CHALMERS
UNIVERSITY OF TECHNOLOGY

Learning to Estimate RIS-Aided mmWave Channels

Downloaded from: https://research.chalmers.se, 2025-12-04 22:50 UTC

Citation for the original published paper (version of record):
He, J., Wymeersch, H., Di Renzo, M. et al (2022). Learning to Estimate RIS-Aided mmWave Channels. IEEE Wireless Communications Letters, 11(4): 841-845.
http://dx.doi.org/10.1109/LWC.2022.3147250

N.B. When citing this work, cite the original published paper.

© 2022 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, or reuse of any copyrighted component of this work in other works.

(article starts on next page)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2022.3147250, IEEE Wireless Communications Letters

Learning to Estimate RIS-Aided mmWave Channels

Jiguang He, Henk Wymeersch, *Senior Member, IEEE,* Marco Di Renzo, *Fellow, IEEE*, and Markku Juntti, *Fellow, IEEE*

***Abstract*—Inspired by the remarkable learning and prediction performance of deep neural networks (DNNs), we apply one special type of DNN framework, known as model-driven deep unfolding neural network, to reconfigurable intelligent surface (RIS)-aided millimeter wave (mmWave) single-input multiple-output (SIMO) systems. We focus on uplink cascaded channel estimation, where known and fixed base station combining and RIS phase control matrices are considered for collecting observations. To boost the estimation performance and reduce the training overhead, the inherent channel sparsity of mmWave channels is leveraged in the deep unfolding method. It is verified that the proposed deep unfolding network architecture can outperform the least squares (LS) method with a relatively smaller training overhead and online computational complexity.**

***Index Terms*—Deep unfolding, reconfigurable intelligent surface, cascaded channel estimation, deep neural network.**

I. Introduction

Reconfigurable intelligent surfaces (RISs) have recently been introduced for enhanced energy efficiency (EE), spectrum efficiency (SE), positioning accuracy, as well as network/physical-layer security [1]–[5]. The RIS, either being passive, active, or a hybrid combination of the former two, is used to smartly control the radio propagation environment, by virtue of multi-function capabilities, e.g., reflection, refraction, diffraction, scattering, and even absorption [6]. In the literature, the RIS is commonly used as an intelligent reflector, which breaks the well-known law of reflection [7], to mitigate the blockage effect and expand the connectivity range, especially for millimeter wave (mmWave) communications.

Since the RIS phase control and joint active and passive beamforming are sensitive to the channel state information (CSI) accuracy, the full potential of RIS cannot be achieved when the channel estimation (CE) is poorly performed. Therefore, accurate yet efficient CE methods for the individual channels or the cascaded channel are of vital importance. In our previous works, we took advantage of the inherent channel sparsity and rank-deficiency features of the mmWave multiple-input multiple-output (MIMO) channels and we applied the iterative reweighted method and the atomic norm minimization (ANM) method for estimating the channel parameters of RIS-aided mmWave MIMO systems [8], [9]. These works fall into the category of conventional model-based approaches, which suit only for a small- or medium-sized RIS, base station (BS), and mobile station (MS).

As the number of RIS elements and BS/MS antennas continues to grow (this is an inevitable trend in the mmWave spectrum), more training overhead to obtain adequate CE performance, within the channel coherence time, via conventional model-based methods is required. Besides, the associated computational complexity will become inevitably high. These considerations motivate the application of data-driven or hybrid approaches for CE in RIS-aided communications [10], [11]. In [12], a convolutional neural network (CNN) was considered for RIS CE in a multi-user scenario. Therein, however, each user needs to first estimate its own channel and then map the estimate to the corresponding ground-truth channel, which naturally increases the computational complexity. Therefore, in this paper, we resort to a model-driven deep unfolding approach, which has already been used in MIMO detection and sparse signal recovery [13], [14], for estimating the cascaded channel in RIS-aided mmWave single-input multiple-output (SIMO) systems.

Deep unfolding mimics the operation of conventional (projected) gradient descent algorithms, and it is capable of directly mapping the received pilot signals to the cascaded channel. Its computationally intensive training process can be executed offline and the online implementation/prediction phase only entails low-complexity calculations, e.g., matrix multiplications and additions, and element-wise operations. Besides, the step sizes and regularization parameters can be combined and optimized during the training of the deep unfolding model, which is not possible in traditional gradient descent methods. In this study, specifically, the rank-deficiency property of the cascaded channel is explicitly considered in the deep unfolding framework. It is verified that the deep unfolding scheme can outperform the least squares (LS) estimation and the ANM methods [9] in terms of normalized mean square error (NMSE) with reduced online computational complexity.

Notations: A bold lowercase letter $\mathbf{a}$ denotes a vector, and a bold capital letter $\mathbf{A}$ denotes a matrix. $(\cdot)^{\mathsf{T}}$ and $(\cdot)^{\mathsf{H}}$ denote the matrix or vector transpose and Hermitian transpose, respectively. $\mathrm{diag}(\mathbf{a})$ denotes a square diagonal matrix with the entries of $\mathbf{a}$ on its diagonal, $\mathbb{E}[\cdot]$ is the expectation operator, $\mathbf{0}$ denotes the all-zero vector, $\mathbf{I}_M$ denotes the $M \times M$ identity matrix, and $j = \sqrt{-1}$. $\|\cdot\|_{\mathrm{F}}$ and $\|\cdot\|_*$ denote the Frobenius norm and nuclear norm of a matrix, respectively, and $\|\cdot\|_2$ denotes the Euclidean norm of a vector.

II. System Model

We consider a nearly-passive RIS-aided mmWave SIMO network, where the MS communicates with the BS via the

J. He and M. Juntti are with Centre for Wireless Communications, FI-90014, University of Oulu, Finland, and J. He is also with International Institute of Next Generation Internet, Macau University of Science and Technology, Taipa 999078, China. H. Wymeersch is with Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden. M. Di Renzo is with Université Paris-Saclay, CNRS, CentraleSupélec, Laboratoire des Signaux et Systèmes, 3 Rue Joliot-Curie, 91192 Gif-sur-Yvette, France.

This work is supported by Horizon 2020, European Union's Framework Programme for Research and Innovation, under grant agreement no. 871464 (ARIADNE). This work is also partially supported by the Academy of Finland 6G Flagship project (grant 318927), as well as the H2020 RISE-6G project under grant agreement number 101017011.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2022.3147250, IEEE Wireless Communications Letters

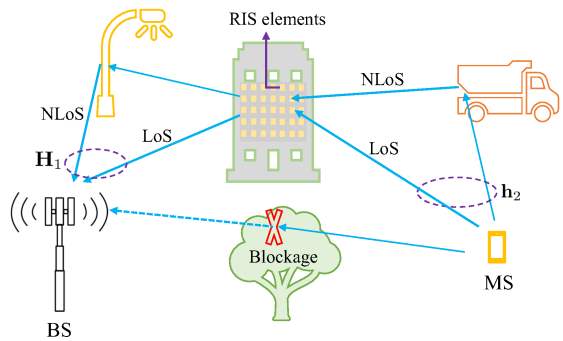

Fig. 1: A typical scenario for maintaining the connectivity by deploying an RIS in a mmWave SIMO network.

RIS, as shown in Fig. 1. The BS and RIS are equipped with multiple antennas and with nearly-passive scattering elements, respectively, while the MS is equipped with a single antenna. We further assume that the direct MS-BS channel is blocked.[1] The RIS-BS channel, which is denoted as $\mathbf{H}_1 \in \mathbb{C}^{M\times N}$ with M and N being the number of antennas and scattering elements at the BS and RIS, respectively, can be written as

$$\mathbf{H}_1 = \sum_{i=1}^{L_1} g_{1,i}\boldsymbol{\alpha}(\phi_{1,i})\boldsymbol{\alpha}^{\mathsf{H}}(\theta_{1,i}) = \mathbf{A}(\boldsymbol{\phi}_1)\mathrm{diag}(\mathbf{g}_1)\mathbf{A}^{\mathsf{H}}(\boldsymbol{\theta}_1), \tag{1}$$

where $L_1 \ll \min\{M, N\}$ is the number of channel paths, including one line-of-sight (LoS) path (with $i = 1$) and multiple non-line-of-sight (NLoS) paths (with $i > 1$), and $g_{1,i} \in \mathbb{C}$, $\theta_{1,i} \in \mathbb{R}$, and $\phi_{1,i} \in \mathbb{R}$ denote the propagation path gain, the angle of departure (AoD), and the angle of arrival (AoA) associated with the ith propagation path. The array response vector $\boldsymbol{\alpha}(\phi_{1,i}) \triangleq [1, \quad e^{j\pi\sin(\phi_{1,i})}, \ldots, e^{j(M-1)\pi\sin(\phi_{1,i})}]^{\mathsf{T}} \in \mathbb{C}^{M\times 1}$ is obtained by assuming half-wavelength inter-element spacing, and $\boldsymbol{\alpha}(\theta_{1,i}) \in \mathbb{C}^{N\times 1}$ can be formulated in the same manner. We assume $g_{1,1} \sim \mathcal{CN}(0, \sigma^2_{\text{LoS}})$, and define the vectors $\tilde{\mathbf{g}}_1 \triangleq [g_{1,2}, \ldots, g_{1,L_1}]^{\mathsf{T}} \sim \mathcal{CN}(\mathbf{0}, \sigma^2_{\text{NLoS}}\mathbf{I}_{L_1-1})$, $\mathbf{g}_1 \triangleq [g_{1,1}, \tilde{\mathbf{g}}_1^{\mathsf{T}}]^{\mathsf{T}}$, $\boldsymbol{\phi}_1 \triangleq [\phi_{1,1}, \ldots, \phi_{1,L_1}]^{\mathsf{T}}$, $\boldsymbol{\theta}_1 \triangleq [\theta_{1,1}, \ldots, \theta_{1,L_1}]^{\mathsf{T}}$, $\mathbf{A}(\boldsymbol{\phi}_1) \triangleq [\boldsymbol{\alpha}(\phi_{1,1}), \ldots, \boldsymbol{\alpha}(\phi_{1,L_1})]$, and $\mathbf{A}(\boldsymbol{\theta}_1) \triangleq [\boldsymbol{\alpha}(\theta_{1,1}), \ldots, \boldsymbol{\alpha}(\theta_{1,L_1})]$.

Similarly, the MS-RIS channel $\mathbf{h}_2 \in \mathbb{C}^{N\times 1}$ can be written as

$$\mathbf{h}_2 = \sum_{i=1}^{L_2} g_{2,i}\boldsymbol{\alpha}(\phi_{2,i}) = \mathbf{A}(\boldsymbol{\phi}_2)\mathbf{g}_2. \tag{2}$$

All the channel parameters in $\mathbf{h}_2$ are defined as those in $\mathbf{H}_1$. We also assume $g_{2,1} \sim \mathcal{CN}(0, \sigma^2_{\text{LoS}})$, $\tilde{\mathbf{g}}_2 \triangleq [g_{2,2}, \ldots, g_{2,L_2}]^{\mathsf{T}} \sim \mathcal{CN}(\mathbf{0}, \sigma^2_{\text{NLoS}}\mathbf{I}_{L_2-1})$, and define $\mathbf{g}_2 \triangleq [g_{2,1}, \tilde{\mathbf{g}}_2^{\mathsf{T}}]^{\mathsf{T}}$ and $\mathbf{A}(\boldsymbol{\phi}_2) \triangleq [\boldsymbol{\alpha}(\phi_{2,1}), \ldots, \boldsymbol{\alpha}(\phi_{2,L_2})]$ with $L_2 \ll N$.

The end-to-end uplink MS-RIS-BS channel (including the effect of the RIS) can be written as

$$\mathbf{h} = \mathbf{H}_1\boldsymbol{\Omega}\mathbf{h}_2 = \mathbf{H}_1\mathrm{diag}(\mathbf{h}_2)\boldsymbol{\omega}, \tag{3}$$

where $\boldsymbol{\Omega} = \mathrm{diag}(\boldsymbol{\omega})$ is the RIS phase control matrix, with $\boldsymbol{\omega} = [\omega_1, \ldots, \omega_N]^{\mathsf{T}}$ and $|\omega_i| = 1$ for $\forall i$ [1]. We are interested in low-cost and low-complexity implementations of RISs, we hence focus on RISs that can control only the phase response.

In (3), $\mathbf{H}_1\mathrm{diag}(\mathbf{h}_2)$ is referred to as the cascaded channel. By knowing it, we can optimize the RIS phase control matrix and BS beamforming/combining vector. Let us define it as $\mathbf{H}_c \in \mathbb{C}^{M\times N}$, i.e.,

$$\mathbf{H}_c = \mathbf{H}_1\mathrm{diag}(\mathbf{h}_2). \tag{4}$$

Based on the considered assumptions, we have $\mathrm{rank}(\mathbf{H}_1) = L_1$ and $\mathrm{rank}(\mathrm{diag}(\mathbf{h}_2)) = N$. Thus $\mathrm{rank}(\mathbf{H}_c) \leq \min\{\mathrm{rank}(\mathbf{H}_1), \mathrm{rank}(\mathrm{diag}(\mathbf{h}_2))\} = L_1$. The inherent channel sparsity (represented by the rank deficiency of the cascaded channel) can be applied in order to enable an efficient yet accurate CE of (4).

III. Learning to Estimate

In this section, we first introduce the channel sounding procedure, and then describe the optimization problem formulation for recovering the cascaded channel by using conventional (model-based) methods. Finally, we describe the model-driven deep unfolding method for estimating the rank-deficient cascaded channel.

A. Channel Sounding

During the sounding process, pilot signals are sent from the MS to the BS via the RIS. A different RIS phase control matrix is considered for each channel use while the combining matrix at the BS is fixed. The received signal at channel use k, for $k = 1, \ldots, K$, can be written as

$$\mathbf{y}[k] = \mathbf{W}^{\mathsf{H}}[k]\mathbf{H}_1\boldsymbol{\Omega}[k]\mathbf{h}_2 s[k] + \mathbf{W}^{\mathsf{H}}[k]\mathbf{n}[k], \tag{5}$$

where $\mathbf{W}[k] \in \mathbb{C}^{M\times N_{\text{W}}}$ is the combining matrix at the BS with N_{W} denoting the number of columns,[2] $s[k]$ is the pilot signal sent by the MS, and $\mathbf{n}[k] \sim \mathcal{CN}(0, \sigma^2)$ is the additive white Gaussian noise (AWGN) at the BS.

The received signal $\mathbf{y}[k]$ in (5) can be reformulated as

$$\mathbf{y}[k] = \mathbf{W}^{\mathsf{H}}[k]\mathbf{H}_c\boldsymbol{\omega}[k]s[k] + \mathbf{W}^{\mathsf{H}}[k]\mathbf{n}[k], \tag{6}$$

where $\boldsymbol{\Omega}[k] = \mathrm{diag}(\boldsymbol{\omega}[k])$.

Without loss of generality, we assume $s[1] = s[K] = 1$ and $\mathbf{W}[1] = \mathbf{W}[K] = \mathbf{W}$. The received signals $\mathbf{Y} = [\mathbf{y}[1], \ldots, \mathbf{y}[K]]$ can be rewritten as

$$\mathbf{Y} = \mathbf{W}^{\mathsf{H}}\mathbf{H}_c\bar{\boldsymbol{\Omega}} + \mathbf{W}^{\mathsf{H}}\mathbf{N}, \tag{7}$$

where $\bar{\boldsymbol{\Omega}} = [\boldsymbol{\omega}[1], \ldots, \boldsymbol{\omega}[K]]$ and $\mathbf{N} = [\mathbf{n}[1], \ldots, \mathbf{n}[K]]$. An additional vectorization step is considered for all the terms in (7), resulting in

$$\mathbf{y} = (\bar{\boldsymbol{\Omega}}^{\mathsf{T}} \otimes \mathbf{W}^{\mathsf{H}})\mathbf{h}_c + \mathbf{n}, \tag{8}$$

where $\mathbf{y} = \mathrm{vec}(\mathbf{Y}), \mathbf{h}_c = \mathrm{vec}(\mathbf{H}_c)$, and $\mathbf{n} = \mathrm{vec}(\mathbf{W}^{\mathsf{H}}\mathbf{N})$. Based on the vectorized received signal $\mathbf{y}$, we need to estimate

[1] When the direct MS-BS channel exists, we can estimate it by setting the RIS into an absorption mode. Then, we can estimate the cascaded channel by subtracting the direct MS-BS channel.

[2] We consider an analog combining matrix, which is a suitable choice for fulfilling the requirements of reduced-complexity hybrid precoding architectures commonly assumed for mmWave MIMO transceivers. When $N_{\text{W}} > N_{\text{RF}}$, with N_{RF} being the number of radio frequency (RF) chains at the BS, we need $K\lceil N_{\text{W}}/N_{\text{RF}}\rceil$ channel uses to complete the sounding process. Otherwise, K channel uses are sufficient.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2022.3147250, IEEE Wireless Communications Letters

the vectorized cascaded channel $\mathbf{h}_c$. Let us define $\mathbf{\Psi} = \bar{\mathbf{\Omega}}^{\mathsf{T}} \otimes \mathbf{W}^{\mathsf{H}}$. Then, (7) can be simplified as $\mathbf{y} = \mathbf{\Psi}\mathbf{h}_c + \mathbf{n}$. Based on the obtained signal model, the objective of this paper is to extract $\mathbf{h}_c$ from the noisy received signal $\mathbf{y}$ by assuming that the matrix $\mathbf{\Psi}$ is known. This is accomplished by using the deep unfolding method, which is detailed next.

B. Optimization Problem Formulation

In order to recover $\mathbf{h}_c$ from the noisy observation $\mathbf{y}$, we formulate the following regularized optimization problem

$$\hat{\mathbf{h}}_c = \arg\min_{\mathbf{h}_c} \|\mathbf{y} - \mathbf{\Psi}\mathbf{h}_c\|_2^2 + \lambda \operatorname{rank}(\mathbf{H}_c), \tag{9}$$

which takes into consideration the rank deficiency of the cascaded channel $\mathbf{H}_c$ and the impact of noise [15]. The regularization parameter $\lambda > 0$ is introduced to control the tradeoff between the data fitting and the rank of the cascaded channel. The optimization problem in (9) can be further reformulated as

$$\hat{\mathbf{h}}_c = \arg\min_{\mathbf{h}_c} \|\mathbf{y} - \mathbf{\Psi}\mathbf{h}_c\|_2^2 + \lambda \|\mathbf{H}_c\|_*, \tag{10}$$

which is obtained by relaxing $\operatorname{rank}(\mathbf{H}_c)$ with its nuclear norm, i.e., $\|\mathbf{H}_c\|_*$. This is a convenient approach because $\operatorname{rank}(\mathbf{H}_c)$ is a noncovex function of $\mathbf{H}_c$. In addition, $\|\mathbf{H}_c\|_{\mathrm{F}} \leq \|\mathbf{H}_c\|_* \leq \sqrt{r}\|\mathbf{H}_c\|_{\mathrm{F}}$ with $r \geq 1$. Notably, when $\operatorname{rank}(\mathbf{H}_c) = 1$ (e.g., $\mathbf{H}_1$ has only the LoS path), we have $\|\mathbf{H}_c\|_* = \|\mathbf{H}_c\|_{\mathrm{F}}$. Also, the singular values of $\mathbf{H}_c$ have a high probability to fulfill the following condition: $\sigma_1 \gg \sigma_2 > \cdots > \sigma_{L_1}$ with σ_i being the ith largest singular value of $\mathbf{H}_c$, which results in $\|\mathbf{H}_c\|_* \approx \|\mathbf{H}_c\|_{\mathrm{F}}$. Thus, we further replace $\operatorname{rank}(\mathbf{H}_c)$ in (10) with $\|\mathbf{H}_c\|_{\mathrm{F}}$, i.e., $\|\mathbf{h}_c\|_2$, which yields

$$\hat{\mathbf{h}}_c = \arg\min_{\mathbf{h}_c} \|\mathbf{y} - \mathbf{\Psi}\mathbf{h}_c\|_2^2 + \lambda \|\mathbf{h}_c\|_2. \tag{11}$$

To accurately solve this optimization problem, we need to carefully choose the regularization parameter λ. The optimal value of λ is, however, difficult to obtain. As a reference, λ is chosen equal to $4\sigma^2\sqrt{\frac{MN(M+N)\log(M+N)}{N_{\mathrm{W}}K}}$ [15].

C. Model-Driven Deep Unfolding

Deep unfolding is a deep neural network framework that mimics the conventional gradient descent method. The difference lies in that deep unfolding is able to learn from a large amount of (synthetic) data with enhanced performance and reduced online implementation complexity and number of iterations. Typically, this is exemplified in a reduced number of layers in the deep unfolding network. When solving the optimization problem in (11) by using the (conventional) gradient descent method, we iteratively update $\mathbf{h}_c^{(i)}$, with (i) denoting the iteration index, as follows

$$\mathbf{h}_c^{(i)} = \mathbf{h}_c^{(i-1)} - \beta \nabla f(\mathbf{h}_c^{(i-1)}), \tag{12}$$

where $0 < \beta < 1$ is the step size and $\nabla f(\mathbf{h}_c^{(i-1)})$ is the gradient of $f(\mathbf{h}_c) = \|\mathbf{y} - \mathbf{\Psi}\mathbf{h}_c\|_2^2 + \lambda\|\mathbf{h}_c\|_2$ evaluated at $\mathbf{h}_c^{(i-1)}$, which can be expressed as

$$\nabla f(\mathbf{h}_c^{(i-1)}) = (\mathbf{\Psi}^{\mathsf{H}}\mathbf{\Psi}\mathbf{h}_c^{(i-1)} - \mathbf{\Psi}^{\mathsf{H}}\mathbf{y}) + \lambda \mathbf{h}_c^{(i-1)}/\|\mathbf{h}_c^{(i-1)}\|_2, \tag{13}$$

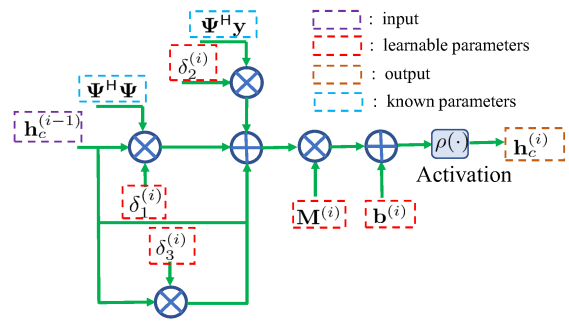

Fig. 2: The ith layer of the deep unfolding network model for estimating the cascaded channel vector.

when λ is fixed. The initial value, i.e, $\mathbf{h}_c^{(0)}$, can be set equal to the all-zero vector. In this case, the denominator $\|\mathbf{h}_c^{(i-1)}\|_2$ in the last term of (13) needs to be modified as $\|\mathbf{h}_c^{(i-1)}\|_2 + \epsilon$ with $\epsilon > 0$ when $i = 1$.

Substituting (13) into (12), we obtain

$$\mathbf{h}_c^{(i)} = \mathbf{h}_c^{(i-1)} - \beta\mathbf{\Psi}^{\mathsf{H}}\mathbf{\Psi}\mathbf{h}_c^{(i-1)} + \beta\mathbf{\Psi}^{\mathsf{H}}\mathbf{y} - \beta\lambda\mathbf{h}_c^{(i-1)}/\|\mathbf{h}_c^{(i-1)}\|_2. \tag{14}$$

The Gram matrix $\mathbf{\Psi}^{\mathsf{H}}\mathbf{\Psi}$, the compressed statistics $\mathbf{\Psi}^{\mathsf{H}}\mathbf{y}$, and $\mathbf{h}_c^{(i-1)}$, are needed to apply the gradient descent algorithm. Therefore, these three terms constitute the input variables of the deep unfolding neural network model.

To be specific, the ith layer of the deep unfolding model for mimicking the gradient descent iteration in (14) is introduced in Fig. 2. The accuracy and convergence speed of the channel estimate in (14) highly depends on the specific choice of the step size β and the regularization parameter λ. In the considered deep unfolding model, these parameters are learnable parameters that are automatically determined during the data-driven training phase. To this end, we introduce three generalized learnable parameters $\delta_1^{(i)} \in [-1,\ 0]$, $\delta_2^{(i)} \in [0,\ 1]$, and $\delta_3^{(i)} \in [-1,\ 0]$, for the ith layer of the deep unfolding model in Fig. 2. In (14), more specifically, $-\beta$ and β in the second and third term are replaced by $\delta_1^{(i)}$ and $\delta_2^{(i)}$, respectively, and the product $-\beta\lambda/\|\mathbf{h}_c^{(i-1)}\|_2$ in the last term is unfolded in the learnable parameter $\delta_3^{(i)}$. As shown in Fig. 2, in order to further enhance the prediction capabilities of the deep unfolding network model, we serially concatenate the term $\mathbf{h}_c^{(i-1)} + \delta_1^{(i)}\mathbf{\Psi}^{\mathsf{H}}\mathbf{\Psi}\mathbf{h}_c^{(i-1)} + \delta_2^{(i)}\mathbf{\Psi}^{\mathsf{H}}\mathbf{y} + \delta_3^{(i)}\mathbf{h}_c^{(i-1)}$ with a learnable weight matrix $\mathbf{M}^{(i)}$, a bias vector $\mathbf{b}^{(i)}$, and a nonlinear activation function.

The complete deep unfolding network model is illustrated in Fig. 3 and it comprises L layers from Fig. 2. In particular, the observations $\{\mathbf{\Psi}^{\mathsf{H}}\mathbf{\Psi}, \mathbf{\Psi}^{\mathsf{H}}\mathbf{y}\}$ are input to all the layers of the deep unfolding network model. The online computational complexity of the proposed scheme is $\mathcal{O}(M^2N^2L)$, which is smaller than the ANM based scheme that requires $\mathcal{O}((M+N)^6)$ per iteration [9].

IV. Numerical Results

In this section, we evaluate the performance of the proposed deep unfolding network model against two benchmark schemes: (i) the LS estimator and (ii) the direct solution of (11)

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2022.3147250, IEEE Wireless Communications Letters

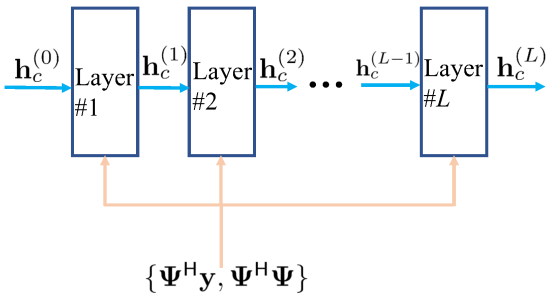

Fig. 3: Complete deep unfolding network model for channel estimation, which comprises L layers from Fig. 2

by using the CVX toolbox.[3] We evaluate the impact of the training overhead, the training SNR, the number of paths, and the angular parameter distribution. As far as the RIS phase control matrices as concerned, their diagonal elements are set equal to the column vectors of a discrete Fourier transform (DFT) matrix. A set of orthonormal vectors are considered for $\mathbf{W}$, e.g., the normalized column vectors from a DFT matrix. It is worth mentioning that we transform the data from the complex-valued domain to the real-valued domain before applying the deep unfolding network model. The channel $\mathbf{h}_c^{(0)}$ is set equal to the all-zero vector. In the first $L-1$ layers, we use *relu* activation functions, while no activation function is applied in the last layer. The loss function during the training phase is the NMSE between the output cascaded channel vector and the ground-truth cascaded channel vector. The *Adam* algorithm is used for training, whose learning rate is 0.001 during the first 20 epochs and it is halved in the remaining epochs, and the batch size is 64. We use $1e5$ samples for training and $1e4$ samples for testing. The parameter setup is summarized in Table I.

A. Impact of the Training Overhead

Fig. 4 shows the NMSE for a 1×16 SIMO system (i.e., $M = 16$) by using the proposed deep unfolding approach as a function of the training overhead. During the training phase, the SNR is $\gamma = 1/\sigma^2 = 20$ dB. From Fig. 4, we see that the proposed scheme with $K = 24$ channel uses for training outperforms the LS estimator even if the latter uses a longer training phase with $K = 32$. Also, as expected, the higher the training overhead is, the lower the NMSE of the proposed scheme is. It is worth noting that the proposed deep unfolding method outperforms the numerical solution of (11) by using CVX, and the ANM algorithm. This is attributed to the learning capability of the deep unfolded network through the learnable parameters introduced in Fig. 2.[4]

B. Impact of the Number of Paths

In this subsection, we study the impact of the number of paths, which is increased from one to two and three. The SNR for training is $\gamma = 20$ dB and $K = 28$. The corresponding

[3]The scripts of the implementation are available at https://github.com/jiguanghe/RISCE.

[4]If the regularization parameter λ in (11) can be optimally designed, better performance can be expected.

TABLE I: Parameter Setup.

Parameter	Value	Parameter	Value
L_1	$\{1, 2, 3\}$	$\mathbf{h}_c^{(0)}$	$\mathbf{0}$
L_2	$\{1, 2, 3\}$	Optimizer	*Adam*
σ^2_{LoS}	1	Learning rate	$0.001, 0.0005$
σ^2_{NLoS}	0.01	Batch size	64
M	16	$\rho(\cdot)$	*relu*
N	32	Training samples	$1e5$
L	15	Testing samples	$1e4$
$\sin(\boldsymbol{\theta}_1)$	$\mathcal{U}[0,\ 1]$	Loss	NMSE
$\sin(\boldsymbol{\phi}_1)$	$\mathcal{U}[0,\ 1]$	$\sin(\boldsymbol{\phi}_2)$	$\mathcal{U}[0,\ 1]$
N_{W}	8	K	$\{24, 28\}$

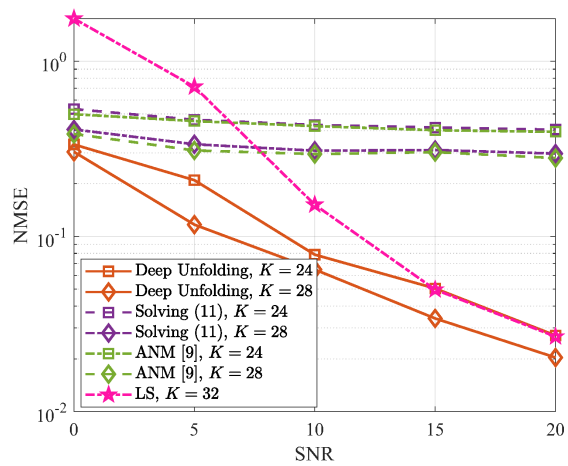

Fig. 4: Impact of training overhead on estimating the cascaded channel for a 1×16 SIMO system with $N = 32$ and $L_1 = L_2 = 1$. Deep unfolding vs. LS estimation, ANM [9], and the direct solution of (11) [15].

NMSE is shown in Fig. 5. When the number of paths increases, the rank of the cascaded channel increases accordingly. In this case, Fig. 5 shows that the NMSE increases as the number of paths increases. In other words, by keeping the training overhead fixed, the proposed deep unfolding method benefits from the sparsity of the channel, i.e., the number of paths is small.

C. Impact of the Training SNR

In this subsection, we study the impact of the training SNR on the estimation performance. Two SNR values, i.e., $\gamma = 0$ dB, $\gamma = 20$ dB, and an SNR that varies in the set $\gamma \in \{0, 5, 10, 15, 20\}$ dB are considered. The corresponding NMSE is shown in Fig. 6. We observe that the proposed deep unfolding method provides the best NMSE when it is trained at a high SNR except if the operating SNR during the test phase is too low. In other words, nearly noise-free training samples bring the best performance in the high SNR regime, and vice versa, as depicted in Fig. 6. The training SNR plays a critical role in the prediction performance of the deep unfolding models.[5]

D. Impact of the Angular Parameter Distribution

Unlike the previous study in which the angular parameters are distributed as $\mathcal{U}[0,\ 1]$, in this subsection, we evaluate the

[5]If the SNR value(s) can be incorporated into the proposed deep unfolding structure, better performance is expected, which is left for our future investigation.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2022.3147250, IEEE Wireless Communications Letters

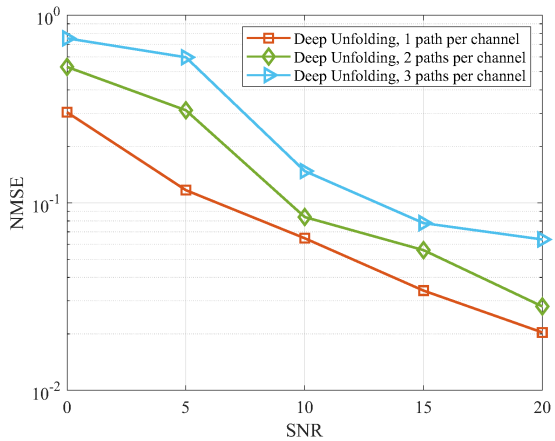

Fig. 5: Impact of the number of paths on estimating the cascaded channel for a 1×16 SIMO system with $N = 32$.

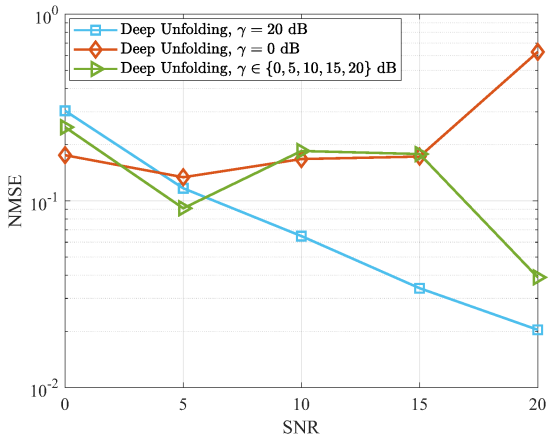

Fig. 6: Impact of the training SNR on estimating the cascaded channel for a 1×16 SIMO system with $N = 32$.

impact of the angular parameter distribution when estimating the cascaded channel. The corresponding NMSE is shown in Fig. 7. We observe that the NMSE decreases when the range of $\sin(\boldsymbol{\theta}_1), \sin(\boldsymbol{\phi}_1), \sin(\boldsymbol{\phi}_2)$ decreases. In other words, the estimation accuracy of the proposed deep unfolding method increases when the individual channels are subject to a reduced variability.

V. Conclusions

In this letter, we have introduced a deep unfolding model for efficiently estimating the end-to-end RIS channel in mmWave SIMO systems. With the aid of simulation results, we have shown that the proposed approach can outperform three benchmark schemes based on the LS method, a CVX-based numerical solution of the channel estimation problem, and the ANM algorithm. The impact of the number of paths, the training SNR, and the angular parameter distribution on the estimation accuracy has been investigated. In addition, the proposed deep unfolding network model has a low online prediction complexity, since it requires the computation of vector matrix multiplications and additions. On the other hand, the LS estimation methods usually require a matrix inversion.

Possible generalizations of the present work include the channel estimation problem in RIS-aided multiuser MIMO systems, and the joint optimization of the active and passive beamforming at the BS and at the RIS, respectively.

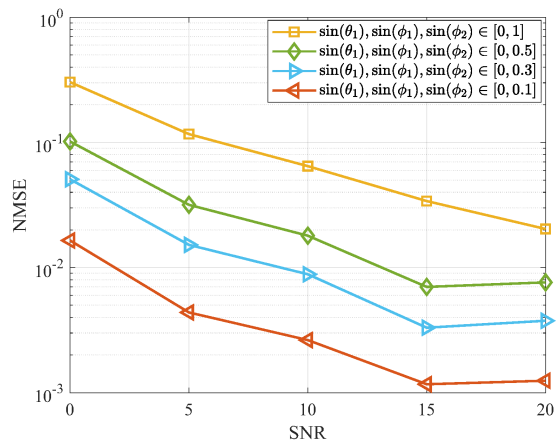

Fig. 7: Impact of the angular parameter distribution on estimating the cascaded channel for a 1×16 SIMO system with $N = 32$.

References

[1] Q. Wu and R. Zhang, "Beamforming optimization for wireless network aided by intelligent reflecting surface with discrete phase shifts," *IEEE Trans. Commun.*, pp. 1–1, 2019.

[2] C. Liaskos, A. Tsioliaridou, A. Pitsillides, S. Ioannidis, and I. F. Akyildiz, "Using any surface to realize a new paradigm for wireless communications," *Communications of the ACM*, vol. 61, pp. 30–33, 2018.

[3] M. Di Renzo *et al.*, "Smart radio environments empowered by reconfigurable AI meta-surfaces: an idea whose time has come," *EURASIP J. Wireless Comm. and Networking*, vol. 2019, p. 129, 2019.

[4] H. Wymeersch, J. He, B. Denis, A. Clemente, and M. Juntti, "Radio localization and mapping with reconfigurable intelligent surfaces: Challenges, opportunities, and research directions," *IEEE Veh. Technol. Mag.*, vol. 15, no. 4, pp. 52–61, 2020.

[5] D. Li, "Fairness-aware multiuser scheduling for finite-resolution intelligent reflecting surface-assisted communication," *IEEE Commun. Lett.*, vol. 25, no. 7, pp. 2395–2399, 2021.

[6] 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, 2020.

[7] E. Basar *et al.*, "Wireless communications through reconfigurable intelligent surfaces," *IEEE Access*, vol. 7, pp. 116 753–116 773, 2019.

[8] J. He, M. Leinonen, H. Wymeersch, and M. Juntti, "Channel estimation for RIS-aided mmWave MIMO channels," in *proc. IEEE Global Communications Conference*, 2020, pp. 1–6.

[9] J. He, H. Wymeersch, and M. Juntti, "Channel estimation for RIS-aided mmWave MIMO systems via atomic norm minimization," *IEEE Trans. Wireless Commun.*, pp. 1–1, 2021.

[10] A. Zappone, M. Di Renzo, and M. Debbah, "Wireless networks design in the era of deep learning: Model-based, AI-based, or both?" *IEEE Trans. Commun.*, vol. 67, no. 10, pp. 7331–7376, 2019.

[11] K. Tekbyk, G. K. Kurt, C. Huang, A. R. Ekti, and H. Yanikomeroglu, "Channel estimation for full-duplex RIS-assisted HAPS backhauling with graph attention networks," in *proc. IEEE ICC*, 2021, pp. 1–6.

[12] A. M. Elbir, A. Papazafeiropoulos, P. Kourtessis, and S. Chatzinotas, "Deep channel learning for large intelligent surfaces aided mm-wave massive MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 9, no. 9, pp. 1447–1451, 2020.

[13] N. Samuel, T. Diskin, and A. Wiesel, "Learning to detect," *IEEE Trans. Signal Process.*, vol. 67, no. 10, pp. 2554–2564, 2019.

[14] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *proc. international conference on machine learning*, 2010, pp. 399–406.

[15] Y. Chen and Y. Chi, "Harnessing structures in big data via guaranteed low-rank matrix estimation: Recent theory and fast algorithms via convex and nonconvex optimization," *IEEE Signal Process. Mag.*, vol. 35, no. 4, pp. 14–31, 2018.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/