Received 6 February 2021; accepted 23 February 2021. Date of publication 2 March 2021; date of current version 11 March 2021.

Digital Object Identifier 10.1109/OJCOMS.2021.3063171

Channel Estimation for Reconfigurable Intelligent Surface Aided MISO Communications: From LMMSE to Deep Learning Solutions

NEEL KANTH KUNDU (Graduate Student Member, IEEE), AND MATTHEW R. MCKAY (Fellow, IEEE)

Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong

CORRESPONDING AUTHOR: M. R. MCKAY (e-mail: m.mckay@ust.hk)

This work was supported by the Hong Kong Research Grants Council under Grant 16202918 and Grant C6012-20G. The work of Neel Kanth Kundu was supported by the Hong Kong Ph.D. Fellowship Scheme under Grant PF17-00157.

A preliminary version of this work was presented at the 2020 IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC) [1].

ABSTRACT We consider multi-antenna wireless systems aided by reconfigurable intelligent surfaces (RIS). RIS presents a new physical layer technology for improving coverage and energy efficiency by intelligently controlling the propagation environment. In practice however, achieving the anticipated gains of RIS requires accurate channel estimation. Recent attempts to solve this problem have considered the least-squares (LS) approach, which is simple but also sub-optimal. The optimal channel estimator, based on the minimum mean-squared-error (MMSE) criterion, is challenging to obtain and is non-linear due to the non-Gaussianity of the effective channel seen at the receiver. Here we present approaches to approximate the optimal MMSE channel estimator. As a first approach, we analytically develop the best linear estimator, the LMMSE, together with a corresponding majorization-minimization-based algorithm designed to optimize the RIS phase shift matrix during the training phase. This estimator is shown to yield improved accuracy over the LS approach by exploiting second-order statistical properties of the wireless channel and the noise. To further improve performance and better approximate the globally-optimal MMSE channel estimator, we propose data-driven non-linear solutions based on deep learning. Specifically, by posing the MMSE channel estimation problem as an image denoising problem, we propose two convolutional neural network (CNN)-based methods to perform the denoising and approximate the optimal MMSE channel estimation solution. Our numerical results show that these CNN-based estimators give superior performance compared with linear estimation approaches. They also have low computational complexity requirements, thereby motivating their potential use in future RIS-aided wireless communication systems.

INDEX TERMS Reconfigurable intelligent surface, MISO, LMMSE, MMSE, majorization-minimization, deep learning, convolutional neural network, channel estimation, achievable rate.

I. INTRODUCTION

FUTURE communication systems will need to support billions of connected devices, while offering extremely high data rates and system reliability [2]. Meeting these challenges will require the development of innovative physical layer technologies. Various approaches are being explored, such as the development of new antenna technologies [3], multiplexing schemes [4], and signaling strategies [5], [6].

Reconfigurable intelligent surfaces (RIS), alternatively labelled intelligent reflecting surfaces (IRS), are also attracting considerable attention, and are being envisioned as a promising physical layer technology for enhancing the performance of future wireless systems [7]–[9]. RIS can potentially improve network coverage, and can enable spectral and energy efficiency gains that are commensurate with massive MIMO systems, but with much fewer antennas at the base station (BS) [10]. A distinctive feature of RIS is that they do not have active components like power amplifiers and analog-to-digital converters, but instead are composed of almost passive elements that can intelligently directionally

control the propagation of impinging electromagnetic waves to improve end-to-end performance.

Various recent contributions have considered the use of RIS to improve the performance of multiple-antenna communication systems. For such systems, improvements in network spectral efficiency and energy efficiency have been demonstrated through the use of joint active and passive beamforming at the BS and the RIS unit, respectively, [10]–[17]. A common assumption of these works is that perfect channel state information is available for optimizing the beamforming vectors. In practice however, channel state information must be estimated, which poses new challenges due to the almost passive nature of the RIS elements, which have no transmit or receive processing chains. A few works have attempted to tackle this issue by proposing a least-squares (LS) criterion to estimate the cascaded channel in the uplink (UL), and then using channel reciprocity to estimate the equivalent downlink (DL) channel required for active and passive beamforming design [18]–[22]. A different approach [23] has proposed to adopt an alternating LS algorithm, which separately estimates the BS to RIS channel and the RIS to user equipment (UE) channel (but does not consider the direct BS-UE link), assuming that pilots are transmitted from the BS. In general, while offering simple implementation complexity, the existing LS-based approaches are suboptimal, as they do not take into account statistical features of the channel and the noise.

In this work, we consider the design of optimal channel estimation strategies for RIS-aided multiple antenna communications, based on the minimum mean squared error (MMSE) criterion. Currently, there is little work on the design of MMSE channel estimators for RIS-aided communications, with the exception of the recent contribution [24], which considered a configuration with a line-of-sight channel between the BS and RIS, known perfectly to the BS. This strong assumption leads to a linear Gaussian measurement model, and gives a closed form expression for the MMSE channel estimate.

Here we consider a general scenario in which all channels (BS to RIS, RIS to UE, and BS to UE) are subject to Rayleigh fading and are unknown. The optimal MMSE estimator in this case does not admit a closed-form analytical solution, due to the complicated statistical properties of the cascaded channel between the BS, RIS and UE (which does not follow a Gaussian distribution). To deal with this problem, first we derive the best *linear* estimator that minimizes the mean squared error, i.e., the LMMSE, which we show admits a closed-form expression depending on the second-order statistics of the channel and the additive noise. The LMMSE performance depends on the phase shifts employed at the RIS during channel estimation, which may be optimized. Finding the optimum phase shifts is complex however, since it involves solving a non-convex optimization problem. To approximately solve this problem, we present an algorithm based on the majorization-minimization (MM) principle, which presents an efficient numerical solution. The MM-based algorithm demonstrates the existence of numerous locally-optimal RIS phase-shift matrices. These include the analytical DFT phase-shift matrix that has been shown to be optimal for the case of LS channel estimators [20]. The DFT-based phase-shift matrix also presents a desirable practical solution for LMMSE, which we show also allows for tractable analytical performance characterization.

We then move beyond the class of linear filters to present algorithms for approximating the optimal (non-linear) MMSE channel estimator for RIS-aided multiple-antenna communications. To this end, we introduce data-driven deep learning approaches. Specifically, we propose convolutional neural network (CNN)-based channel estimators that approximate the optimal MMSE solution. Our approach is to consider a linear LS channel estimate as a noisy 'image' at the neural network input, and then to apply a CNN-based image denoising network to 'clean' this image and yield an improved RIS channel estimate. We consider two CNN-based architectures, Denoising CNN (DnCNN) [25] and Fast and Flexible Denoising Network (FFDNet) [26], that have different properties and performance trade-offs. Our numerical simulations show that the CNN-based channel estimators can offer significant performance improvements when compared to the linear estimators, LMMSE and LS. The high performance and reasonably low implementation complexity of these CNN-based channel estimators makes them promising candidates for practical incorporation into future RIS-aided wireless communication systems.

The rest of the paper is organized as follows. Section II presents the system model, defines the channel estimation problem, and reviews the existing LS channel estimator. Section III presents our derivation of the LMMSE channel estimator, and develops the MM-based algorithm to optimize the RIS phase-shift matrix. In Section IV, we introduce the CNN-based channel estimators, while Section V presents numerical results. Finally, some concluding remarks are offered in Section VI.

Notation: Matrices and vectors are represented by boldface upper case ($\boldsymbol{A}$) and boldface lower case ($\boldsymbol{a}$) letters respectively. Hermitian (conjugate transpose), inverse, and trace of a matrix $\boldsymbol{A}$ are denoted by $\boldsymbol{A}^H$, $\boldsymbol{A}^{-1}$ and $\mathrm{tr}(\boldsymbol{A})$, respectively. $\mathrm{diag}(\boldsymbol{a})$ with $\boldsymbol{a} \in \mathbb{C}^N$ returns a $N \times N$ diagonal matrix with $\boldsymbol{a}$ on its diagonal, and $\mathrm{vec}(\boldsymbol{A})$ returns a vector by stacking the columns of $\boldsymbol{A}$. $\mathbb{E}[\cdot]$ denotes the expectation operator and $\otimes$ the Kronecker product. $\boldsymbol{B} = [\boldsymbol{A}]_{i:j,k:l}$ denotes the matrix formed by the elements in the $i-$th to $j-$th rows and $k-$th to $l-$th columns of $\boldsymbol{A}$. $\mathbf{1}_N, \mathbf{0}_N \in \mathbb{C}^N$ represent the vector of all ones and all zeros, respectively.

II. SYSTEM MODEL

We consider a multiple-input single-output (MISO) communication system where a BS with M antennas serves a single antenna UE with the help of an RIS having K passive elements; see Fig. 1. The RIS elements can intelligently control the phase of the incoming electromagnetic wave.

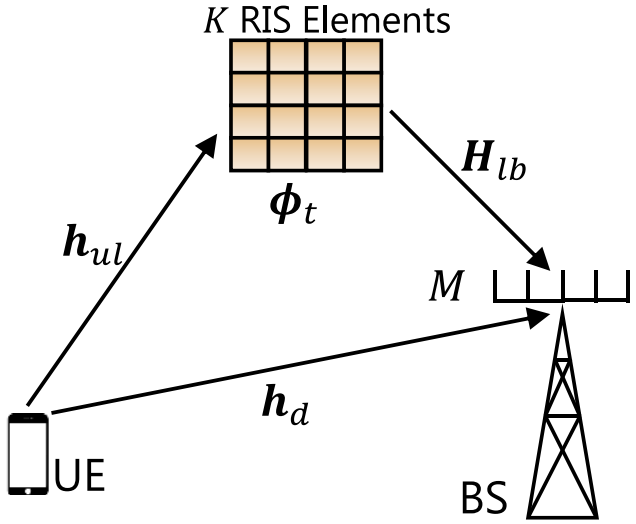


FIGURE 1. Schematic of the RIS-aided MISO communication system.

Assuming the system operates in time-division duplex mode, the DL channel can be estimated from the UL channel, due to channel reciprocity. The RIS elements are passive with no computing power, hence the UL channel is estimated at the BS. During training, the phase shifts of the RIS elements can be configured to assist the UL channel estimation. The received pilot signal at the BS during the t-th training step is given by

$$\boldsymbol{y}_t = \big(\boldsymbol{h}_d + \boldsymbol{H}_{lb}\text{diag}\big(\boldsymbol{\phi}_t\big)\boldsymbol{h}_{ul}\big)x_t + \boldsymbol{n}_t \tag{1}$$

where $\boldsymbol{y}_t \in \mathbb{C}^M$ is the received signal at the BS, and $x_t \in \mathbb{C}$, $|x_t| = 1$ is the transmitted pilot symbol from the UE. Also, $\boldsymbol{h}_d \in \mathbb{C}^M$ is the direct channel between the UE and BS, $\boldsymbol{H}_{lb} \in \mathbb{C}^{M\times K}$ is the channel matrix between the RIS and BS, $\boldsymbol{h}_{ul} \in \mathbb{C}^K$ is the channel between the UE and RIS, and $\boldsymbol{n}_t \sim \mathcal{CN}(0, \sigma^2\boldsymbol{I}_M)$ represents additive white Gaussian noise (AWGN) at the BS. The vector $\boldsymbol{\phi}_t = [e^{j\theta_{t,1}}, \ldots, e^{j\theta_{t,K}}]^T \in \mathbb{C}^K$ is the phase shift vector of the RIS, where $0 \le \theta_{t,k} \le 2\pi$, $k = 1, \ldots, K$, $t = 1, \ldots, T_p$ is the phase shift of the $k-$th RIS element during the t-th training step. Throughout the paper, we define the training SNR as $\gamma_{\text{tr}} = 1/\sigma^2$.

Due to the passive nature of the RIS elements, the cascaded channel $\boldsymbol{V} := \boldsymbol{H}_{lb}\text{diag}(\boldsymbol{h}_{ul}) = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_K]$ is estimated, rather than separately estimating $\boldsymbol{H}_{lb}$ and $\boldsymbol{h}_{ul}$. The model (1) may be equivalently expressed in terms of $\boldsymbol{V}$ as [20]

$$\boldsymbol{y}_t = \big(\boldsymbol{h}_d + \boldsymbol{V}\boldsymbol{\phi}_t\big)x_t + \boldsymbol{n}_t. \tag{2}$$

We assume that the channels are subjected to quasi-static Rayleigh fading, which remain constant during the coherence time $T_c > T_p$, where T_p is the pilot duration. The direct channel vector between UE and BS is modeled as [27]

$$\boldsymbol{h}_d = \boldsymbol{R}_{ub}^{1/2}\boldsymbol{h}_{wd} \tag{3}$$

where $\boldsymbol{h}_{wd} \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I}_K)$, and $\boldsymbol{R}_{ub}$ is an $M \times M$ positive definite matrix with unit diagonal entries, representing the spatial correlation matrix at the BS. The channel matrix between the RIS and BS is modeled as [27]

$$\boldsymbol{H}_{lb} = \boldsymbol{R}_{lb}^{1/2}\boldsymbol{H}_{wlb}\boldsymbol{S}_{lb}^{1/2}, \tag{4}$$

where $\boldsymbol{H}_{wlb} \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I}_M \otimes \boldsymbol{I}_K)$ while $\boldsymbol{R}_{lb}$ and $\boldsymbol{S}_{lb}$ are $M \times M$ and $K \times K$ positive definite matrices with unit diagonal entries, representing the spatial correlation matrices at the UE and RIS, respectively, and $\boldsymbol{h}_{ul} \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I}_K)$. Collecting all received signals $\boldsymbol{y}_t$ from (1) across $t = 1, 2, \ldots, T_p$, it can be compactly written as a linear measurement model [20, eq. (8)]

$$\boldsymbol{y} = \boldsymbol{G}\boldsymbol{z} + \boldsymbol{n}, \tag{5}$$

where

$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \vdots \\ \boldsymbol{y}_{T_p} \end{bmatrix}, \quad \boldsymbol{n} = \begin{bmatrix} \boldsymbol{n}_1 \\ \boldsymbol{n}_2 \\ \vdots \\ \boldsymbol{n}_{T_p} \end{bmatrix}, \quad \boldsymbol{z} = \begin{bmatrix} \boldsymbol{h}_d \\ \boldsymbol{v}_1 \\ \vdots \\ \boldsymbol{v}_K \end{bmatrix} \tag{6}$$

represent the vector of observations, noise and unknowns respectively. Note $\boldsymbol{n} \sim \mathcal{CN}(0, \sigma^2\boldsymbol{I}_{MT_p})$, while [20, eq. (3)-(5)]

$$\boldsymbol{G} = \boldsymbol{X}\boldsymbol{\Psi}, \tag{7}$$

with $\boldsymbol{X} = \text{diag}([x_1\boldsymbol{1}_M, \ldots, x_{T_p}\boldsymbol{1}_M])$, $\boldsymbol{\Psi} = \boldsymbol{\Phi} \otimes \boldsymbol{I}_M$ and

$$\boldsymbol{\Phi} = \begin{bmatrix} 1 & \phi_{1,1} & \ldots & \phi_{1,K} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \phi_{T_p,1} & \ldots & \phi_{T_p,K} \end{bmatrix}. \tag{8}$$

The RIS channel estimation problem requires estimating the vector $\boldsymbol{z}$, which contains the direct and cascaded channels, from the linear measurement model (5).

The optimum estimate of $\boldsymbol{z}$ that minimizes the mean squared error (MSE) $\mathbb{E}[\|\boldsymbol{z} - \hat{\boldsymbol{z}}\|^2]$ is the MMSE estimate. It is given by [28, eq. (10.5)]

$$\hat{\boldsymbol{z}}_{\text{mmse}} = \mathbb{E}\big[\boldsymbol{z}|\boldsymbol{y}\big]. \tag{9}$$

For the RIS channel model, the unknown random vector $\boldsymbol{z}$ is not Gaussian, but rather it involves the cascaded channel, which is a product of independent Gaussians. This makes the posterior distribution $p(\boldsymbol{z}|\boldsymbol{y})$ complicated, and hence, makes it difficult to find $\hat{\boldsymbol{z}}_{\text{mmse}}$ in closed form. Prior works on RIS channel estimation have tackled this problem by instead using the sub-optimal LS channel estimator, reviewed next.

A. EXISTING LS CHANNEL ESTIMATION APPROACHES

The LS estimate of the vector $\boldsymbol{z}$ in (5) is given by [18]–[20]

$$\hat{\boldsymbol{z}}_{\text{ls}} = \arg\min_{\boldsymbol{z}} \|\boldsymbol{y} - \boldsymbol{G}\boldsymbol{z}\|_2^2 = \Big(\boldsymbol{G}^H\boldsymbol{G}\Big)^{-1}\boldsymbol{G}^H\boldsymbol{y} \tag{10}$$

which exists when $T_p \ge K + 1$. Using (5) and (10), the LS estimate can be expressed as

$$\hat{\boldsymbol{z}}_{\text{ls}} = \boldsymbol{z} + \boldsymbol{w} \tag{11}$$

where $\boldsymbol{w} \sim \mathcal{CN}(0, \sigma^2(\boldsymbol{G}^H\boldsymbol{G})^{-1})$. The quality of the LS estimate depends on $\boldsymbol{G}$, which in turn depends on the phase

shift matrix $\boldsymbol{\Phi}$ at the RIS. Different choices of $\boldsymbol{\Phi}$ have been proposed. The authors of [20] have shown that the choice that minimizes the estimation variance per element is given by the first $K+1$ columns of the $T_p \times T_p$ DFT matrix $[\boldsymbol{F}_{T_p}]_{t,k} = e^{-j2\pi(t-1)(k-1)/T_p}$. This choice of $\boldsymbol{\Phi}$ makes $\boldsymbol{G}^H\boldsymbol{G}$ a scaled identity matrix, giving the MSE [20, eq. (31)]

$$\text{MSE}_{\hat{z}_{\text{ls}}} = \frac{M(K+1)}{T_p}\sigma^2. \tag{12}$$

Another simpler method that has been proposed for $\boldsymbol{\Phi}$ is one that estimates the cascaded channel by sequentially switching on one RIS element at a time [18], [19], [29]. This so-called 'on-off' method is sub-optimal, producing a higher MSE than the DFT approach [20, eq. (17)]. Note that for both the DFT and on-off approaches, the $\boldsymbol{\Phi}$ is fixed, and hence it can be set prior to channel estimation commencing.

Unlike the MMSE estimator introduced in (9), the LS channel estimators are known to be sub-optimal since they do not use the prior knowledge of the channel distribution [30]. In the following, we will present enhanced channel estimation solutions, designed based on the MMSE criterion.

III. LMMSE CHANNEL ESTIMATOR

While it is difficult to find the optimum MMSE channel estimator in closed form, the best *linear* estimator that minimizes the MSE—the LMMSE estimator—admits a closed form expression, which we present in the following. This estimator, unlike LS, depends on the second order statistics of both the channel and the noise.

Using the linear measurement model (5), the LMMSE estimate of $\boldsymbol{z}$ can be expressed as [28, eq. (12.26)]

$$\hat{\boldsymbol{z}}_{\text{lmmse}} = \mathbb{E}[\boldsymbol{z}] + \boldsymbol{C}_{zz}\boldsymbol{G}^H\left(\boldsymbol{G}\boldsymbol{C}_{zz}\boldsymbol{G}^H + \sigma^2\boldsymbol{I}_{MT_p}\right)^{-1}\boldsymbol{y}, \tag{13}$$

where $\boldsymbol{C}_{zz} = \mathbb{E}[\boldsymbol{z}\boldsymbol{z}^H]$. Since the channel matrices $\boldsymbol{H}_{lb}$ and $\boldsymbol{h}_{ul}$ are independent and $\boldsymbol{V} = \boldsymbol{H}_{lb}\text{diag}(\boldsymbol{h}_{ul})$, we have $\mathbb{E}[\boldsymbol{V}] = \boldsymbol{0}$, which gives $\mathbb{E}[\boldsymbol{z}] = \boldsymbol{0}$. Let $\boldsymbol{H}_{lb}^i$ be the $i-$th column of $\boldsymbol{H}_{lb}$ and h_{ul}^i be the $i-$th element of $\boldsymbol{h}_{ul}$, then the i-th column of $\boldsymbol{V}$ is given by

$$\boldsymbol{v}_i = h_{ul}^i\boldsymbol{H}_{lb}^i. \tag{14}$$

We note that $\text{vec}(\boldsymbol{H}_{lb}) \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{S}_{lb}\otimes\boldsymbol{R}_{lb})$, and hence $\boldsymbol{H}_{lb}^i \sim \mathcal{CN}(\boldsymbol{0}, [\boldsymbol{S}_{lb}\otimes\boldsymbol{R}_{lb}]_{(i-1)M+1:iM,(i-1)M+1:iM})$. Since $\boldsymbol{h}_d$ and $\boldsymbol{v}_i$ are independent, we have $\mathbb{E}[\boldsymbol{h}_d\boldsymbol{v}_i^H] = \boldsymbol{0}$, $\forall\ i = 1, 2, \ldots, K$, and $\boldsymbol{C}_{zz}$ is a block diagonal matrix given by

$$\begin{aligned}\boldsymbol{C}_{zz} = \mathbb{E}\left[\boldsymbol{z}\boldsymbol{z}^H\right] &= \mathbb{E}\left\{\begin{bmatrix}\boldsymbol{h}_d\\ \boldsymbol{v}_1\\ \vdots\\ \boldsymbol{v}_K\end{bmatrix}\begin{bmatrix}\boldsymbol{h}_d^H & \boldsymbol{v}_1^H & \ldots & \boldsymbol{v}_K^H\end{bmatrix}\right\}\\ &= \begin{bmatrix}\boldsymbol{R}_{ub} & \boldsymbol{0}_{M\times M} & \ldots & \boldsymbol{0}_{M\times M}\\ \boldsymbol{0}_{M\times M} & \boldsymbol{R}_1 & \ldots & \boldsymbol{0}_{M\times M}\\ \vdots & \vdots & \ddots & \vdots\\ \boldsymbol{0}_{M\times M} & \boldsymbol{0}_{M\times M} & \ldots & \boldsymbol{R}_K\end{bmatrix}\end{aligned} \tag{15}$$

where $\boldsymbol{R}_i = [\boldsymbol{S}_{lb}\otimes\boldsymbol{R}_{lb}]_{(i-1)M+1:iM,(i-1)M+1:iM}$.

The error covariance matrix for the LMMSE channel estimate is given by [28, eq. (12.29)], [20, eq. (14)]

$$\boldsymbol{C}_{\hat{z}_{\text{lmmse}}} = \left(\boldsymbol{C}_{zz}^{-1} + \frac{\boldsymbol{G}^H\boldsymbol{G}}{\sigma^2}\right)^{-1}. \tag{16}$$

Further, using [20, eq. (13)-(14)] and (7), the error covariance matrix can be further written as

$$\boldsymbol{C}_{\hat{z}_{\text{lmmse}}} = \left(\boldsymbol{C}_{zz}^{-1} + \frac{\boldsymbol{\Phi}^H\boldsymbol{\Phi}\otimes\boldsymbol{I}_M}{\sigma^2}\right)^{-1}, \tag{17}$$

and the MSE of the LMMSE estimator is given by $\text{MSE}_{\hat{z}_{\text{lmmse}}} = \text{tr}(\boldsymbol{C}_{\hat{z}_{\text{lmmse}}})$.

Similar to the LS estimators introduced previously, the performance of the LMMSE estimator depends on the choice of phase-shift matrix $\boldsymbol{\Phi}$. The optimum phase-shift matrix can be obtained by solving the following optimization problem

$$\begin{aligned}\min_{\boldsymbol{\Phi}}\ & \text{tr}\left(\left(\boldsymbol{C}_{zz}^{-1} + \frac{\boldsymbol{\Phi}^H\boldsymbol{\Phi}\otimes\boldsymbol{I}_M}{\sigma^2}\right)^{-1}\right)\\ \text{s.t}\ & \ \boldsymbol{\Phi}\ \ \text{satisfies (8)},\\ & \left|\phi_{t,k}\right| = 1,\ t = 1, \ldots T_p,\ k = 1, \ldots, K.\end{aligned} \tag{18}$$

This optimization problem is non-convex due to the unit modulus constraints, making an analytical solution difficult. Similar problems also arise when designing optimal pilots for MIMO channel estimation [31]. In the following we will present an algorithm, employing a MM computational approach, to efficiently solve the optimization in (18).

A. MAJORIZATION-MINIMIZATION (MM) PHASE SHIFT OPTIMIZATION

MM refers to a class of algorithms that solve difficult optimization problems by solving a series of simpler problems which often admit closed-form solutions. MM successively forms a surrogate function or majorizer of the objective function at the current iterate, and then optimizes the surrogate function. These algorithms have been applied to approximately solve difficult non-convex optimization problems in signal processing [10], [32], [33]. Here we present an MM-based algorithm to solve the non-convex problem (18). Our approach is inspired from [33], which designed unit-modulus pilot signals for MIMO channel estimation.

We use the minimum pilot duration $T_p = K+1$ and rewrite the objective function of (18) as follows

$$\text{MSE}(\boldsymbol{\Phi}) = \text{tr}\left(\boldsymbol{C}_{\hat{z}_{\text{lmmse}}}\right) \tag{19}$$

where

$$\boldsymbol{C}_{\hat{z}_{\text{lmmse}}} = \left(\boldsymbol{C}_{zz}^{-1} + (\boldsymbol{\Phi}\otimes\boldsymbol{I}_M)^H\boldsymbol{W}^{-1}(\boldsymbol{\Phi}\otimes\boldsymbol{I}_M)\right)^{-1}, \tag{20}$$

with $\boldsymbol{W} = \sigma^2\boldsymbol{I}_{M(K+1)}$. Denoting $\tilde{\boldsymbol{\Phi}} = \boldsymbol{\Phi}\otimes\boldsymbol{I}_M$ and using [33, eq. (23)], $\boldsymbol{C}_{\hat{z}_{\text{lmmse}}}$ can be equivalently expressed as

$$\boldsymbol{C}_{\hat{z}_{\text{lmmse}}} = \boldsymbol{C}_{zz} - \boldsymbol{C}_{zz}\tilde{\boldsymbol{\Phi}}^H\left(\tilde{\boldsymbol{\Phi}}\boldsymbol{C}_{zz}\tilde{\boldsymbol{\Phi}}^H + \boldsymbol{W}\right)^{-1}\tilde{\boldsymbol{\Phi}}\boldsymbol{C}_{zz}. \tag{21}$$

Hence, the objective function becomes

$$\text{MSE}(\boldsymbol{\Phi}) = \text{tr}\Big(\boldsymbol{C}_{zz} - \boldsymbol{C}_{zz}\tilde{\boldsymbol{\Phi}}^H \boldsymbol{P}^{-1}\tilde{\boldsymbol{\Phi}}\boldsymbol{C}_{zz}\Big) \tag{22}$$

where $\boldsymbol{P} = \tilde{\boldsymbol{\Phi}}\boldsymbol{C}_{zz}\tilde{\boldsymbol{\Phi}}^H + \boldsymbol{W}$. The structure of this objective function is similar to the objective function considered in [33, eq. (40)], though there are some differences. While both objective functions depend on an auxiliary variable which is given by the Kronecker product of the optimization variable and an identity matrix (in our case $\tilde{\boldsymbol{\Phi}}$), the ordering of the matrices is reversed in our problem. In addition to this difference in the objective function, the structure of the optimization variable $\boldsymbol{\Phi}$ is different for our problem. This arises as a consequence of the direct channel between the transmitter and receiver, which requires that the first column of $\boldsymbol{\Phi}$ to be a vector of all ones. Moreover, our problem does not have a convolution structure for the optimization variable, which is the case in [33, eq. (4)].

Despite these differences, we may leverage some of the technical arguments from [33] to approximately solve the non-convex problem in (18). To be more specific, we recall that there are two main steps in a general MM algorithm. In the first step (majorization step), a surrogate function is constructed that can locally approximate the objective function. For this step, which involves matrix inequalities, we may closely follow the arguments in [33], due to the similarity of the objective functions upon introducing the auxiliary variable $\tilde{\boldsymbol{\Phi}}$. In the second step (minimization step), the surrogate function is minimized, and different technical developments are required. This is because a closed form solution should be obtained for the optimization variable $\boldsymbol{\Phi}$, which for our problem has a different structure and different constraints.

For the "majorization" step, the surrogate function g_{MSE} must satisfy the following

$$(i)\ g_{\text{MSE}}(\boldsymbol{\Phi}, \boldsymbol{\Phi}_t) \geq \text{MSE}(\boldsymbol{\Phi})\ \forall\ \boldsymbol{\Phi} \in \text{dom}(\boldsymbol{\Phi}) \tag{23}$$

$$(ii)\ g_{\text{MSE}}(\boldsymbol{\Phi}_t, \boldsymbol{\Phi}_t) = \text{MSE}(\boldsymbol{\Phi}_t) \tag{24}$$

where $\text{dom}(\boldsymbol{\Phi})$ contains all $\boldsymbol{\Phi}$ that satisfy (8) and $|\phi_{t,k}| = 1,\ t = 1, \ldots K+1,\ k = 1, \ldots, K$. Mirroring arguments from [33], and specifically applying [33, Lemma 2, 4], it can be shown that $g_{\text{MSE}}(\boldsymbol{\Phi}, \boldsymbol{\Phi}_t)$ satisfies

$$\begin{aligned} g_{\text{MSE}}(\boldsymbol{\Phi}, \boldsymbol{\Phi}_t) \leq\ & -2\text{Re}\Big\{\text{tr}\Big(\Big(\lambda_t\tilde{\boldsymbol{\Phi}}_t - \boldsymbol{A}_t\boldsymbol{A}_t^H\tilde{\boldsymbol{\Phi}}_t\boldsymbol{C}_{zz}\Big)^H\tilde{\boldsymbol{\Phi}}\Big)\Big\} \\ & - 2\text{Re}\Big\{\text{tr}\Big(\boldsymbol{C}_{zz}\boldsymbol{A}_t^H\tilde{\boldsymbol{\Phi}}\Big)\Big\} + \lambda_t\|\tilde{\boldsymbol{\Phi}}\|_F^2 + \text{const.} \end{aligned} \tag{25}$$

where $\tilde{\boldsymbol{\Phi}}_t = \boldsymbol{\Phi}_t \otimes \boldsymbol{I}_M$ and $\boldsymbol{A}_t = (\tilde{\boldsymbol{\Phi}}_t\boldsymbol{C}_{zz}\tilde{\boldsymbol{\Phi}}_t^H + \boldsymbol{W})^{-1}\tilde{\boldsymbol{\Phi}}_t\boldsymbol{C}_{zz}$. From the constraints in (18), we note that $\|\tilde{\boldsymbol{\Phi}}\|_F^2 = \text{const.}$ Thus, after constructing the surrogate function at the $t-$th iterate given by (25), the non-convex problem in (18) can be approximately solved by iteratively solving the following optimization problem

$$\begin{aligned} \min_{\boldsymbol{\Phi}}\ & -2\text{Re}\Big\{\text{tr}\Big(\boldsymbol{B}^H\tilde{\boldsymbol{\Phi}}\Big)\Big\} \\ \text{s.t}\ & \tilde{\boldsymbol{\Phi}} = \boldsymbol{\Phi} \otimes \boldsymbol{I}_M \\ & \boldsymbol{\Phi}\ \text{ satisfies (8)}, \\ & |\phi_{j,k}| = 1,\ j = 1, \ldots K+1,\ k = 1, \ldots, K \end{aligned} \tag{26}$$

where $\boldsymbol{B} = \lambda_t\tilde{\boldsymbol{\Phi}}_t - \boldsymbol{A}_t\boldsymbol{A}_t^H\tilde{\boldsymbol{\Phi}}_t\boldsymbol{C}_{zz} + \boldsymbol{A}_t\boldsymbol{C}_{zz}$, and $\lambda_t\boldsymbol{I} \succeq \boldsymbol{C}_{zz}^T \otimes (\boldsymbol{A}_t\boldsymbol{A}_t^H)$. The tightest upperbound of λ_t is given by $\lambda_t = \lambda_{\max}(\boldsymbol{C}_{zz}^T \otimes (\boldsymbol{A}_t\boldsymbol{A}_t^H))$. Since the computational cost of computing the largest eigenvalue can be large due to the large matrix dimensions, we use the upper bound [33, eq. (59)-(60)]

$$\lambda_{\max}\Big(\boldsymbol{C}_{zz}^T \otimes \Big(\boldsymbol{A}_t\boldsymbol{A}_t^H\Big)\Big) \leq \|\boldsymbol{C}_{zz}\|_1\Big\|\boldsymbol{A}_t\boldsymbol{A}_t^H\Big\|_1, \tag{27}$$

and use $\lambda_t = \|\boldsymbol{C}_{zz}\|_1\|\boldsymbol{A}_t\boldsymbol{A}_t^H\|_1$ [33, eq. (61)], where $\|\cdot\|_1$ denotes the maximum absolute column sum matrix norm [34].

Now we turn to the "minimization" step, for which we determine a closed form minimizer for the optimization problem in (26). Since $\tilde{\boldsymbol{\Phi}} = \boldsymbol{\Phi} \otimes \boldsymbol{I}_M$, it can be shown that

$$\text{tr}\Big(\boldsymbol{B}^H\tilde{\boldsymbol{\Phi}}\Big) = \text{tr}\left(\left(\sum_{i=1}^{M}\boldsymbol{B}_i\right)^H\boldsymbol{\Phi}\right) \tag{28}$$

where

$$\boldsymbol{B}_i = \boldsymbol{B}_{i:M:i+KM,i:M:i+KM} \tag{29}$$

denotes the submatrix of $\boldsymbol{B}$ extracted from the $K+1$ rows and columns of $\boldsymbol{B}$ with indices $[i, i+M, \ldots, i+MK]$. Denoting $\tilde{\boldsymbol{B}} = \sum_{i=1}^{M}\boldsymbol{B}_i$, the equivalent optimization problem to be solved at each iteration is given by

$$\begin{aligned} \min_{\boldsymbol{\Phi}}\ & -2\text{Re}\Big\{\text{tr}\Big(\tilde{\boldsymbol{B}}^H\boldsymbol{\Phi}\Big)\Big\} \\ \text{s.t}\ & \boldsymbol{\Phi}\ \text{ satisfies (8)}, \\ & |\phi_{j,k}| = 1,\ j = 1, \ldots K+1,\ k = 1, \ldots, K \end{aligned} \tag{30}$$

The MM update $\boldsymbol{\Phi}_{t+1}$ at each iteration can be obtained by solving the following equivalent optimization problem

$$\begin{aligned} \min_{\boldsymbol{\Phi}}\ & \Big\|\tilde{\boldsymbol{B}} - \boldsymbol{\Phi}\Big\|_F^2 \\ \text{s.t}\ & \boldsymbol{\Phi}\ \text{ satisfies (8)}, \\ & |\phi_{j,k}| = 1,\ j = 1, \ldots K+1,\ k = 1, \ldots, K \end{aligned} \tag{31}$$

which can be done by projecting $\tilde{\boldsymbol{B}}$ onto the constraint set as

$$\begin{aligned} \big[\boldsymbol{\Phi}_{t+1}\big]_{:,2:K+1} &= \exp\Big\{j\underline{\Big/\Big[\tilde{\boldsymbol{B}}\Big]_{:,2:K+1}}\Big\} \\ \big[\boldsymbol{\Phi}_{t+1}\big]_{:,1} &= \boldsymbol{1}_{K+1} \end{aligned} \tag{32}$$

where $[\boldsymbol{A}]_{:,i:j}$ denotes elements in all the rows from the i-th to the j-th column of $\boldsymbol{A}$. The overall MM based algorithm is summarized in Algorithm 1.

It has been shown that MM based algorithms converge to a stationary point for bounded objective functions [35]. Using similar arguments as in [33, Th. 1], it can be shown that the generated sequence of points $\boldsymbol{\Phi}_t$, $t = 0, 1, \ldots$ monotonically

Algorithm 1: MM-Based Algorithm for Solving (18)

Input: $\epsilon, \boldsymbol{W}, \boldsymbol{C}_{zz}$
Output: $\boldsymbol{\Phi}$

Set $t = 0$ and initialize $[\boldsymbol{\Phi}_0]_{i,j} = e^{j\theta_{i,j}}$ with $\theta_{i,j} \sim \text{Unif}[0, 2\pi]\ \forall\, i = 1, \ldots, K+1, j = 2, \ldots, K+1$ and $[\boldsymbol{\Phi}_0]_{:,1} = \mathbf{1}_{K+1}$. $\tilde{\boldsymbol{\Phi}}_0 = \boldsymbol{\Phi}_0 \otimes \boldsymbol{I}_M$

$\text{MSE}_0 = \text{tr}\left(\left(\boldsymbol{C}_{zz}^{-1} + \tilde{\boldsymbol{\Phi}}_0^H \boldsymbol{W}^{-1} \tilde{\boldsymbol{\Phi}}_0\right)^{-1}\right)$

repeat

  $\tilde{\boldsymbol{\Phi}}_t = \boldsymbol{\Phi}_t \otimes \boldsymbol{I}_M$

  $\boldsymbol{A}_t = \left(\tilde{\boldsymbol{\Phi}}_t \boldsymbol{C}_{zz} \tilde{\boldsymbol{\Phi}}_t^H + \boldsymbol{W}\right)^{-1} \tilde{\boldsymbol{\Phi}}_t \boldsymbol{C}_{zz}$

  $\lambda_t = \|\boldsymbol{C}_{zz}\|_1 \|\boldsymbol{A}_t \boldsymbol{A}_t^H\|_1$

  $\boldsymbol{B} = \lambda_t \tilde{\boldsymbol{\Phi}}_t - \boldsymbol{A}_t \boldsymbol{A}_t^H \tilde{\boldsymbol{\Phi}}_t \boldsymbol{C}_{zz} + \boldsymbol{A}_t \boldsymbol{C}_{zz}$

  $\tilde{\boldsymbol{B}} = \sum_{i=1}^{M} \boldsymbol{B}_{i:M:i+KM, i:M:i+KM}$

  $[\boldsymbol{\Phi}_{t+1}]_{:,2:K+1} = \exp\{j \angle [\tilde{\boldsymbol{B}}]_{:,2:K+1}\}$

  $[\boldsymbol{\Phi}_{t+1}]_{:,1} = \mathbf{1}_{K+1}$

  $\tilde{\boldsymbol{\Phi}}_{t+1} = \boldsymbol{\Phi}_{t+1} \otimes \boldsymbol{I}_M$

  $\text{MSE}_{t+1} = \text{tr}\left(\left(\boldsymbol{C}_{zz}^{-1} + \tilde{\boldsymbol{\Phi}}_{t+1}^H \boldsymbol{W}^{-1} \tilde{\boldsymbol{\Phi}}_{t+1}\right)^{-1}\right)$

  $t = t + 1$

until $\text{MSE}_{t-1} - \text{MSE}_t \leq \epsilon$

decreases the objective function $\text{MSE}(\boldsymbol{\Phi})$, and the algorithm converges to a stationary point.

In Fig. 2, we demonstrate the MSE performance of the LMMSE estimator with $\boldsymbol{\Phi}$ designed based on Algorithm 1. The plots show the evolution of the MSE of $\hat{\boldsymbol{z}}$ as the iterations of the MM algorithm increase. Trajectories are shown for 100 random initialization points. For comparison, the figure also shows the MSE obtained from the LS estimator, and the LMMSE channel estimator with $\boldsymbol{\Phi}$ chosen equal to the DFT matrix. It is observed that the phase-shift matrices initialized with random phases have poor MSE (even worse than the LS estimator). However, as the iterations of the MM algorithm increase, the MSE monotonically decreases for the different initialization points, and all solutions converge to a stationary point. Further, we observe that the DFT matrix is also a locally optimum solution that attains almost the same MSE as the solutions produced by Algorithm 1, with the added advantage of being independent of the noise variance and channel statistics. Our empirical study revealed that when the MM algorithm was initialized with the DFT matrix, this matrix did not change with successive iterations, suggesting that the DFT matrix is a stationary point. The box-plot shows the distribution of the MSE of the different local optima produced by Algorithm 1 with the 100 different random initialization points. We observe that the MSE of local optima are very close to each other and also to the DFT-based solution with the difference being of the order 10^{-3} dB. The same trend is observed for all four sets of system parameters considered. Overall, since the DFT-based design achieved nearly the same performance as the MM algorithm with random initialisation, and due to its relative simplicity, this choice appears well motivated for practical implementation.

B. MSE ANALYSIS FOR DFT-BASED DESIGN

Here we further analyse the performance of the LMMSE channel estimator, considering the DFT-based phase shift matrix. As we have shown, the DFT matrix achieves a local optima in terms of MSE performance that matches closely with the numerical MM-based solutions, despite being very simple (i.e., not requiring numerical iterations), while it is also more amenable to further analytical study.

With the DFT-based phase shift matrix, the MSE of the LMMSE channel estimator can be written as

$$\text{MSE}_{\hat{\boldsymbol{z}}_{\text{lmmse}}} = \text{tr}\left(\left(\boldsymbol{C}_{zz}^{-1} + \frac{T_p \boldsymbol{I}_{M(K+1)}}{\sigma^2}\right)^{-1}\right), \tag{33}$$

which can be equivalently expressed as

$$\text{MSE}_{\hat{\boldsymbol{z}}_{\text{lmmse}}} = \sum_{i=1}^{M} \left(\frac{1}{\frac{1}{\lambda_i(\boldsymbol{R}_{ub})} + \frac{T_p}{\sigma^2}} + \frac{K}{\frac{1}{\lambda_i(\boldsymbol{R}_{lb})} + \frac{T_p}{\sigma^2}}\right), \tag{34}$$

where $\lambda_i(\boldsymbol{R}_{ub}), \lambda_i(\boldsymbol{R}_{lb})$ are the eigenvalues of $\boldsymbol{R}_{ub}$ and $\boldsymbol{R}_{lb}$ respectively. From [36, Corollory 2.5], the function $f(\boldsymbol{x}) = \sum_{i=1}^{n} \frac{1}{\frac{1}{x_i}+c}$ is Schur-concave in $\boldsymbol{x} = [x_1, \ldots, x_n]$, since $\frac{1}{\frac{1}{x_i}+c}$ is a concave function in x_i. Thus, the MSE in (34) is Schur-concave in both $\boldsymbol{\lambda}(\boldsymbol{R}_{ub})$ and $\boldsymbol{\lambda}(\boldsymbol{R}_{lb})$. Further, due to the trace constraint on the covariance matrices, it follows that [36, Lemma 2.2],

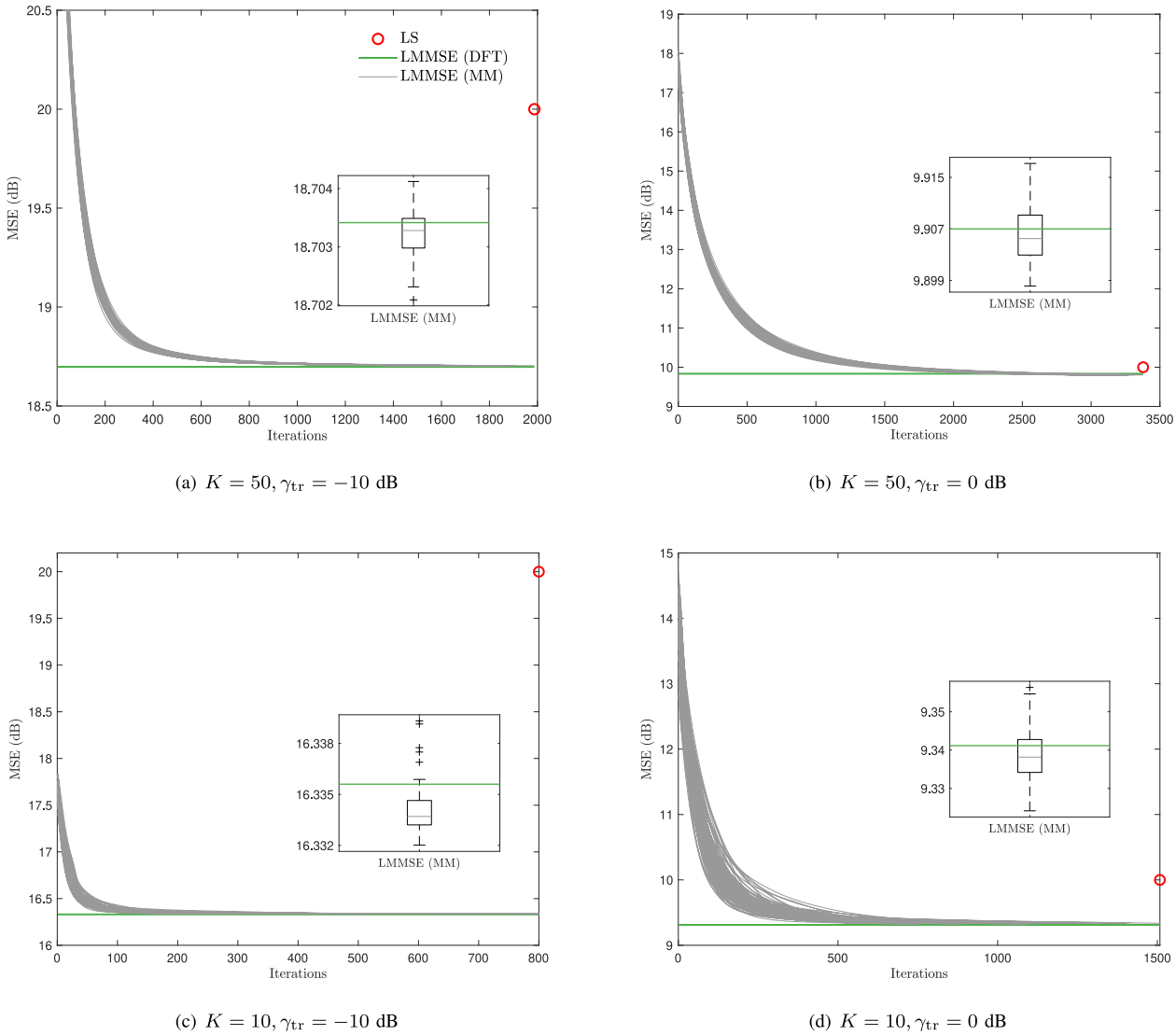
$$\boldsymbol{\lambda}(\boldsymbol{R}_{ub}) \succ [1, \ldots, 1], \text{ and } \boldsymbol{\lambda}(\boldsymbol{R}_{lb}) \succ [1, \ldots, 1]. \tag{35}$$

Thus, the MSE is maximized when $\boldsymbol{R}_{ub} = \boldsymbol{R}_{lb} = \boldsymbol{I}_M$ and the maximum MSE is given by

$$\text{MSE}^{\max}_{\hat{\boldsymbol{z}}_{\text{lmmse}}} = \frac{M(K+1)}{1 + \frac{T_p}{\sigma^2}}. \tag{36}$$

When comparing two covariance matrices $\boldsymbol{R}_x$ and $\boldsymbol{R}_y$, it is said that $\boldsymbol{R}_x$ is 'more correlated' than $\boldsymbol{R}_y$ if $\boldsymbol{\lambda}(\boldsymbol{R}_x) \succ \boldsymbol{\lambda}(\boldsymbol{R}_y)$ [37, Sec. 4.1.2]. Thus, as the degree of correlation of $\boldsymbol{R}_{ub}$ and $\boldsymbol{R}_{lb}$ increases, the MSE of the LMMSE channel estimate decreases. Note also that, even for the MSE maximizing scenario (36), it holds that $\text{MSE}^{\max}_{\hat{\boldsymbol{z}}_{\text{lmmse}}} < \text{MSE}_{\hat{\boldsymbol{z}}_{\text{ls}}}$, with $\text{MSE}_{\hat{\boldsymbol{z}}_{\text{ls}}}$ the MSE of the LS estimate, given by (12). Hence, the MSE of the LMMSE channel estimate is lower than that of the LS estimate for all SNR ranges, and also for all spatial correlation profiles.

We present a comparison of the MSE performance of the LMMSE and LS channel estimates in Fig. 3, considering the same simulation scenario as introduced in the previous subsection. Fig. 3 shows the theoretical and the simulated MSE performance of the LS and LMMSE estimate of the combined direct and cascaded channels $\boldsymbol{z}$ at an SNR of -10 dB, as a function of the spatial correlation coefficients. The theoretical MSE for the LS and LMMSE estimates are calculated using (12) and (34) respectively. It is observed that while the MSE of the LS estimate remains constant, regardless



(a) $K = 50, \gamma_{\text{tr}} = -10$ dB

(b) $K = 50, \gamma_{\text{tr}} = 0$ dB

(c) $K = 10, \gamma_{\text{tr}} = -10$ dB

(d) $K = 10, \gamma_{\text{tr}} = 0$ dB

FIGURE 2. The plots show the MSE of $\hat{z}$ obtained from the LMMSE channel estimator as the iterations of the MM algorithm increase, with $M = 10$, exponential spatial correlation matrices with $[\boldsymbol{R}_{ub}]_{i,j} = [\boldsymbol{R}_{lb}]_{i,j} = 0.6^{|i-j|}$, $[\boldsymbol{S}_{lb}]_{i,j} = 0.6^{|i-j|}$, and two different values of $K = 50, 10$ for 100 different random initialization points at two different $\gamma_{\text{tr}} = -10, 0$ dB. The MSE of the LS and the LMMSE channel estimator with Φ equal to the DFT matrix are also shown for comparison.

of the spatial correlation, the MSE of the LMMSE channel estimate decreases as the channels become more correlated, in line with the theoretical analysis presented above.

We may further analytically quantify the MSE improvement of the LMMSE estimate over the simple LS estimate at high and low SNR conditions. Using a Taylor series expansion at high SNR (i.e., for $\sigma^2 \to 0$), the MSE of the LMMSE estimate in (34) can be expressed as

$$\text{MSE}_{\hat{\boldsymbol{z}}_{\text{lmmse}}} = \text{MSE}_{\hat{\boldsymbol{z}}_{\text{ls}}} - \left(\frac{\sigma^2}{T_p}\right)^2 \times \left(\text{tr}\left(\boldsymbol{R}_{ub}^{-1}\right) + K\text{tr}\left(\boldsymbol{R}_{lb}^{-1}\right)\right) + o\left(\sigma^4\right). \quad (37)$$

This shows that, to leading order, the performance of the LMMSE estimator approaches that of the LS estimator as the SNR grows very large. However, for large but finite SNR, there is a performance improvement of the LMMSE approach determined by the trace of the inverse spatial covariance matrices. The effect of $\boldsymbol{R}_{lb}$ is relatively higher than that of $\boldsymbol{R}_{ub}$ on the MSE improvement, since $\boldsymbol{R}_{lb}$ appears in the LMMSE estimate for each of the K effective cascaded SIMO channels from the UE to the BS, via each RIS element.

Considering now the case of low SNR (i.e., for $\sigma^2 \to \infty$), the MSE in (34) admits

$$\text{MSE}_{\hat{\boldsymbol{z}}_{\text{lmmse}}} = M(K+1) - \frac{T_p}{\sigma^2}\left(\text{tr}\left(\boldsymbol{R}_{ub}^2\right) + K\text{tr}\left(\boldsymbol{R}_{lb}^2\right)\right) + o\left(\frac{1}{\sigma^2}\right). \quad (38)$$

Importantly, the MSE remains bounded as $\sigma^2 \to \infty$. This is in contrast to the LS estimate, as evident from (12), which shows that $\text{MSE}_{\hat{\boldsymbol{z}}_{\text{ls}}} \to \infty$ as $\sigma^2 \to \infty$. Hence, while the LMMSE estimator yields improved performance over LS

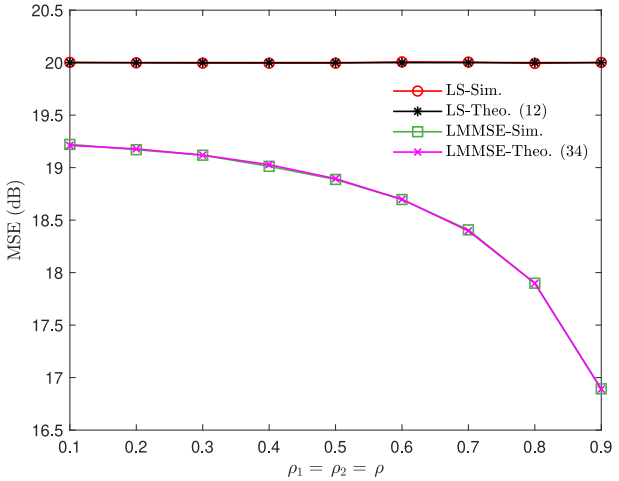


FIGURE 3. The plots compare the MSE of the LS and LMMSE channel estimate of the combined direct and cascaded channels z, as the correlation coefficients ρ_1, ρ_2 increase. Results shown for $M = 10$, $K = 50$ and $\gamma_{\text{tr}} = -10$ dB.

for all (finite) SNR values, the gain is significant in the low SNR regime.

The asymptotic expansions can be further simplified for the exponential spatial correlation model, for which $[\boldsymbol{R}_{ub}]_{i,j} = \rho_1^{|i-j|}$, $[\boldsymbol{R}_{lb}]_{i,j} = \rho_2^{|i-j|}$ with $0 < \rho_1, \rho_2 < 1$. Specifically, the MSE at high SNR (37) simplifies to

$$\text{MSE}_{\hat{\boldsymbol{z}}_{\text{lmmse}}} = \text{MSE}_{\hat{\boldsymbol{z}}_{\text{ls}}} - \left(\frac{\sigma^2}{T_p}\right)^2 \times \left[\frac{M + (M-2)\rho_1^2}{1-\rho_1^2} + \frac{K\big(M + (M-2)\rho_2^2\big)}{1-\rho_2^2}\right] + o\left(\sigma^4\right), \quad (39)$$

while the MSE at low SNR (38) simplifies to

$$\text{MSE}_{\hat{\boldsymbol{z}}_{\text{lmmse}}} = M(K+1) - \frac{T_p}{\sigma^2}\left[\frac{M\big(1-\rho_1^4\big) - 2\rho_1^2\big(1-\rho_1^{2M}\big)}{\big(1-\rho_1^2\big)^2} + \frac{K\big(M\big(1-\rho_2^4\big) - 2\rho_2^2\big(1-\rho_2^{2M}\big)\big)}{\big(1-\rho_2^2\big)^2}\right] + o\left(\frac{1}{\sigma^2}\right). \quad (40)$$

Here, the high SNR expression follows after applying the following property of the exponential correlation matrix [38, eq. (30)],

$$\text{tr}\left(\boldsymbol{R}^{-1}\right) = \frac{M + (M-2)\rho^2}{1-\rho^2}, \quad (41)$$

whereas the low SNR expression follows upon noting

$$\text{tr}\left(\boldsymbol{R}^2\right) = \|\boldsymbol{R}\|_F^2 = \sum_{i,j} \rho^{2|i-j|} = \frac{M\big(1-\rho^4\big) - 2\rho^2\big(1-\rho^{2M}\big)}{\big(1-\rho^2\big)^2} \quad (42)$$

for an $M \times M$ exponential correlation matrix $[\boldsymbol{R}]_{i,j} = \rho^{|i-j|}$.

An advantage of the expressions in (39) and (40) is that they quantify the MSE of the LMMSE estimate explicitly in terms of the correlation coefficients ρ_1, ρ_2 at high and low SNR respectively. From these expressions it is clear that the MSE of the LMMSE channel estimate decreases monotonically as the correlation coefficients ρ_1, ρ_2 increase (with the effect of ρ_2 being relatively larger than that of ρ_1). This is in line with the general conclusions made previously based on majorization arguments.

IV. DEEP LEARNING APPROACHES TO MMSE CHANNEL ESTIMATION

Despite yielding performance improvements over LS, the proposed LMMSE estimator is only optimal within the class of linear estimators. Beyond the class of linear estimators, the optimal MMSE estimator is difficult to determine analytically, due to the difficulty in evaluating the posterior mean in (9). To address this issue, we now propose data-driven non-linear channel estimators that approximate the optimal MMSE solution using deep neural networks.

In general, our approach is to introduce CNN-based estimators that take as input the sub-optimal LS estimate and produce as output an improved channel estimate that removes noise from the LS estimate, and approximates the optimal MMSE solution. Our approach is motivated by the universal function approximation property of CNNs, which states that a deep CNN can be used to approximate any continuous function to an arbitrary degree of accuracy when the depth of the network is large enough and given enough training samples [39].

We use the LS channel estimator with DFT-based phase-shift matrix $\boldsymbol{\Phi}$ as the input to our CNN. (One could also use the LMMSE estimate as input, however we choose the LS estimate due to its simplicity, and since it does not require knowledge of the noise variance nor the channel correlation matrices.) With this choice, it is convenient to reshape (11) in matrix form as

$$\hat{\boldsymbol{Z}}_{\text{ls}} = \boldsymbol{Z} + \boldsymbol{\zeta}, \quad (43)$$

where

$$\boldsymbol{Z} = [\boldsymbol{h}_d, \boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_K], \quad \hat{\boldsymbol{Z}}_{\text{ls}} = \left[\hat{\boldsymbol{h}_{d\text{ls}}}, \hat{\boldsymbol{v}_{1\text{ls}}}, \hat{\boldsymbol{v}_{2\text{ls}}}, \ldots, \hat{\boldsymbol{v}_{K\text{ls}}}\right] \quad (44)$$

and $\boldsymbol{\zeta}$ is the additive noise whose elements are independent and identically distributed (i.i.d) as $\mathcal{CN}(0, \frac{\sigma^2}{T_p})$ [20, eq. (31)]. In this form, the LS estimate $\hat{\boldsymbol{Z}}_{\text{ls}}$ may be seen as a noisy version of the unknown 'image' $\boldsymbol{Z}$.

CNN-based deep learning image denoising algorithms, which have been well studied by the image processing community, may be applied to learn the mapping from noisy images to clean images [25], [26], and therefore to filter out the image noise. Based on these ideas, we present two different approaches to learn the MMSE channel estimator for the RIS system. First, we present an approach based on the DnCNN architecture [25], which does not use knowledge

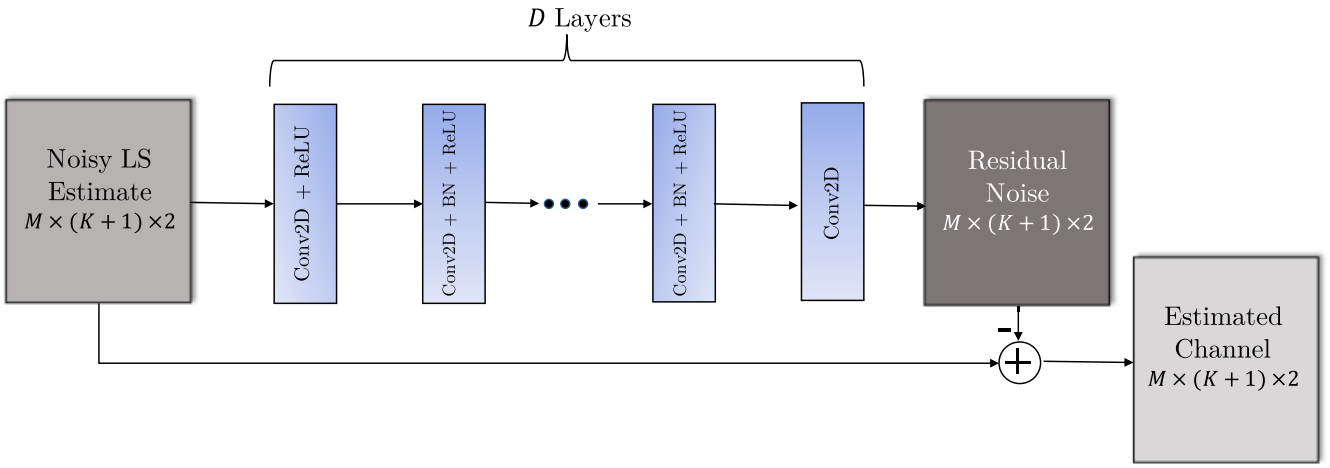


FIGURE 4. Network architecture of the DnCNN-based channel estimator.

of the noise variance. Then, we present an approach based on the FFDNet architecture [26], which utilizes the noise variance information to further improve channel estimation performance, particularly at low SNR. In terms of speed, FFDNet has a faster inference time, due to its lower model complexity when compared to DnCNN.

A. DNCNN NETWORK ARCHITECTURE

The input to the DnCNN is the LS channel estimate with dimension $M \times (K+1)$. Since deep learning architectures are naturally designed to deal with real-valued data, we reshape the LS estimate into a tensor of size $M \times (K+1) \times 2$, where the real and imaginary parts of $\hat{\mathbf{Z}}_{\text{ls}}$ are separated and placed as two separate channels, similar to the RGB channels in image data. The DnCNN network architecture is shown in Fig. 4. The input tensor is processed by a number of convolutional layers to produce the output. The depth of the network is D, with each layer using filters of size 3×3, along with zero padding of the input feature matrix such that the output contains multiple feature maps of the same dimension, i.e., $M \times (K+1)$. The number of feature maps N_f is fixed for all the layers except the last layer, which has 2 feature maps to match the dimension of the input tensor. The first layer performs a convolution operation (Conv2D), then applies the ReLU non-linear activation function. The middle $D-2$ layers perform Conv2D, apply batch normalization (BN), and then apply the ReLU activation function. BN helps to speed up the training process and also leads to better generalization performance of the network [40]. Finally, the last layer applies the Conv2D operation, but does not apply the non-linear activation function to avoid filtering out negative values.

With DnCNN, instead of directly learning the mapping from the noisy LS channel estimate to the noise-free channel matrix, we learn the noise map first, and then subtract it from the input noisy LS estimate to get the cleaned channel matrix. It has been shown (see [25]) that learning the noise map gives better image denoising performance across different signal-to-noise ratios (SNRs) compared with directly learning the denoised image. Moreover, the residual learning strategy where the input layer is directly connected to the output layer by a skip connection is easier to train, and provides better generalization performance on the test dataset [41].

The operation $\mathcal{F}$ of the DnCNN-based channel estimator can be described as

$$\hat{\mathbf{Z}}_D = \mathcal{F}\left(\hat{\mathbf{Z}}_{\text{ls}}; \boldsymbol{\Theta}\right) \tag{45}$$

where $\hat{\mathbf{Z}}_{\text{ls}}$ is the noisy LS channel matrix, $\hat{\mathbf{Z}}_D$ is the estimated channel matrix output by the DnCNN, and $\boldsymbol{\Theta}$ is the neural network parameters that are optimized during the initial training process. The DnCNN is trained by minimizing the MSE loss over the training set, comprising N_{tr} labelled data $\{\mathbf{Z}^i, \hat{\mathbf{Z}}^i_{\text{ls}}\}_{i=1}^{N_{\text{tr}}}$, generated from an ensemble of channel realizations. The MSE loss is defined as

$$\mathcal{L}(\boldsymbol{\Theta}) = \frac{1}{N_{\text{tr}}} \sum_{i=1}^{N_{\text{tr}}} \left\| \mathbf{Z}^i - \mathcal{F}\left(\hat{\mathbf{Z}}^i_{\text{ls}}; \boldsymbol{\Theta}\right) \right\|_F^2. \tag{46}$$

The training of DnCNN can be carried out offline by using simulated channel realizations. Once completed, the DnCNN may be employed at the BS for UL channel estimation.

B. FFDNET ARCHITECTURE

The network architecture of DnCNN does not utilize knowledge of the additive noise variance. The authors in [26] have proposed a CNN-based image denoising architecture, FFDNet, which provides better denoising performance by utilizing the noise variance information. We now present the architecture of FFDNet that utilizes the noise variance information to further improve the channel estimation performance.

Similar to the DnCNN, the input to the FFDNet is the noisy LS channel estimate of size $M \times (K+1) \times 2$, after

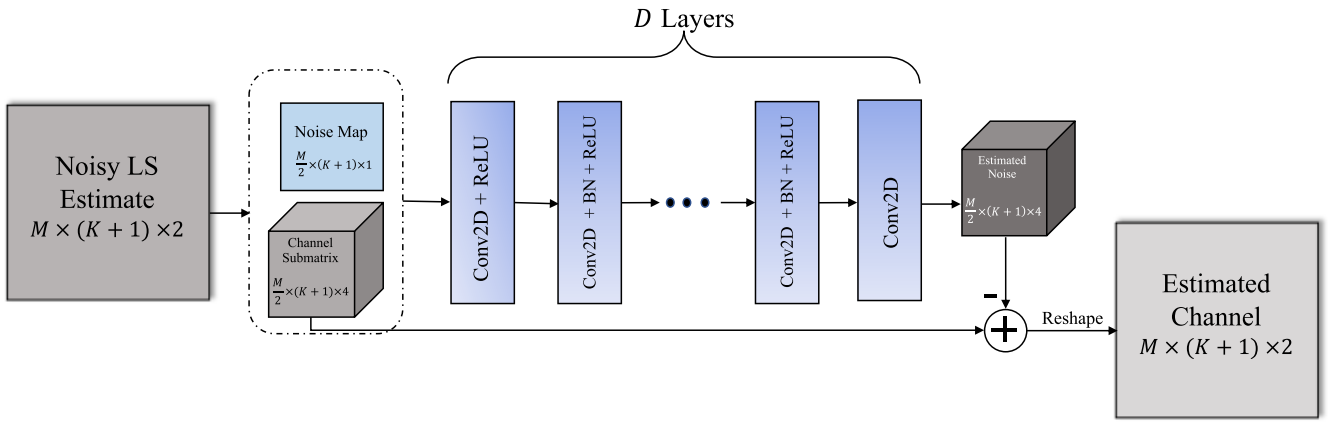


FIGURE 5. Network architecture of the FFDNet-based channel estimator.

separating the real and imaginary parts. In order to reduce the inference time and improve the efficiency of the channel estimator, the input tensor is further reshaped into a size of $\frac{M}{2} \times (K+1) \times 4$ [26]. Further, the noise map $\boldsymbol{N}$ of shape $\frac{M}{2} \times (K+1)$ having each entry set to $\frac{\sigma}{\sqrt{2T_p}}$ (the standard deviation of the additive noise in the LS estimate (43)) is concatenated with the reshaped LS channel matrix such that the input to the FFDNet-based channel estimator is a tensor of shape $\frac{M}{2} \times (K+1) \times 5$. This provides flexibility to the network to handle different noise levels. Fig. 5 shows the network architecture of the FFDNet-based channel estimator. Similar to DnCNN, the input tensor is processed by a number of convolutional layers. The network has a depth of D layers where the first layer uses the Conv2D + ReLU operation, the subsequent $D-1$ layers use the Conv2D + BN + ReLU operation, and finally the last layer uses only the Conv2D operation to output the estimated noise map. The filter size is fixed to 3×3. The number of feature maps in all the convolutional layers is fixed to N_f except the last layer which has 4 feature maps to match the shape of the input channel submatrices. All the convolutional layers use 'same' padding in the Conv2D operation so that the output dimension is fixed to $\frac{M}{2} \times (K+1) \times 4$. Similar to DnCNN, we use a residual learning strategy and learn the mapping of the residual noise from the noisy LS estimate. The residual noise is subtracted from the input to get the denoised channel submatrices, which are then reshaped to obtain the estimated channel output from the FFDNet. The FFDNet-based channel estimator is formulated as

$$\hat{\boldsymbol{Z}}_F = \mathcal{G}\left(\hat{\boldsymbol{Z}}_{\text{ls}}, \boldsymbol{N}; \boldsymbol{\Theta}\right) \tag{47}$$

where $\hat{\boldsymbol{Z}}_{\text{ls}}$ is the noisy input LS channel matrix, $\boldsymbol{N}$ is the noise map which depends on the variance of the additive noise, $\hat{\boldsymbol{Z}}_F$ is the estimated channel matrix output of FFDNet, and $\boldsymbol{\Theta}$ is the neural network parameter matrix which is optimized during the initial training process. Unlike the DnCNN (45), FFDNet can adapt to different noise levels through the noise map $\boldsymbol{N}$. The training dataset for FFDNet contains N_{tr} labelled data points $\{\boldsymbol{Z}^i, \boldsymbol{N}^i, \hat{\boldsymbol{Z}}^i_{\text{ls}}\}_{i=1}^{N_{\text{tr}}}$. Similar to DnCNN, FFDNet is also trained offline by minimizing the MSE loss function

$$\mathcal{L}(\boldsymbol{\Theta}) = \frac{1}{N_{\text{tr}}} \sum_{i=1}^{N_{\text{tr}}} \left\| \boldsymbol{Z}^i - \mathcal{G}\left(\hat{\boldsymbol{Z}}^i_{\text{ls}}, \boldsymbol{N}^i; \boldsymbol{\Theta}\right) \right\|_F^2 \tag{48}$$

using simulated channel realizations. After offline training, FFDNet can be employed at the BS for UL channel estimation.

V. PERFORMANCE RESULTS

A. CHANNEL ESTIMATION PERFORMANCE

In this section, we present the performance of the DnCNN and FFDNet-based channel estimators for RIS assisted MISO communications, and compare their performance with the LMMSE (with DFT phase-shift matrix) and LS estimators. We consider simulation scenarios with $M = 10$ BS antennas and $K = 50$ or $K = 10$ RIS elements. The pilot duration is set to $T_p = K + 1$. As for the previous figures, we use the exponential correlation model for the spatial correlation matrices,

$$[\boldsymbol{R}_{ub}]_{i,j} = \rho_1^{|i-j|}, [\boldsymbol{R}_{lb}]_{i,j} = \rho_2^{|i-j|}, [\boldsymbol{S}_{lb}]_{i,j} = \rho_3^{|i-j|} \tag{49}$$

where $0 < \rho_1, \rho_2, \rho_3 < 1$.

Training of the DnCNN is done at a constant SNR of $\gamma_{\text{tr}} = 0$ dB. The dataset for training FFDNet contains samples from 3 different SNR levels $\gamma_{\text{tr}} = -5$ dB, 0 dB and 5 dB. This changes the input noise map accordingly and should help FFDNet to learn the residual noise better than DnCNN. The training, validation and test datasets contain 16000, 8000, 6000 samples respectively. The network is trained using the well-known Adam algorithm [42] with the default parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, and with learning rate 0.001 and mini-batch size 100. The network is trained until the validation loss remains unchanged for 5 consecutive epochs.

A demonstration of the evolution of the training and validation losses of both estimators is presented in Fig. 6. We observe that convergence is fast for both estimators, with FFDNet having slightly faster convergence rate. As

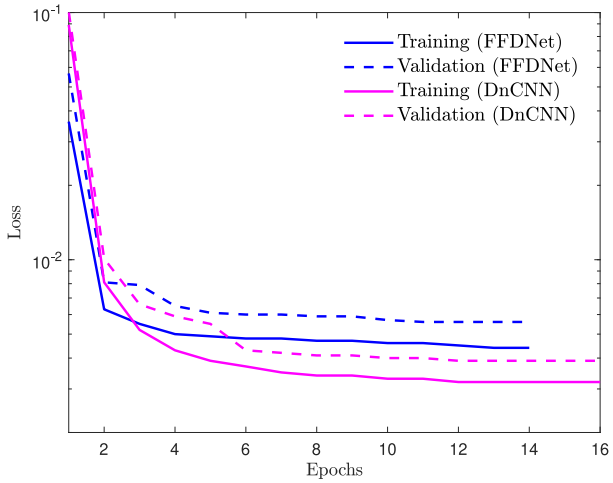


FIGURE 6. The plots show the training and the validation loss of DnCNN and FFDNet architectures as the training epochs proceed. Results are shown for $M = 10$, $K = 50$ and $\rho_1 = \rho_2 = \rho_3 = 0.6$.

expected, for both estimators, the validation loss is slightly higher than the training loss. Further, the training and validation losses are slightly higher for FFDNet compared with DnCNN. This should not be taken to imply inferior MSE performance on test datasets, however; such performance analysis will be investigated in the following. Similar trends to those observed in Fig. 6 were also observed for other parameter choices (results not shown for brevity).

The MSE performance comparison of the different estimators is shown in Fig. 7 for the direct channel $\boldsymbol{h}_d$ and the cascaded channel $\boldsymbol{V}$. For DnCNN and FFDNet, results are shown for $D = 8$ and $N_f = 4$. It is observed that the performance of DnCNN and FFDNet is superior to that of both linear estimators, LMMSE and LS. The superiority over LMMSE demonstrates the advantages of applying non-linear estimation, allowing the CNN-based estimators to capture the (non-Gaussian) distributional properties of the channel, whereas the LMMSE utilizes only the second order statistics. Note that while the direct channel $\boldsymbol{h}_d$ has a Gaussian distribution, the estimate of $\boldsymbol{h}_d$ obtained from the LMMSE estimate is not optimal since the received signal at the BS comprises a superposition of signals from the direct and the cascaded channels. Hence, while LMMSE yields the optimal MMSE solution for Gaussian channels, for the MISO configuration with RIS, using a non-linear channel estimator leads to a better MSE performance than the LMMSE estimator even for the direct channel. It is also observed that the MSE performance of DnCNN and FFDNet based channel estimators improve as the channels become more highly correlated, similar to the case of LMMSE, as shown previously. Among the deep learning based solutions, the FFDNet-based channel estimator performs slightly better than the DnCNN-based channel estimator at low SNR due to its ability to adapt to different SNR values. One potential disadvantage of FFDNet relative to DnCNN is that it requires knowledge of the noise variance, and hence cannot be used if a reliable estimate is unavailable at the BS.

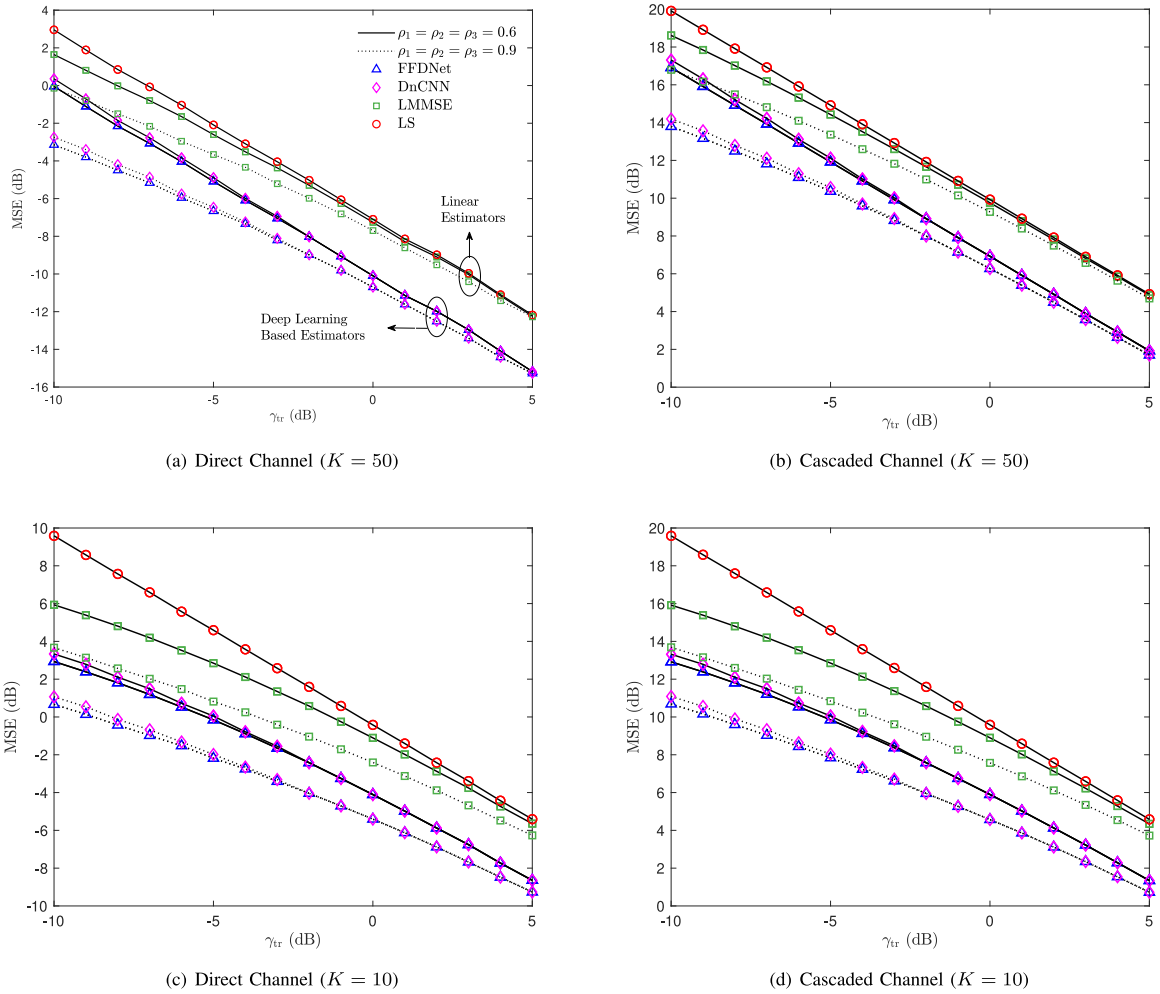
Next, consider the effect of varying K. For the direct channel $\boldsymbol{h}_d$, while the number of channel elements to be estimated does not depend on K, the pilot duration $T_p = K+1$ is longer for the case $K = 50$ compared with $K = 10$. This leads to a smaller MSE, as observed by comparing Fig. 7(a) and Fig. 7(c). For the LMMSE estimator, this trend is also evident from the theoretical MSE expression (34). Specifically, the MSE of the direct channel is captured by the first term in (34), which is decreasing in K when the pilot length is set to $T_p = K+1$. The same is true for the LS estimate, for which the MSE of the direct channel is given by [20, eq. (30)] $\frac{M\sigma^2}{K+1}$.

For the cascaded channel $\boldsymbol{V}$, from Figures 7(b) and 7(d), the opposite trend is observed, with the MSE for the $K = 50$ case generally being larger than that for the case $K = 10$, most notably at low training SNR. Unlike for the direct channel, for the cascaded channel the number of elements to estimate increases with K, making the effect of increasing K on the MSE performance less obvious. On the one hand, the increased pilot length for larger K reduces the estimation error per channel element, while on the other hand, the errors accumulate across a larger set of estimated channel coefficients. Our results suggest that the aggregation of channel estimation errors has a more significant effect, particularly in the low SNR regime. This is seen also from the theoretical MSE expression for the LMMSE estimate (34), where for low training SNR (i.e., large σ^2) the MSE of the cascaded channel, reflected by the second term in (34), is seen to monotonically and almost linearly increase with K. As the training SNR is increased, a monotonic increase of MSE in K is still observed from (34), albeit now at a slower rate. It is also noteworthy that the MSE for the LS estimator, given theoretically by $\sigma^2 M \frac{K}{K+1}$ [20, eq. (30)], becomes almost independent of K for $K \gg 1$, a trend that is also evident from Figures 7(b) and 7(d).

It is well-known that the performance of CNNs depend on the quality of the training data. Hence, it is important to study the robustness of the proposed CNN-based channel estimators for scenarios where the statistical features of the channels used in (online) testing differ from those used in (offline) training. To this end, we consider a "Mismatch" scenario where DnCNN and FFDNet are trained on a training dataset with $\rho_1 = \rho_2 = \rho_3 = 0.6$ and their performance is tested using a dataset with $\rho_1 = \rho_2 = \rho_3 = 0.9$. The MSE performance comparison is shown in Fig. 8. A "No-Mismatch" scenario is also presented for reference, for which the training and test datasets have the same channel statistics (with $\rho_1 = \rho_2 = \rho_3 = 0.9$). It is observed that both estimators are quite robust to statistical mismatches, while incurring a slight performance degradation of roughly 1 dB at low SNRs. Thus, such statistical mismatches are not expected to present a significant issue in practice.

B. COMPUTATIONAL COMPLEXITY

We now compare the computational complexity of the channel estimators. As a metric, we consider the number



(a) Direct Channel ($K = 50$)

(b) Cascaded Channel ($K = 50$)

(c) Direct Channel ($K = 10$)

(d) Cascaded Channel ($K = 10$)

FIGURE 7. The plots show the MSE performance of LS, LMMSE, DnCNN and FFDNet for the direct channel and the cascaded channel with $M = 10$ and two different values of $K = 10, 50$. The MSE performance is shown for two spatial correlation scenarios. The hyperparameters are chosen as $D = 8$, $N_f = 4$, for both DnCNN and FFDNet.

of floating point operations (FLOPs) required for channel estimation during online deployment.

For the LS channel estimator, which we recall is provided at the input of the CNN-based estimators, the filter $(\boldsymbol{G}^H\boldsymbol{G})^{-1}\boldsymbol{G}^H$ in (10) is independent of the channel realizations and depends only on the transmit pilot symbols from the UE and the phase shift matrix adopted at the RIS, which can be precomputed and stored at the BS. Thus, the main complexity involved with evaluating the LS estimate arises from matrix multiplications, and the number of FLOPS is given by [43]

$$C_{\text{ls}} = \mathcal{O}\Big(M^2(K+1)^2\Big). \tag{50}$$

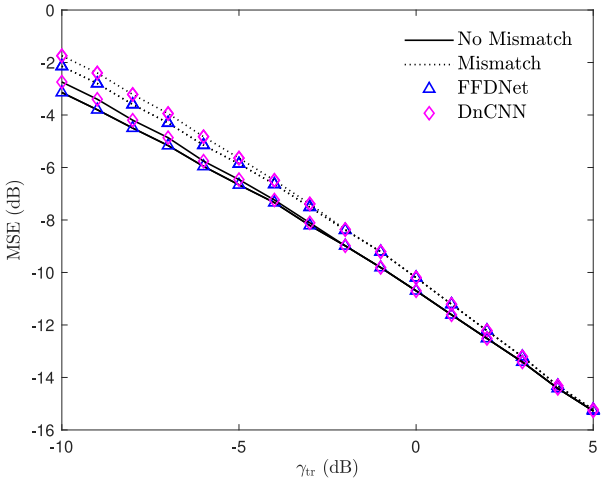
For the CNN-based estimators, the main computational complexity arises during the neutral network training phase. However, training the CNNs is carried out in an offline manner, and only needs to be updated at the rate of channel statistical changes (which typically occur over relatively long time scales). Moreover, our numerical results have shown that the CNN estimators are quite robust to statistical mismatches between the actual channels and those used for model training.

For a general CNN architecture, the computational complexity during online implementation is given by $C_{\text{CNN}} = \mathcal{O}(\sum_{l=1}^{L} h_l w_l F_l^2 N_{l-1} N_l)$ [44], where L is the total number of convolutional layers, h_l is the height and w_l is width of the output feature map of the $l-$th layer, F_l is the filter size in the $l-$the layer, and N_{l-1}, N_l is the number of input and output feature maps of the $l-$th layer respectively [44]. Based on this result, while accounting for the additional complexity involved with forming the initial LS estimate (50), the computational complexity of the DnCNN and FFDNet-based channel estimators can be computed as

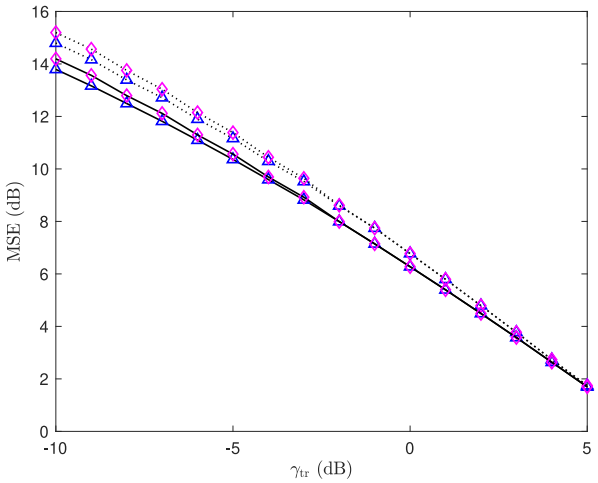
$$C_{\text{DnCNN}} = \mathcal{O}\Big(M^2(K+1)^2 + 9MN_f \times (K+1)\big(4 + (D-2)N_f\big)\Big) \tag{51}$$

and

$$C_{\text{FFDNet}} = \mathcal{O}\Big(M^2(K+1)^2 + 9MN_f(K+1) \times \big(4.5 + 0.5(D-2)N_f\big)\Big) \tag{52}$$



(a) Direct Channel



(b) Cascaded Channel

FIGURE 8. The plots show the MSE performance of DnCNN and FFDNet for the direct channel and the cascaded channel when there is a mismatch in the statistics of the training and test datasets. The 'Mismatch' scenario corresponds to a training dataset with $\rho_1 = \rho_2 = \rho_3 = 0.6$ and a test dataset with $\rho_1 = \rho_2 = \rho_3 = 0.9$. In the 'No-Mismatch' scenario both the training and test datasets have the same channel statistics with $\rho_1 = \rho_2 = \rho_3 = 0.9$. Results are show for $M = 10$, $K = 50$ and the hyperparameters are chosen as $D = 8$, $N_f = 4$, for both DnCNN and FFDNet.

respectively. Note that the complexity differences between these estimators arise from the smaller size of the feature maps in FFDNet, along with the presence of an additional noise map in the first layer of FFDNet. The height of the feature map is M in DnCNN, whereas it is $M/2$ in the FFDNet architecture.

A characteristic feature of FFDNet is that it employs knowledge of the noise variance as its input, in addition to the initial LS estimate. Implementation of the LMMSE estimate also utilizes knowledge of the noise variance (as well as channel covariance information). Assuming the LMMSE updates are performed in an online manner, the LMMSE implementation requires evaluation of the inverse of a $M(K+1) \times M(K+1)$ matrix in (13).

TABLE 1. Computational complexity of the LS, LMMSE, DnCNN-and FFDNet-based channel estimators in terms of the number of FLOPs for the two simulation scenarios considered in Fig. 7.

	LS	LMMSE	DnCNN	FFDNet
$K = 10$	2.4×10^4	1.355×10^6	1.34×10^5	8.934×10^4
$K = 50$	5.2×10^5	1.334×10^8	1.03×10^6	8.318×10^5

Therefore, the computational complexity of the LMMSE estimate is [43]

$$C_{\text{lmmse}} = \mathcal{O}\Big(M^3(K+1)^3\Big). \tag{53}$$

Due to the matrix inversion operation, the complexity of LMMSE is cubic in M, K, whereas it is only quadratic in M, K for LS and the CNN-based estimators as they require only matrix multiplication operations [43], [45].

The computational complexities of the channel estimators are compared in Table 1, which presents the exact number of FLOPs (using [46]), for each of the two simulation scenarios considered earlier. The complexities of the CNN-based estimators are sandwiched between those of the LS and LMMSE estimators, with LS having the lowest complexity, and LMMSE having the highest. Among the deep learning-based estimators, FFDNet has a lower complexity, since its architecture employs intermediate feature maps with reduced dimensionality. The observed trends are consistent with previous results presented for LMMSE, LS and CNN-based channel estimators in the context of mmWave systems [43], [45].

In interpreting the effect of changing K, we add the caveat that so far we have focused only on the MSE performance. The fact that increasing K also leads to an increase in the pilot length can reduce spectral efficiency, by allowing less time for data transmission. This system-level issue will be analyzed in the following subsection. Prior to conducting this analysis however, we explore the effect of the different hyperparameters of the neural network-based estimators, in terms of performance and complexity.

There are two main hyperparameters for the DnCNN and FFDNet-based channel estimators, D and N_f. Table 2 shows the MSE of the direct and the cascaded channel for two different feature maps $N_f = 4, 8$ and for different depths of the network $D = 4, 6, 8, 10$, at a fixed training SNR of $\gamma_{\text{tr}} = 5$ dB with $M = 10, K = 50$. It also shows the average CPU runtime (in ms) of DnCNN and FFDNet for inference on a test data set of size 1000. It is observed that the MSE performance is not very sensitive to the choice of N_f and D. Also, the difference in runtimes are quite modest, with an increase in average runtime observed as N_f and D increase, due to the higher model complexity. Generally, the setting $N_f = 4$ and $D = 8$ provides a good trade-off between MSE performance and average runtime for both FFDNet and DnCNN. Further note that the average runtime of FFDNet is lower than that of DnCNN, in accordance with the computational complexity analysis present above and also with the original FFDNet-based image denoising paper [26].

TABLE 2. Performance of the DnCNN-and FFDNet-based channel estimators for different choices of hyperparameters, N_f and D for $M = 10$, $K = 50$ at $\gamma_{\text{tr}} = 5$ dB.

	N_f	4				8			
	D	4	6	8	10	4	6	8	10
FFDNet	Direct Channel MSE (dB)	-14.44	-14.77	-15.07	-15.01	-14.54	-14.53	-14.85	-14.96
	Cascaded Channel MSE (dB)	2.47	2.15	1.91	1.97	3.39	2.38	2.08	2.2
	Run Time (ms)	115.7	122.6	130.5	140.5	118.7	127.6	142.6	156.6
DnCNN	Direct Channel MSE (dB)	-14.28	-14.5	-15.05	-15.08	-13.60	-14.22	-14.76	-14.82
	Cascaded Channel MSE (dB)	2.51	2.25	1.98	2.15	3.52	2.45	2.15	2.26
	Run Time (ms)	171.5	192.4	217.4	232.2	187.6	218.9	232.9	257.3

C. SYSTEM-LEVEL PERFORMANCE: DOWNLINK DATA TRANSMISSION

Next we demonstrate the system level performance gains in terms of achievable rate when the proposed channel estimates are employed for DL data transmission. During the data transmission phase, the received signal at the UE is given by [18, eq. (3)]

$$y_u = \sqrt{P_{tx}}\left(\boldsymbol{h}_d^T + \boldsymbol{\phi}_d^T \boldsymbol{V}^T\right)\boldsymbol{w} x_d + n_d \tag{54}$$

where $\boldsymbol{\phi}_d = [e^{j\theta_1}, \ldots, e^{j\theta_K}]^T \in \mathbb{C}^K$ is the phase-shift vector of the RIS during the data transmission phase where, $0 \leq \theta_k \leq 2\pi,\ k = 1, \ldots, K$ are the phase shifts introduced by the K RIS elements, $\boldsymbol{w} \in \mathbb{C}^M$ is the beamforming vector at the BS, $x_d \in \mathbb{C}$ is the information bearing signal with $\mathbb{E}[|x_d|^2] = 1$, $\sqrt{P_{tx}}$ is the transmit power and $n_d \sim \mathcal{CN}(0, \sigma_d^2)$ is the AWGN at the UE. The phase-shift vector and the beamforming vector are selected based on [18], which proposed a low complexity design that maximizes the received SNR at the UE,

$$\gamma_r = \frac{P_{tx}}{\sigma_d^2}\left|\left(\boldsymbol{h}_d^T + \boldsymbol{\phi}_d^T \boldsymbol{V}^T\right)\boldsymbol{w}\right|^2. \tag{55}$$

Let $\hat{\boldsymbol{h}}_d$ and $\hat{\boldsymbol{V}}$ be the estimates of the UL channels $\boldsymbol{h}_d$ and $\boldsymbol{V}$ respectively. Then, using these as plug-in estimates for the optimal $\boldsymbol{w}$ and $\boldsymbol{\phi}_d$ that maximize (55) [18] leads to the beamformers

$$\boldsymbol{\phi}_{d_{\text{opt}}} = \exp\left\{-j\angle\hat{\boldsymbol{V}}^T\hat{\boldsymbol{h}}_d^*\right\}, \tag{56a}$$

$$\boldsymbol{w}_{\text{opt}} = \frac{\hat{\boldsymbol{h}}_d^* + \hat{\boldsymbol{V}}^*\boldsymbol{\phi}_{d_{\text{opt}}}^*}{\left\|\hat{\boldsymbol{h}}_d^* + \hat{\boldsymbol{V}}^*\boldsymbol{\phi}_{d_{\text{opt}}}^*\right\|}. \tag{56b}$$

With these choices, the achievable rate at the UE is given by

$$R = \left(1 - \frac{T_p}{T_c}\right)\log_2\left(1 + \bar{\gamma}\left|\left(\boldsymbol{h}_d^T + \boldsymbol{\phi}_{d_{\text{opt}}}^T \boldsymbol{V}^T\right)\boldsymbol{w}_{\text{opt}}\right|^2\right) \tag{57}$$

where $\bar{\gamma} = P_{tx}/\sigma_d^2$ is the average transmit SNR, T_c is the channel coherence time, and we recall that T_p is the pilot duration.

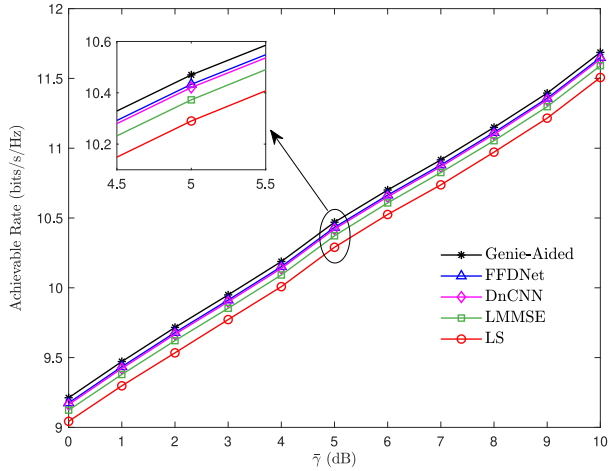
We study the achievable rate at the UE for the different channel estimation schemes. We adopt the same system parameters as before, though here we assume a training SNR of $\gamma_{\text{tr}} = -10$ dB, and consider a channel coherence time $T_c = 196$. The achievable rates (57), with the beamforming vectors (56a) and (56b) computed based on each of the channel estimation schemes, are shown in Figs. 9(a) and 9(b) as a function of the transmit SNR $\bar{\gamma}$. As before, the downlink data transmission performance of LMMSE channel estimator is shown only for the DFT-based design.

We see that the achievable rate is maximum for the beamforming vectors designed using the DnCNN and FFDNet-based channel estimators, with their performance approaching to that of the beamforming vectors constructed based on perfect knowledge of the channels (i.e., the 'Genie-Aided' channel estimator). The performance improvement of the DnCNN and FFDNet based estimators over the linear estimators is most evident for smaller numbers of RIS elements, K.

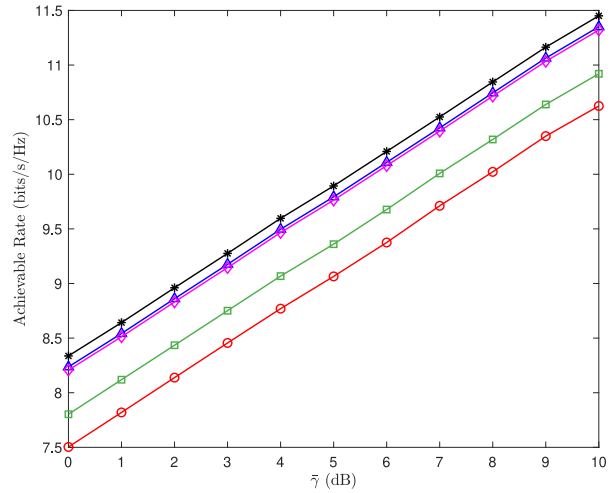
Next, we more closely study the effect of increasing K on the achievable rate of the system. As K increases, the received SNR scales with K^2 [16], [47], but at the same time there is a penalty in the pre-log factor of (57) due to the channel estimation overhead, since the pilot duration $T_p = K + 1$ increases with K. Fig. 10 shows the achievable rate as the number of RIS elements K changes, with fixed $\bar{\gamma} = 0$ dB and $\gamma_{\text{tr}} = -10$ dB. It is observed that first the achievable rate increases with K due to the increasing beamforming gain from the RIS elements (proportional to K^2); however, after reaching a maximum value, the achievable rate starts decreasing. This is because, at large values of K, the channel estimation overhead penalty in the pre-log factor of (57) dominates the beamforming gain. The above observation is consistent with a result reported in [48], where the authors demonstrated a trade-off between achievable rate and the number of RIS elements for a RIS-aided single-antenna system, while similarly accounting for channel estimation overhead.

VI. DISCUSSION

In this paper, we have proposed channel estimators for RIS -assisted MISO communications based on the MMSE criterion. First, we derived the best linear estimator, the LMMSE, that admits a closed-form representation depending on the second order statistics of the channels and the noise. The performance of the LMMSE filter depends on the choice of the phase shift matrix employed of the RIS, and hence, we proposed an MM-based algorithm to optimize these phase shifts to yield the smallest MSE. The MM-based algorithm was shown to have good convergence properties, and was used to benchmark the performance of the simple DFT-based phase shifts, which have been shown previously to be optimal for the simple LS-based channel estimators. We demonstrated that the DFT-based phase shifts constituted a local



(a) $K = 50$



(b) $K = 10$

FIGURE 9. The plots show the achievable rate at the UE versus the average transmit SNR, $\bar{\gamma}$. Results shown for $M = 10$, and two different values of $K = 10, 50$, using different channel estimation schemes for designing the beamforming vectors. The 'Genie-Aided' curve corresponds to the case where the channels are known perfectly.

optimal in the space of feasible phase shift matrices, and produced very similar performance to that achievable with other local optimal attained using the computational MM approach. This, in turn, provides practical motivation for using the DFT-based phase shift matrix with the LMMSE filter.

Due to the non-Gaussianity of the effective channels in the MISO RIS-aided system, the LMMSE channel estimator is not the globally optimal MMSE solution. While the globally optimal solution appears analytically intractable, we have proposed two CNN-based image denoising machine learning architectures to approximate this solution. The performance gains of the CNN-based channel estimators over the linear estimators (i.e., LMMSE as well as simple LS) were shown to be quite substantial. The computational cost is also low, particularly since the neural networks can be trained in an offline manner.

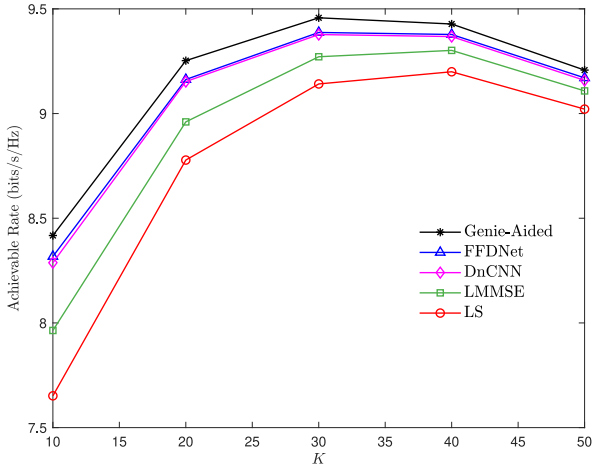


FIGURE 10. The plots show the achievable rate at the UE versus the number of RIS elements K, using different channel estimation schemes for designing the beamforming vectors. Results shown for $M = 10$ and $\bar{\gamma} = 0$ dB. The 'Genie-Aided' curve corresponds to the case where the channels are known perfectly.

Generally speaking, the problem that we address is well suited to employing a computational neural network solution, since it does not admit a tractable analytical form and the solution is non-linear. Our numerical comparisons between the neural network-based solutions and the LMMSE solution demonstrates that the non-linearity captured by the CNN-based estimators can substantially improve performance, and suggests that such non-linear schemes may be preferable in the context of RIS-aided wireless communication systems. Our results, in general, contribute to the increasing body of work focused on applying deep learning techniques to solve complex optimization problems in wireless communication systems (e.g., [41], [45], [49]–[55]).

The channel estimates produced by the estimators that we propose can be employed in the implementation of passive and active beamforming vectors (see Section V-C). Some other recent works have proposed alternative data-driven techniques for constructing such beamformers [56]–[59]. One such approach [56] proposed a deep reinforcement learning-based method for active and passive beamforming design, assuming perfect knowledge of the channel matrices, while other approaches [57], [58] have proposed to employ a fully-connected deep neural network (FNN) to directly learn the beamformers from the received pilot signals, without explicitly estimating the channels. In another approach [59], the authors proposed a codebook-based passive beamforming design, where a deep neural network is used to learn the mapping from the received pilot signals to the optimum beamformers from a predefined set of candidate beamformers. While having noteworthy connections to our work, the problem being addressed is different, and the methods proposed were not designed to produce estimates of the direct and cascaded channels.

For the data-driven channel estimators that we propose, we chose CNN-based architectures in favor of other data-driven techniques like FNN [57], [58]. This choice was made

for a few different reasons. A primary reason is that CNNs involve relatively few parameters, particularly with respect to FNN architectures, which leads to a reduction in implementation complexity. The CNN architectures are also quite well suited to the problem of interest as they use filters to learn the local spatial correlations of the input data. Similar to the case of image data, in which the nearby pixels are correlated, for the RIS channel models that we consider, the adjacent channel coefficients are (spatially) correlated. It is of course possible that other data-driven techniques, like FNN, could potentially be used to propose alternative channel estimation strategies; however, we expect any performance gains with respect to CNN-based methods to be minimal. This is implicitly evident from our numerical results, which have shown that the downlink achievable rate performance of beamformers obtained from the CNN-based channel estimates is close to that achieved with perfect channel knowledge.

Overall, our work has demonstrated the applicability of data-driven CNN-based optimal channel estimators for a single user MISO system aided by a single RIS. Extensions of our model, such as accounting for multiple RIS, multiple antenna users, and multiple user scenarios, present meaningful directions to explore in future investigations.

ACKNOWLEDGMENT

The authors thank Daniel Palomar for discussions about MM optimization that was helpful for developing the numerical algorithm in Section III-A.

REFERENCES

[1] N. K. Kundu and M. R. McKay, "A deep learning-based channel estimation approach for MISO communications with large intelligent surfaces," in *Proc. IEEE 31st Annu. Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2020, pp. 1–6.

[2] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, "5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view," *IEEE Access*, vol. 6, pp. 55765–55779, 2018.

[3] J. Zhang, E. Björnson, M. Matthaiou, D. W. K. Ng, H. Yang, and D. J. Love, "Prospective multiple antenna technologies for beyond 5G," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 8, pp. 1637–1660, Aug. 2020.

[4] L. Zhu, Z. Xiao, X.-G. Xia, and D. O. Wu, "Millimeter-wave communications with non-orthogonal multiple access for B5G/6G," *IEEE Access*, vol. 7, pp. 116123–116132, 2019.

[5] N. K. Kundu, R. K. Mallik, and M. R. McKay, "Signal design for frequency-phase keying," *IEEE Trans. Wireless Commun.*, vol. 19, no. 6, pp. 4067–4079, Jun. 2020.

[6] E. Basar, "Reconfigurable intelligent surface-based index modulation: A new beyond MIMO paradigm for 6G," *IEEE Trans. Commun.*, vol. 68, no. 5, pp. 3187–3196, May 2020.

[7] M. Di Renzo *et al.*, "Smart radio environments empowered by reconfigurable AI meta-surfaces: An idea whose time has come," *EURASIP J. Wireless Commun. Netw.*, vol. 2019, no. 1, pp. 1–20, May 2019.

[8] M. Di Renzo *et al.*, "Smart radio environments empowered by reconfigurable intelligent surfaces: How it works, state of research, and the road ahead," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 11, pp. 2450–2525, Nov. 2020.

[9] C. Huang *et al.*, "Holographic MIMO surfaces for 6G wireless networks: Opportunities, challenges, and trends," *IEEE Wireless Commun.*, vol. 27, no. 5, pp. 118–125, Oct. 2020.

[10] C. Huang, A. Zappone, G. C. Alexandropoulos, M. Debbah, and C. Yuen, "Reconfigurable intelligent surfaces for energy efficiency in wireless communication," *IEEE Trans. Wireless Commun.*, vol. 18, no. 8, pp. 4157–4170, Aug. 2019.

[11] C. Huang, A. Zappone, M. Debbah, and C. Yuen, "Achievable rate maximization by passive intelligent mirrors," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 3714–3718.

[12] Q. Wu and R. Zhang, "Intelligent reflecting surface enhanced wireless network: Joint active and passive beamforming design," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 1–6.

[13] Q. Wu and R. Zhang, "Intelligent reflecting surface enhanced wireless network via joint active and passive beamforming," *IEEE Trans. Wireless Commun.*, vol. 18, no. 11, pp. 5394–5409, Nov. 2019.

[14] Q. Nadeem, A. Kammoun, A. Chaaban, M. Debbah, and M.-S. Alouini, "Asymptotic max-min SINR analysis of reconfigurable intelligent surface assisted MISO systems," *IEEE Trans. Wireless Commun.*, vol. 19, no. 12, pp. 7748–7764, Dec. 2020.

[15] P. Wang, J. Fang, X. Yuan, Z. Chen, and H. Li, "Intelligent reflecting surface-assisted millimeter wave communications: Joint active and passive precoding design," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 14960–14973, Dec. 2020.

[16] S. Zhou, W. Xu, K. Wang, M. Di Renzo, and M.-S. Alouini, "Spectral and energy efficiency of IRS-assisted MISO communication with hardware impairments," *IEEE Wireless Commun. Lett.*, vol. 9, no. 9, pp. 1366–1369, Sep. 2020.

[17] N. K. Kundu and M. R. McKay, "RIS-assisted MISO communication: Optimal beamformers and performance analysis," Jul. 2020, [Online]. Available: https://arxiv.org/abs/2007.08309

[18] D. Mishra and H. Johansson, "Channel estimation and low-complexity beamforming design for passive intelligent surface assisted MISO wireless energy transfer," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, May 2019, pp. 4659–4663.

[19] Q.-U.-A. Nadeem, A. Kammoun, A. Chaaban, M. Debbah, and M.-S. Alouini, "Intelligent reflecting surface assisted multi-user MISO communication," Jun. 2019. [Online]. Available: https://arxiv.org/abs/1906.02360

[20] T. L. Jensen and E. De Carvalho, "An optimal channel estimation scheme for intelligent reflecting surfaces based on a minimum variance unbiased estimator," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, May 2020, pp. 5000–5004.

[21] B. Zheng and R. Zhang, "Intelligent reflecting surface-enhanced OFDM: Channel estimation and reflection optimization," *IEEE Wireless Commun. Lett.*, vol. 9, no. 4, pp. 518–522, Apr. 2020.

[22] C. You, B. Zheng, and R. Zhang, "Intelligent reflecting surface with discrete phase shifts: Channel estimation and passive beamforming," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2020, pp. 1–6.

[23] L. Wei, C. Huang, G. C. Alexandropoulos, and C. Yuen, "Parallel factor decomposition channel estimation in RIS-assisted multi-user MISO communication," in *Proc. IEEE 11th Sens. Array Multichannel Signal Process. Workshop (SAM)*, Jun. 2020, pp. 1–5.

[24] Q.-U.-A. Nadeem, H. Alwazani, A. Kammoun, A. Chaaban, M. Debbah, and M.-S. Alouini, "Intelligent reflecting surface-assisted multi-user MISO communication: Channel estimation and beamforming design," *IEEE Open J. Commun. Soc.*, vol. 1, pp. 661–680, May 2020.

[25] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.

[26] K. Zhang, W. Zuo, and L. Zhang, "FFDNet: Toward a fast and flexible solution for CNN-based image denoising," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4608–4622, Sep. 2018.

[27] D. Gesbert, H. Bolcskei, D. A. Gore, and A. J. Paulraj, "Outdoor MIMO wireless channels: Models and performance prediction," *IEEE Trans. Commun.*, vol. 50, no. 12, pp. 1926–1934, Dec. 2002.

[28] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, 1st ed. Englewood Cliffs, NJ, USA: Prentice Hall PTR, 1993.

[29] Y. Yang, B. Zheng, S. Zhang, and R. Zhang, "Intelligent reflecting surface meets OFDM: Protocol design and rate maximization," *IEEE Trans. Commun.*, vol. 68, no. 7, pp. 4522–4535, Jul. 2020.

[30] M. Biguesh and A. B. Gershman, "Training-based MIMO channel estimation: A study of estimator tradeoffs and optimal training signals," *IEEE Trans. Signal Process.*, vol. 54, no. 3, pp. 884–893, Mar. 2006.

[31] J. H. Kotecha and A. M. Sayeed, "Transmit signal design for optimal estimation of correlated MIMO channels," *IEEE Trans. Signal Process.*, vol. 52, no. 2, pp. 546–557, Feb. 2004.

[32] Y. Sun, P. Babu, and D. P. Palomar, "Majorization-minimization algorithms in signal processing, communications, and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 3, pp. 794–816, Feb. 2017.
[33] Z. Wang, P. Babu, and D. P. Palomar, "Design of PAR-constrained sequences for MIMO channel estimation via majorization–minimization," *IEEE Trans. Signal Process.*, vol. 64, no. 23, pp. 6132–6144, Dec. 2016.
[34] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2012.
[35] J. Song, P. Babu, and D. P. Palomar, "Optimization methods for designing sequences with low autocorrelation sidelobes," *IEEE Trans. Signal Process.*, vol. 63, no. 15, pp. 3998–4009, Aug. 2015.
[36] D. P. Palomar and Y. Jiang, *MIMO Transceiver Design via Majorization Theory*. Boston, MA, USA: Now Publ., Inc., 2007.
[37] E. Jorswieck and B. Holger, *Majorization and Matrix-Monotone Functions in Wireless Communications*. Boston, MA, USA: Now Publ., Inc., 2007.
[38] R. K. Mallik, "The exponential correlation matrix: Eigen-analysis and applications," *IEEE Trans. Wireless Commun.*, vol. 17, no. 7, pp. 4690–4705, Jul. 2018.
[39] D.-X. Zhou, "Universality of deep convolutional neural networks," *Appl. Comput. Harmon. Anal.*, vol. 48, no. 2, pp. 787–794, Mar. 2020.
[40] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Jul. 2015, pp. 448–456.
[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[42] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, May 2015, pp. 1–15.
[43] P. Dong, H. Zhang, G. Y. Li, N. NaderiAlizadeh, and I. S. Gaspar, "Deep CNN for wideband mmWave massive MIMO channel estimation using frequency correlation," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, May 2019, pp. 4529–4533.
[44] K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5353–5360.
[45] P. Dong, H. Zhang, G. Y. Li, I. S. Gaspar, and N. NaderiAlizadeh, "Deep CNN-based channel estimation for mmWave massive MIMO systems," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 5, pp. 989–1000, Sep. 2019.
[46] R. Hunger, "Floating point operations in matrix-vector calculus," Inst. Circuit Theory Signal Process., Munich Univ. Technol., München, Germany, Rep. 2007, 2005.
[47] E. Basar, M. Di Renzo, J. De Rosny, M. Debbah, M.-S. Alouini, and R. Zhang, "Wireless communications through reconfigurable intelligent surfaces," *IEEE Access*, vol. 7, pp. 116753–116773, 2019.
[48] N. K. Kundu and M. R. McKay, "Large intelligent surfaces with channel estimation overhead: Achievable rate and optimal configuration," *IEEE Wireless Commun. Lett.*, early access, Jan. 22, 2021, doi: 10.1109/LWC.2021.3053593.
[49] H. He, S. Jin, C.-K. Wen, F. Gao, G. Y. Li, and Z. Xu, "Model-driven deep learning for physical layer communications," *IEEE Wireless Commun.*, vol. 26, no. 5, pp. 77–83, Oct. 2019.
[50] Z. Qin, H. Ye, G. Y. Li, and B.-H. F. Juang, "Deep learning in physical layer communications," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 93–99, Apr. 2019.
[51] A. Zappone, M. Di Renzo, M. Debbah, T. T. Lam, and X. Qian, "Model-aided wireless artificial intelligence: Embedding expert knowledge in deep neural networks for wireless system optimization," *IEEE Veh. Technol. Mag.*, vol. 14, no. 3, pp. 60–69, Sep. 2019.
[52] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Artificial neural networks-based machine learning for wireless networks: A tutorial," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3039–3071, 4th Quart., 2019.
[53] Y. Jin, J. Zhang, S. Jin, and B. Ai, "Channel estimation for cell-free mmWave massive MIMO through deep learning," *IEEE Trans. Veh. Technol.*, vol. 68, no. 10, pp. 10325–10329, Oct. 2019.
[54] 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.
[55] Z. Gao, Y. Wang, X. Liu, F. Zhou, and K.-K. Wong, "FFDNet-based channel estimation for massive MIMO visible light communication systems," *IEEE Wireless Commun. Lett.*, vol. 9, no. 3, pp. 340–343, Mar. 2020.
[56] C. Huang, R. Mo, and C. Yuen, "Reconfigurable intelligent surface assisted multiuser MISO systems exploiting deep reinforcement learning," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 8, pp. 1839–1850, Aug. 2020.
[57] Ö. Özdogan and E. Björnson, "Deep learning-based phase reconfiguration for intelligent reflecting surfaces," Sep. 2020. [Online]. Available: https://arxiv.org/abs/2009.13988
[58] J. Gao, C. Zhong, X. Chen, H. Lin, and Z. Zhang, "Unsupervised learning for passive beamforming," *IEEE Commun. Lett.*, vol. 24, no. 5, pp. 1052–1056, May 2020.
[59] A. Taha, M. Alrabeiah, and A. Alkhateeb, "Deep learning for large intelligent surfaces in millimeter wave and massive MIMO systems," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2019, pp. 1–6.

NEEL KANTH KUNDU (Graduate Student Member, IEEE) received the B.Tech. degree in electrical engineering with specialization in communication systems and networking from the Indian Institute of Technology Delhi, New Delhi, India, in 2018. He is currently pursuing the Ph.D. degree with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology.

From May 2016 to July 2016, he was a visiting student intern with the Rice Integrated Systems and Circuits Lab, Rice University, USA. From May 2017 to July 2017, he was a student trainee with Samsung Research Institute, Bengaluru, India. From May 2018 to August 2018, he was a visiting student with the Signal Processing and Computational Biology Lab, Hong Kong University of Science and Technology. His research interests include communication theory, MIMO signal processing, machine learning, and deep learning-based signal processing for future wireless communication systems and quantum communications. He is a recipient of the Hong Kong Ph.D. fellowship.

MATTHEW R. MCKAY (Fellow, IEEE) received the combined B.E. degree in electrical engineering and the B.I.T. degree in computer science from the Queensland University of Technology, Australia, and the Ph.D. degree in electrical engineering from the University of Sydney, Australia.

He is currently a Professor with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology (HKUST) and also with the Department of Chemical and Biological Engineering. He previously held the Hari Harilela Associate Professor title with HKUST. He was a Research Scientist with the Institute for Medical Engineering and Science, Massachusetts Institute of Technology, USA, in 2014, and with the Department of Statistics, Stanford University in 2015. His research interests include computational biology, communications and signal processing, and random matrix theory. He was awarded the University Medal upon graduating from the Queensland University of Technology. He received a 2010 Young Author Best Paper Award by the IEEE Signal Processing Society, and the 2011 Stephen O. Rice Prize by the IEEE Communication Society. He and his co-authors were also awarded best paper awards at six international conferences. He received the 2011 Young Investigator Research Excellence Award by the School of Engineering at HKUST, and the 2013 Best Young Researcher Award (Asia–Pacific Region) by the IEEE Communication Society. He serves as an Area Editor of Feature Articles for the *IEEE Signal Processing Magazine*, previously serving on the editorial board of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, *Mathematics*, and *Random Matrices: Theory and Applications*. In 2018, he was selected as a Young Scientist of the World Economic Forum, and in 2020, as a Young Scientist of the World Laureates Forum.