of the wireless channel estimate, we need to transmit a fixed, known \mathbf{X}_i information symbol matrix based on the USTLD method and observe the received \mathbf{Y}_i symbol matrix and evaluate $E(\|\mathbf{Y}_i - \hat{\mathbf{H}}_{est}\mathbf{X}_i\|_F^2)$ after having have estimated the wireless channel and obtained $\hat{\mathbf{H}}_{est}$. Expanding the minimization of the term $E(\|\mathbf{Y}_i - \hat{\mathbf{H}}_{est}\mathbf{X}_i\|_F^2)$, we get the following mathematical expression: $\text{argmin}_{\hat{\mathbf{H}}_{est}} E(\|\mathbf{Y}_i - \hat{\mathbf{H}}_{est}\mathbf{X}_i\|_F^2) = \text{argmin}_{\hat{\mathbf{H}}_{est}} E(\|(\mathbf{H} - \hat{\mathbf{H}}_{est})\mathbf{X}_i + \mathbf{N}_i\|_F^2)$.

Using the Frobenius norm property in Equation (13.1) and the Cauchy–Bunyakovsky–Schwarz inequality in Equation (13.2) [34]

$$\|\mathbf{AB}\|_F \leq \|\mathbf{A}\|_F \|\mathbf{B}\|_F \quad (13.1)$$

$$\left| \text{tr}(\mathbf{AB}^H) \right| \leq \|\mathbf{A}\|_F \|\mathbf{B}\|_F \quad (13.2)$$

And using the Frobenius norm triangle inequality in [34], we get

$\operatorname{argmin}_{\widehat{\mathbf{H}}_{est}} E \left(\left\| (\mathbf{H} - \widehat{\mathbf{H}}_{est}) \mathbf{X}_i + \mathbf{N}_i \right\|_F^2 \right) \leq \operatorname{argmin}_{\widehat{\mathbf{H}}_{est}} E \left(\left\| \mathbf{H} - \widehat{\mathbf{H}}_{est} \right\|_F^2 \left\| \mathbf{X}_i \right\|_F^2 + \left\| \mathbf{N}_i \right\|_F^2 + 2 \left\| \mathbf{H} - \widehat{\mathbf{H}}_{est} \right\|_F \left\| \mathbf{X}_i \right\|_F \left\| \mathbf{N}_i \right\|_F \right)$. But $E \left(\left\| \mathbf{X}_i \right\|_F \right) = \lambda$, $E \left(\left\| \mathbf{N}_i \right\|_F \right) = \mu$, $E \left(\left\| \mathbf{X}_i \right\|_F^2 \right) = 2N_t$ and $E \left(\left\| \mathbf{N}_i \right\|_F^2 \right) = 2N_r \sigma_i^2$ therefore we can simplify the expression to form Equation (14)

$$\begin{aligned} & \operatorname{argmin}_{\widehat{\mathbf{H}}_{est}} E \left(\left\| (\mathbf{H} - \widehat{\mathbf{H}}_{est}) \mathbf{X}_i + \mathbf{N}_i \right\|_F^2 \right) \\ & \leq \operatorname{argmin}_{\widehat{\mathbf{H}}_{est}} \left(2E \left(\left\| \mathbf{H} - \widehat{\mathbf{H}}_{est} \right\|_F^2 \right) N_t + 2N_r \sigma_i^2 \right. \\ & \quad \left. + 2E \left(\left\| \mathbf{H} - \widehat{\mathbf{H}}_{est} \right\|_F \right) \lambda \mu \right) \end{aligned} \quad (14)$$

where N_t is the number of transmit antennas and N_r is the number of receive antennas in the wireless MIMO configuration. We can see from Equation (14) that the minimization of $E \left(\left\| \mathbf{Y}_i - \widehat{\mathbf{H}}_{est} \mathbf{X}_i \right\|_F^2 \right)$ implies minimizing the channel estimate MSE $E \left(\left\| \mathbf{H} - \widehat{\mathbf{H}}_{est} \right\|_F^2 \right)$ term. Using this fact, we can then link the channel estimation MSE minimization to the minimization of the equivalent noise power term. Based on Equation(12), $E \left(\left\| \mathbf{Y}_i - \widehat{\mathbf{H}}_{est} \mathbf{X}_i \right\|_F^2 \right) = E \left(\left\| -N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i + \mathbf{N}_i \right\|_F^2 \right)$. Therefore, we can minimize the average signal mean squared error $E \left(\left\| \mathbf{Y}_i - \widehat{\mathbf{H}}_{est} \mathbf{X}_i \right\|_F^2 \right)$ expression by minimizing its equivalent expression $E \left(\left\| -N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i + \mathbf{N}_i \right\|_F^2 \right)$. But

$$\begin{aligned} & E \left(\left\| -N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i + \mathbf{N}_i \right\|_F^2 \right) \\ & \geq E \left(\left\| \mathbf{N}_i \right\|_F^2 + \left\| N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i \right\|_F^2 \right. \\ & \quad \left. - 2 \left\| \mathbf{N}_i \right\|_F \left\| N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i \right\|_F \right) \end{aligned} \quad (15)$$

But we know that $E \left(\left\| -N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i + \mathbf{N}_i \right\|_F^2 \right) \geq 0 \Rightarrow E \left(\left\| \mathbf{N}_i \right\|_F^2 \right) + E \left(\left\| N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i \right\|_F^2 \right) \geq E \left(2 \left\| \mathbf{N}_i \right\|_F \left\| N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i \right\|_F \right) \forall \alpha$.

It is easy to see that when the expression $E \left(\left\| \mathbf{N}_i \right\|_F^2 \right) + E \left(\left\| N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i \right\|_F^2 \right)$ increases in value, the expression $E \left(2 \left\| \mathbf{N}_i \right\|_F \left\| N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i \right\|_F \right)$ also increases in value, and the converse is true, which implies that the two expressions are in phase. This means that the mathematical expression, $E \left(2 \left\| \mathbf{N}_i \right\|_F \left\| N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i \right\|_F \right)$, only shifts or translates the graph of $E \left(\left\| \mathbf{N}_i \right\|_F^2 \right) + E \left(\left\| N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i \right\|_F^2 \right)$ vertically downwards on the Cartesian plane. It does not affect the α value where the minimum or stationary point occurs. Thus, to minimize the term $E \left(\left\| -N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i + \mathbf{N}_i \right\|_F^2 \right)$, we need to

only concentrate on minimizing the term $E \left(\left\| \mathbf{N}_i \right\|_F^2 \right) + E \left(\left\| N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i \right\|_F^2 \right)$.

However, knowing that the total equivalent noise power, henceforth the equivalent noise power, is the total noise power contribution from the noise matrix expressions $N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i$ and \mathbf{N}_i . We can define the equivalent noise power as follows in Equation (16):

$$\sigma_{equ}^2 \triangleq E \left(\left\| \mathbf{N}_i \right\|_F^2 \right) + E \left(\left\| N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i \right\|_F^2 \right) \quad (16)$$

As can be seen, that Equation (15) contains the equivalent noise power; thus, for us to minimize the channel estimation MSE, we can find the minimum or lowest equivalent noise power. So mathematically, this is noted as follows in Equation (17):

$$\begin{aligned} \operatorname{argmin}_{\alpha} \sigma_{equ}^2 & \Rightarrow \operatorname{argmin}_{\widehat{\mathbf{H}}_{est}} E \left(\left\| \mathbf{Y}_i - \widehat{\mathbf{H}}_{est} \mathbf{X}_i \right\|_F^2 \right) \\ & \Rightarrow \operatorname{argmin}_{\widehat{\mathbf{H}}_{est}} E \left(\left\| \mathbf{H} - \widehat{\mathbf{H}}_{est} \right\|_F^2 \right) \end{aligned} \quad (17)$$

Based on Equation (16), we expand the expression $E \left(\left\| N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i \right\|_F^2 \right)$ using the following inequality $E \left(\left\| N_r \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \mathbf{X}_i \right\|_F^2 \right) \leq E \left(\left\| N_r \right\|_F^2 \left\| \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1} \right\|_F^2 \left\| \mathbf{X}_i \right\|_F^2 \right)$.

Now let $\boldsymbol{\theta} = \mathbf{X}_r^H \left(\mathbf{X}_r \mathbf{X}_r^H \right)^{-1}$ where $\boldsymbol{\theta} \in \mathbb{C}^{N \times N_t}$ matrix. The matrix $\boldsymbol{\theta}$ has constant complex entries containing the fixed transmitted pilot symbol matrix \mathbf{X}_r . Therefore, the equivalent noise power becomes Equation (18)

$$\sigma_{equ}^2 \leq 2N_r \sigma_i^2 + E \left(\left\| N_r \right\|_F^2 \right) E \left(\left\| \boldsymbol{\theta} \right\|_F^2 \right) E \left(\left\| \mathbf{X}_i \right\|_F^2 \right) \quad (18)$$

since matrices N_r , $\boldsymbol{\theta}$ and \mathbf{X}_i are independent. But $E \left(\left\| N_r \right\|_F^2 \right) = N_r N \sigma_r^2$, $E \left(\left\| \boldsymbol{\theta} \right\|_F^2 \right) = \left\| \boldsymbol{\theta} \right\|_F^2$, and $E \left(\left\| \mathbf{X}_i \right\|_F^2 \right) = 2N_t$ thus, the equivalent noise power upper bound is given in Equation (19)

$$\sigma_{equ}^2 \leq 2N_r \sigma_i^2 + 2N_r N_t N \sigma_r^2 \left\| \boldsymbol{\theta} \right\|_F^2 \quad (19)$$

Using Equations (6.1) and (6.2), we can get the complete equivalent noise power upper bound as stated in Equation (20)

$$\sigma_{equ}^2 \leq \frac{4N_r}{(1-\alpha)\gamma} + \frac{4N_r N_t N \left\| \boldsymbol{\theta} \right\|_F^2}{\left(1 + \frac{M}{N} \alpha\right) \gamma} \quad (20)$$

To get the optimal power fraction that minimizes the equivalent noise power, we rely on the calculus of finding the stationary point of the equivalent noise power with respect to the transmit power fraction. $\frac{d\sigma_{equ}^2}{d\alpha} = 0$, solving this leads to the following optimal power fraction as shown in Equation (21)

$$\alpha_{opt} = \frac{N \sqrt{MN_t} \left\| \boldsymbol{\theta} \right\|_F - N}{M + N \sqrt{MN_t} \left\| \boldsymbol{\theta} \right\|_F} \quad (21)$$

where α_{opt} is the optimal transmit power fraction, M is the number of information symbols transmitted per transmit

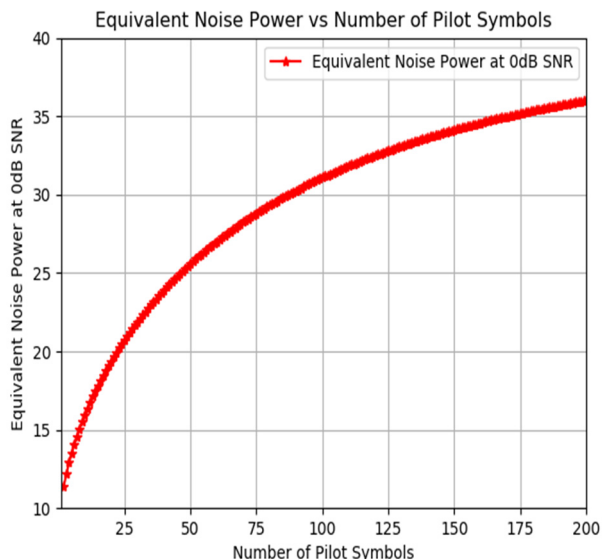


FIGURE 2. Shows the equivalent noise power at 0dB SNR versus number of pilot symbols N .

antenna, N is the number of pilot symbols sent per transmit antenna, N_t is the number of transmit antennas in the MIMO configuration and $\|\theta\|_F$ is the Frobenius norm of the matrix $\theta = X_r^H (X_r X_r^H)^{-1}$.

The next objective is to find the optimal number of pilot symbols that must be transmitted over the wireless channel and used for channel estimation. Since it follows that getting a minimal or lowest equivalent noise power translates to minimum channel estimation MSE, we need to select the number of pilot symbols that produce the lowest possible equivalent noise power. The optimal power fraction that minimizes the equivalent noise power is a function of the number of pilot symbols N . Thus, the critical parameter to select for optimal channel estimation performance is the number of pilot symbols N since the optimal power fraction can be obtained from Fig. 3 after finding the optimal N . We can find the optimal number of pilot symbols N from Fig. 2.

As can be seen in Fig. 2, the realistic values for the number of pilot symbols can only be in the range 2 to 200 since the quasi-static channel fading is constant for slightly more than 200 symbols at a time, hence the limit of 200. However, we cannot use $N = 1$ pilot symbol because the Zadoff-Chu sequence always starts with an element with an amplitude of 1 and a phase of 0° . This means that for $N = 1$, we have a singular square matrix created by $X_r X_r^H$, which is not invertible. We, therefore, can only work with values of N in the range 2 to 200. This is our search space for our optimal number of pilot symbols. The SNR is set to 0dB because it plays an insignificant role in determining the optimal number of pilot symbols since the SNR is just a scaling factor. Setting the SNR to 0dB is equivalent to setting a scaling factor of 1 on the linear scale.

Fig. 2 exhibits the fact that the lowest equivalent noise power value is observed at $N = 2$ pilot symbols. As shown in Section III-C's derivation, the lowest equivalent noise power

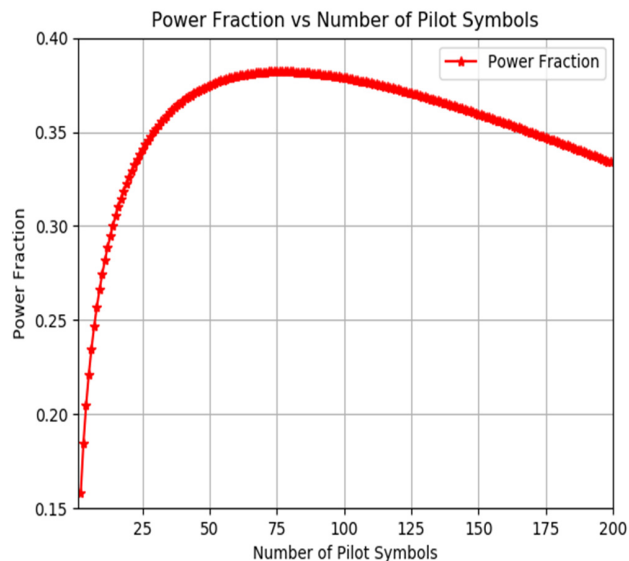


FIGURE 3. Shows the power fraction versus number of pilot symbols N .

value corresponds to the lowest MSE performance. This translates to an optimal power fraction $\alpha_{opt} \approx 0.16$ based on extrapolation from Fig. 3 at $N = 2$. Fig. 3 is produced from Equation (21) and Fig. 2 is produced from Equation (20) with the transmit power fraction set as $\alpha = \alpha_{opt}$.

From Fig. 3, we can see that the optimal power fraction stays within the feasible range of $[0, 1)$ over a different number of pilot symbol values. The other observation is that the power fraction cannot increase and approach unity linearly because the transmit power of the information symbols will become negligible, negatively affecting the BER performance of the transmitted information symbols. Thus whilst increasing the transmit power of the pilot symbols may yield an excellent channel estimation MSE performance, lowering the transmit power of the information symbols close to 0 will yield inferior BER performances, which will defeat the aim of improving the channel estimation accuracy. Thus, a perfect balance must be struck to optimize the BER and MSE performance.

IV. SIMULATION RESULTS

The Monte-Carlo wireless simulation environment is setup as a 2×4 multiple-input-multiple-output (MIMO) wireless channel with Rayleigh quasi-static fading in which the channel gain remains constant for $200 + N$ symbol durations and changes every $200 + N$ symbol durations. The wireless transmit and receive antennas are sufficiently spaced enough such that the wireless channels are de-correlated. The number of information symbols transmitted per frame is $M = 200$ and the number of pilot symbols transmitted per frame is $N = 2$ or $N = 10$. The information and pilot symbols share a fraction of the transmit power based on the optimal power fraction in Equation (21) for $N = 2$. The pilot symbols are generated using the Zadoff-Chu sequence in Equation (1) and based on the number of pilot symbols $N = 2$ or $N = 10$. The

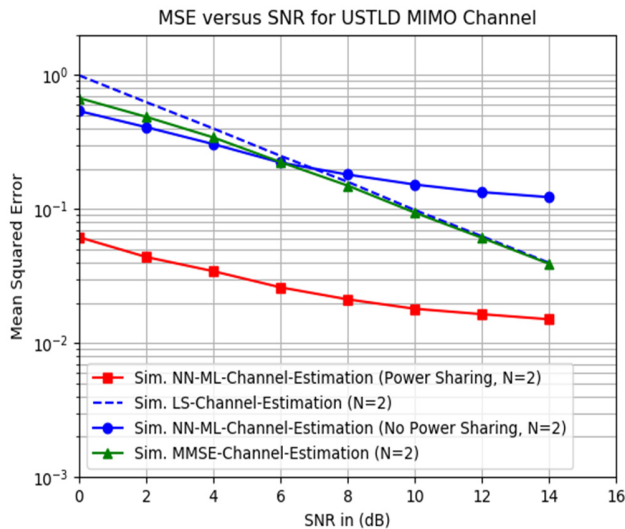


FIGURE 4. Shows the MSE performance of the 2×4 MIMO NN-ML channel estimation vs traditional channel estimation methods.

information symbol modulation order used in the simulation is 16-QAM and 16-PSK. The average power constraint for the 16-QAM and 16-PSK symbols is set to 1.

The NN-ML-based algorithm model architecture seen in Table 1 is loaded into the simulation environment, and the saved optimal synaptic weights are loaded into the machine learning model. The channel estimation algorithms used are the LS, MMSE, NN-ML without power-sharing, and NN-ML with power-sharing. The Monte-Carlo simulation determines the channel estimation algorithms' MSE and BER performance over 0dB to 14dB SNR.

From Fig. 4 we can see that the NN-ML channel estimator without transmit power-sharing underperforms the traditional channel estimation methods LS and MMSE at high SNR. The reason for this is that the hyperparameters in Table 1 are searched for in a multivariable landscape with the objective to find the global minimum validation MSE at 10dB SNR. Because the landscape is multivariable, it lends itself to the possibility of having multiple local stationary points and a single global stationary point. The genetic algorithm (GA) used in Algorithm 1 tries to search for the global stationary point in the multivariable landscape with no guarantees of finding the global stationary point. Our GA runs for only 100 evolutionary generations, with a population size of 10, and this limits the number of permutations of hyperparameter values tested on the neural network. Thus, it restricts the search space for the global stationary point. The hyperparameters found in Table 1 for the case $\alpha = 0$, are clearly suboptimal based on the poor MSE performance at high SNR. The higher SNR range is sensitive to the channel estimator's systematic errors as opposed to the low SNR range.

We also see from Fig. 4 that the NN-ML channel estimation method has an improved MSE performance when transmit power-sharing is used to improve its performance. From Fig. 4, the added transmit power fraction for the pilot symbols

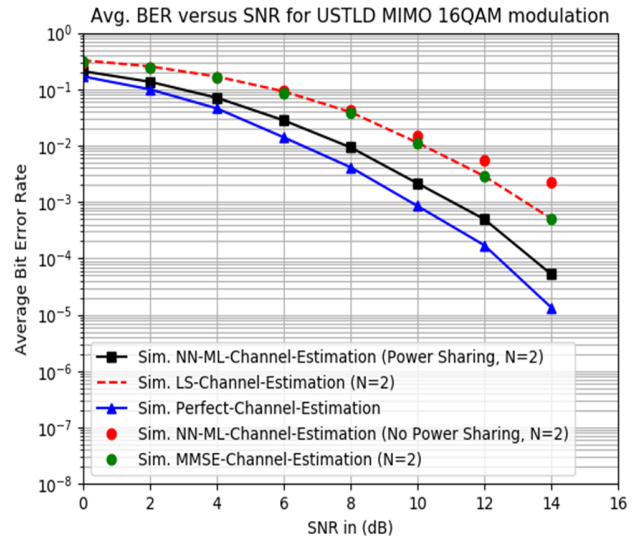


FIGURE 5. Shows the BER performance of the 2×4 MIMO NN-ML channel estimation vs traditional channel estimation methods at the same number of pilot symbols for 16-QAM modulation.

improves the MSE performance. It is also interesting to note that for the defined SNR range, the NN-ML channel estimator with transmit power-sharing outperforms the traditional LS and MMSE methods. At $MSE = 6 \times 10^{-2}$, we have an approximately 12dB gain over the traditional LS and MMSE channel estimation when using the NN-ML with power-sharing channel estimation algorithm. This implies that we can save transmit power with the NN-ML with the power-sharing method; we can also do blind channel estimation without knowledge of the transmitted pilot symbols and/or channel autocorrelation statistics or the received noise variance required by the traditional channel estimation methods.

As shown in Fig. 5, the NN-ML channel estimator's BER performance with transmit power-sharing is the best performing as expected from the MSE accuracy shown in Fig. 4. This shows that minimizing the equivalent noise power does, in fact, minimize the channel estimation MSE performance as shown in Fig. 4 and the signal MSE performance as shown in Fig. 5. The signal MSE performance is linked to the maximum likelihood detector performance. Hence, the NN-ML channel estimator's BER performance with transmit power-sharing has the best BER performance since its equivalent noise power is minimized relative to the other channel estimation algorithms.

There is a loss in diversity at high SNR for the NN-ML channel estimator without transmit power-sharing as the hyperparameters selected for the case $\alpha = 0$ are suboptimal. Fig. 4 MSE performance shows that at high SNR the NN-ML channel estimator without transmit power-sharing has poor performance relative to the other channel estimators. This poor MSE performance has an impact on the BER performance, as shown in Fig. 5. For the NN-ML channel estimator with transmit power-sharing, there is no loss in

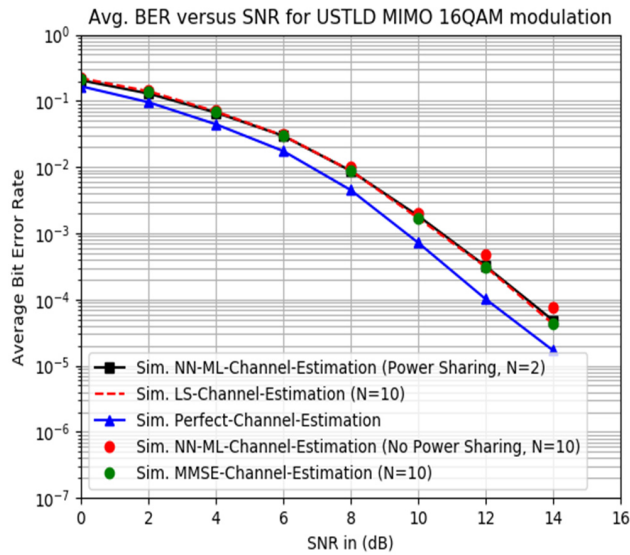


FIGURE 6. Shows the BER performance of the 2×4 MIMO NN-ML channel estimation vs traditional channel estimation methods at different number of pilot symbols for 16-QAM modulation.

diversity because the hyperparameters selected for the case when $\alpha = \alpha_{opt}$ are near-optimal. The approach of selecting hyperparameters using a GA does not guarantee that the stationary points found are globally optimal.

We also observe a near 2dB BER performance gain between the NN-ML channel estimator with transmit power-sharing and the traditional LS and MMSE channel estimation methods. This implies that the NN-ML channel estimator with transmit power-sharing enables a good channel estimation accuracy and link reliability relative to the traditional channel estimation algorithms whilst using a minimal number of pilot symbols to estimate the wireless channel. As shown in Fig. 6, the NN-ML channel estimator with transmit power-sharing has the same BER performance, at $N = 2$, with the traditional LS and MMSE channel estimation methods at $N = 10$. This means that we need 8 extra pilot symbols for the LS and MMSE channel estimation methods to deliver a similar BER performance as the NN-ML channel estimator with transmit power-sharing. That is a waste of expensive wireless channel bandwidth, which should be used to transmit information symbols.

The other observation is that the NN-ML channel estimator without transmit power-sharing, at higher N values, as observed in Fig. 6, has a BER performance that approaches that of the traditional LS and MMSE methods. This is advantageous because totally blind channel estimation can be achieved by this NN-ML method as it only needs the observed/received pilot symbols matrix to perform channel estimation. This means that it does not need to know the wireless channel second-order statistics, nor does it need to know the noise variance. Neither does it need to know the transmitted pilot symbol matrix, unlike the traditional channel estimation methods that require this prior knowledge.

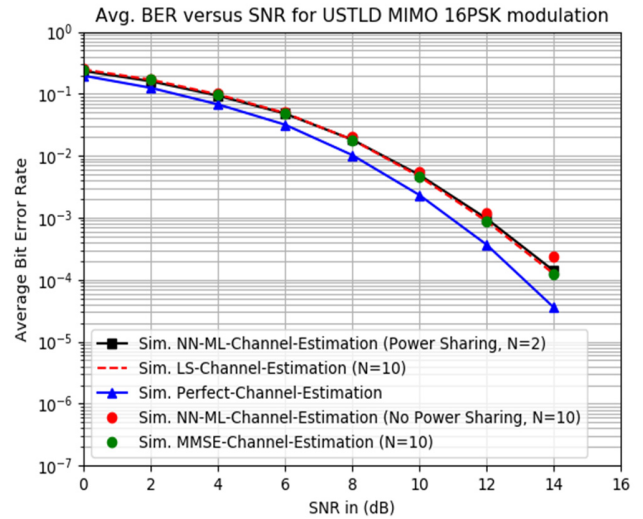


FIGURE 7. Shows the BER performance of the 2×4 MIMO NN-ML channel estimation vs traditional channel estimation methods at different number of pilot symbols for 16-PSK modulation.

In Fig. 7, we see that the same performance gains achieved in Fig. 6 for 16-QAM modulation apply to 16-PSK modulation. This implies that 16-QAM or 16-PSK USTLD modulation in conjunction with NN-ML channel estimator with transmit power-sharing can achieve a comparable BER performance to the traditional channel estimation methods but at 20% of the bandwidth required by the traditional channel estimators LS and MMSE.

V. CONCLUSION

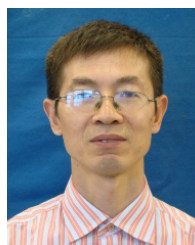
The power-sharing method improves the NN-ML channel estimation MSE accuracy relative to the NN-ML method without transmit power-sharing. The MSE performance of the NN-ML channel estimator algorithm with transmit power-sharing is very good throughout the SNR range relative to the traditional LS and MMSE channel estimation methods. The NN-ML channel estimator with transmit power-sharing MSE performance, at $MSE = 6 \times 10^{-2}$, has an approximately 12dB gain over the traditional LS and MMSE channel estimation methods. The NN-ML algorithm does not require knowledge of the channel autocorrelation statistics and the noise variance to estimate the wireless channel. This implies that the NN-ML algorithm with power-sharing can be used for wireless channel estimation where the transmitted pilot symbols, channel second-order statistics and receiver noise variance are unknown. Another inference from the results is that, since the optimal number of pilot symbols is only 2, it means we can achieve high channel estimation MSE/BER accuracy whilst saving expensive channel bandwidth since the traditional channel estimation algorithms will need a higher number of pilot symbols to achieve similar MSE/BER performance. From the simulation results, the proposed NN-ML channel estimator with transmit power-sharing requires only 20% of the bandwidth utilized by LS and MMSE to achieve the same BER performance for 16-QAM and 16-PSK USTLD modulation.

REFERENCES

- [1] H. Xu, K. Govindasamy, and N. Pillay, "Uncoded space-time labeling diversity," *IEEE Commun. Lett.*, vol. 20, no. 8, pp. 1511–1514, Aug. 2016.
- [2] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 8, pp. 1451–1458, Oct. 1998.
- [3] S. S. Patel, T. Quazi, and H. Xu, "A genetic algorithm for designing uncoded space-time labelling diversity mappers," in *Proc. IEEE Int. Workshop Signal Process. Syst. (SiPS)*, Oct. 2018, pp. 1–6.
- [4] D. Ayanda and H. Xu, "A unified error analysis of un-coded space-time labeling diversity with three transmit antennas in Rician fading channels," in *Proc. IEEE 2nd Wireless Africa Conf. (WAC)*, Aug. 2019, pp. 1–5.
- [5] N. Pillay and H. Xu, "Uncoded space-time labeling diversity—Application of media-based modulation with RF mirrors," *IEEE Commun. Lett.*, vol. 22, no. 2, pp. 272–275, Feb. 2018.
- [6] T. Quazi and H. Xu, "SSD-enhanced uncoded space-time labeling diversity," *Int. J. Commun. Syst.*, vol. 31, no. 11, p. e3592, 2018.
- [7] B. Dlodlo and H. Xu, "Trellis code-aided high-rate M-QAM space-time labeling diversity using a unitary expansion," *Int. J. Commun. Syst.*, vol. 31, no. 11, p. e3590, 2018.
- [8] D. Ayanda, H. Xu, and N. Pillay, "Uncoded M-ary quadrature amplitude modulation space-time labeling diversity with three transmit antennas," *Int. J. Commun. Syst.*, vol. 31, no. 18, p. e3818, 2018.
- [9] D. Ayanda, S. Mughal, and K. A. Abdulsalam, "Performance analysis of rectangular QAM uncoded space-time labeling diversity over Nakagami-m fading channels," in *Proc. IEEE Microw. Theory Techn. Wireless Commun. (MTTW)*, Riga, Latvia, Oct. 2020, pp. 163–167.
- [10] D. Ayanda, H. Xu, and S. Mughal, "High-density M-QAM mapper design for uncoded space-time labeling diversity with three transmit antennas over Nakagami-m fading channels," in *Proc. IEEE AFRICON*, Accra, Ghana, Sep. 2019, pp. 1–7.
- [11] M. Soltani, V. Pourahmadi, A. Mirzaei, and H. Sheikhzadeh, "Deep learning-based channel estimation," *IEEE Commun. Lett.*, vol. 23, no. 4, pp. 652–655, Apr. 2019.
- [12] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "Deep learning-based channel estimation for beamspace mmWave massive MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 852–855, Oct. 2018.
- [13] Y. Yang, F. Gao, X. Ma, and S. Zhang, "Deep learning-based channel estimation for doubly selective fading channels," *IEEE Access*, vol. 7, pp. 36579–36589, 2019.
- [14] S. Gao, P. Dong, Z. Pan, and G. Y. Li, "Deep learning based channel estimation for massive MIMO with mixed-resolution ADCs," *IEEE Commun. Lett.*, vol. 23, no. 11, pp. 1989–1993, Nov. 2019.
- [15] J.-M. Kang, C.-J. Chun, and I.-M. Kim, "Deep-learning-based channel estimation for wireless energy transfer," *IEEE Commun. Lett.*, vol. 22, no. 11, pp. 2310–2313, Nov. 2018.
- [16] Q. Bai, J. Wang, Y. Zhang, and J. Song, "Deep learning-based channel estimation algorithm over time selective fading channels," *IEEE Trans. Cognit. Commun. Netw.*, vol. 6, no. 1, pp. 125–134, Mar. 2020.
- [17] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 114–117, Feb. 2018.
- [18] X. Cheng, D. Liu, C. Wang, S. Yan, and Z. Zhu, "Deep learning-based channel estimation and equalization scheme for FBMC/OQAM systems," *IEEE Wireless Commun. Lett.*, vol. 8, no. 3, pp. 881–884, Jun. 2019.
- [19] Y. Liao, Y. Hua, X. Dai, H. Yao, and X. Yang, "ChanEstNet: A deep learning based channel estimation for high-speed scenarios," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Shanghai, China, May 2019, pp. 1–6.
- [20] J. P. Nair and R. V. R. Kumar, "A bandwidth efficient channel estimation method using superimposed training for MIMO-OFDM systems," in *Proc. IEEE Region Conf. (TENCON)*, Hyderabad, India, Nov. 2008, pp. 1–5.
- [21] W. G. Jeon, K. H. Paik, and Y. S. Cho, "An efficient channel estimation technique for OFDM systems with transmitter diversity," in *Proc. 11th IEEE Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, London, U.K., Sep. 2000, pp. 1246–1250.
- [22] C. Shin, R. W. Heath, Jr., and E. J. Powers, "Blind channel estimation for MIMO-OFDM systems," *IEEE Trans. Veh. Technol.*, vol. 56, no. 2, pp. 670–685, Mar. 2007.
- [23] K. Josiam and D. Rajan, "Bandwidth efficient channel estimation using super-imposed pilots in OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 6, no. 6, pp. 2234–2245, Jun. 2007.
- [24] J. Siew, J. Coon, R. Piechocki, A. Nix, M. Beach, S. Armour, and J. McGeehan, "A bandwidth efficient channel estimation algorithm for MIMO-SCFDE," in *Proc. IEEE 58th Veh. Technol. Conf. (VTC-Fall)*, Orlando, FL, USA, Oct. 2003, pp. 1142–1146.
- [25] A. Koohian, H. Mehrpouyan, M. Ahmadian, and M. Azarbad, "Bandwidth efficient channel estimation for full duplex communication systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, London, U.K., Jun. 2015, pp. 4710–4714.
- [26] M. Pukkila, "Channel estimation modeling. Post-graduate course in radio-communications," Nokia Res. Center, Helsinki, Finland, Tech. Rep. S-72.333, 2000.
- [27] C. R. Murthy, A. K. Jagannatham, and B. D. Rao, "Training-based and semiblind channel estimation for MIMO systems with maximum ratio transmission," *IEEE Trans. Signal Process.*, vol. 54, no. 7, pp. 2546–2558, Jul. 2006.
- [28] K. Kadathlal, H. Xu, and N. Pillay, "Generalised differential scheme for spatial modulation systems," *IET Commun.*, vol. 11, no. 13, pp. 2020–2026, Sep. 2017.
- [29] M. PremKumar, M. P. Chitra, M. Arun, and M. S. Saravanan, "Least squares based channel estimation approach and bit error rate analysis of cognitive radio," in *Proc. Int. Conf. Robot., Automat., Control Embedded Syst. (RACE)*, Chennai, India, Feb. 2015, pp. 1–4.
- [30] M. Simko, C. Mehlhruher, M. Wrulich, and M. Rupp, "Doubly dispersive channel estimation with scalable complexity," in *Proc. Int. ITG Workshop Smart Antennas (WSA)*, Feb. 2010, pp. 251–256.
- [31] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Reading, MA, USA: Addison-Wesley, 1989.
- [32] N. Samuel, T. Diskin, and A. Wiesel, "Learning to detect," *IEEE Trans. Signal Process.*, vol. 67, no. 10, pp. 2554–2564, May 2019.
- [33] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," Dec. 2014, *arXiv:1412.6980*. [Online]. Available: <https://arxiv.org/abs/1412.6980>
- [34] J. M. Steele, *The Cauchy–Schwarz Master Class: An Introduction to the Art of Mathematical Inequalities*. Washington, DC, USA: The Mathematical Association of America, 2004.



BHEKISIZWE MTHETHWA (Member, IEEE) received the B.Sc.Eng. degree in electronics and the M.Sc.Eng. degree (*summa cum laude*) in electronics from the University of KwaZulu-Natal, in 2008 and 2013, respectively. He is currently pursuing the Ph.D. degree, under the guidance of Dr. H. Xu. He has more than ten years industrial experience ranging from Mining consulting as a Control and Instrumentation Engineer to a Senior System Engineer within the Research and Development space. He has been involved in projects from designing and constructing furnaces, software application development, and system engineering work. He is also a registered Professional Engineer with the Engineering Council of South Africa (ECSA). His research interests include spatial modulation, MIMO space-time block coding schemes, and the application of deep learning to the design of wireless systems.



HONGJUN XU (Member, IEEE) received the B.Sc. degree from the Guilin Institute of Electronic Technology (nowadays it is called the Guilin University of Electronic Technology), Guilin, China, in 1984, the M.Sc. degree from the Institute of Telecontrol and Telemeasure, Shi Jian Zhuang, China, in 1989, and the Ph.D. degree from the Beijing University of Aeronautics and Astronautics, Beijing, China, in 1995.

From 1997 to 2000, he was a Postdoctoral Fellow with the University of Natal (Nowadays it is called the University of KwaZulu-Natal), Durban, South Africa, and Inha University, Incheon, South Korea. He joined the University of Natal as a Lecturer in 2002. He has been a Full Professor with the School of Engineering, University of KwaZulu-Natal, since 2011. He is also a Rated Scientist of the National Research Foundation in South Africa.

• • •