MIMO Channel Estimation Using Score-Based Generative Models

Marius Arvinte, Member, IEEE, and Jonathan I. Tamir, Member, IEEE

Abstract—Channel estimation is a critical task in multiple-input multiple-output (MIMO) digital communications that substantially affects end-to-end system performance. In this work, we introduce a novel approach for channel estimation using deep score-based generative models. A model is trained to estimate the gradient of the logarithm of a distribution and is used to iteratively refine estimates given measurements of a signal. We introduce a framework for training score-based generative models for wireless MIMO channels and performing channel estimation based on posterior sampling at test time. We derive theoretical robustness guarantees for channel estimation with posterior sampling in single-input single-output scenarios, and experimentally verify performance in the MIMO setting. Our results in simulated channels show competitive in-distribution performance, and robust out-of-distribution performance, with gains of up to 5 dB in end-to-end coded communication performance compared to supervised deep learning methods. Simulations on the number of pilots show that high fidelity channel estimation with 25% pilot density is possible for MIMO channel sizes of up to $64 \times 256$. Complexity analysis reveals that model size can efficiently trade performance for estimation latency, and that the proposed approach is competitive with compressed sensing in terms of floating-point operation (FLOP) count.

Index Terms—Deep learning, generative, score-based, diffusion, MIMO, channel estimation.

I. INTRODUCTION

MASSIVE MIMO represents a key technology in fifth generation (5G) and envisioned sixth generation (6G) communication systems [1], [2], promising to increase communication reliability by orders of magnitude without increasing bandwidth requirements. With the deployment of millimeter wave (mmWave) band communications, recovering accurate, high-dimensional channel state information (CSI) becomes challenging. The number of antennas (or channel taps) reachable at the receiver for mmWave communications is orders-of-magnitude larger than for traditional wideband frequencies, necessitating improved channel state estimation methods. The number of measurements (pilots) required for channel estimation grows with the number of channel taps, leading to reduced system throughput. In this paper, inspired by recent results that show the potential of score-based (diffusion) generative models for specialized applications such as magnetic resonance imaging (MRI) reconstruction [16], [17], we introduce a training and inference algorithm for wireless channel estimation using score-based generative models in a single-carrier, point-to-point MIMO communication scenario. We use a distribution learning approach for modeling high-dimensional, mmWave MIMO channels in a stochastic environment. We model the log-distribution of channels by learning the high-dimensional gradient—known as the score. To learn the score of the distribution, we use score-based generative models, originally introduced in [18], and that have so far been primarily demonstrated on natural image benchmark datasets [19].

In this paper, inspired by recent results that show the potential of score-based (diffusion) generative models for specialized applications such as magnetic resonance imaging (MRI) reconstruction [16], [17], we introduce a training and inference algorithm for wireless channel estimation using score-based generative models in a single-carrier, point-to-point MIMO communication scenario. We use a distribution learning approach for modeling high-dimensional, mmWave MIMO channels in a stochastic environment. We model the log-distribution of channels by learning the high-dimensional gradient—known as the score. To learn the score of the distribution, we use score-based generative models, originally introduced in [18], and that have so far been primarily demonstrated on natural image benchmark datasets [19]. For training, a database of known channels is used to train a score-based generative model in an unsupervised manner, that is independent of the pilot symbols. During test time, we perform probabilistic channel estimation by sampling from the posterior distribution using annealed Langevin dynamics. We tackle the challenges that arise when performing channel estimation in an out-of-distribution setting, i.e., to environments not seen during training, as well as in a very wide (up to 40 dB) signal-to-noise ratio (SNR) range and in interference scenarios. Numerical simulations show that using the proposed approach, estimation performance is retained even when the

Manuscript received 17 May 2022; revised 9 September 2022; accepted 29 October 2022. Date of publication 21 November 2022; date of current version 12 June 2023. This work was supported in part by the Office of Naval Research (ONR) under Grant N00014-19-1-2590; in part by the National Science Foundation (NSF) Institute for Foundations of Machine Learning (IFML) under Grant 2019844; and in part by InterDigital, an affiliate of National Science Foundation (NSF) Institute for Foundations of Machine Learning (IFML). Color versions of one or more figures in this article are available at https://doi.org/10.1109/TWC.2022.3220784.

Digital Object Identifier 10.1109/TWC.2022.3220784
simulated test channels come from a different distribution than the one during training, and gains of up to 5 dB in energy-per-bit to noise ratio ($E_b/N_0$) are achieved against competing deep learning methods in coded end-to-end communication system simulations. We also investigate the computational complexity of the proposed approach, and identify promising future research directions in developing more compact and efficient deep neural networks to reduce estimation latency.

A. Related Work

Modern statistical wireless channel modeling stems from the Saleh-Valenzuela clustered channel model [20], which characterizes a propagation environment through a sum of impulse responses with stochastic delays and amplitudes. Extensions for MIMO channels have added angles of departure and arrival on both ends of the communication link [21], and have included beamforming effects in the modeling [22]. Standardized clustered delay line (CDL) models for cellular communications are extremely flexible and can model wireless channels in four dimensions (time, frequency, transmit antennas, and receive antennas), and are adopted in 5G specifications [23].

In this work, we use the well-established CDL family of stochastic environments to evaluate the performance of MIMO channel estimation algorithms. Beyond their usage for synthetic channel generation and performance evaluation, statistical channel models have been used to aid compressed sensing (CS) for MIMO channel estimation, based on an assumed low rank channel model in the beamspace domain [24]. The Lasso is a fast estimation method that imposes sparsity in the two-dimensional Fourier (2D-DFT) domain through $\ell_1$-regularization in the channel estimation objective. The EM-GM-AMP algorithm [25] uses the approximate message passing (AMP) algorithm as a backbone in recovering a Gaussian mixture of impulses in the beamspace domain, from a number of undersampled and noisy measurements. Along the same line, the atomic norm decomposition [26] and its extension to fast approximate atomic norm decomposition (faAD) [27] use knowledge of the antenna array shape and the underlying structure of the CDL models (sparsity in the continuous 2D angular domain) to achieve competitive channel estimation results. In contrast to CS methods, score-based models do not assume a specific channel model, and thus are likely amenable to use in real-world wireless propagation environments where sparsity and low rank assumptions may not always hold [28].

The work in [4] is an application of the compressed sensing with generative models (CSGM) framework [29] to wireless channel estimation. This method trains a deep generative adversarial network (GAN) with a low-dimensional latent space using a training set of channels, and at test-time formulates an optimization problem to recover the channel state information matrix using the received pilot symbols and the pre-trained model. In contrast to GANs, score-based generative models do not use adversarial training, and make less restrictive assumptions about the low-dimensional nature of the wireless channels, instead aiming to learn the score of the distribution even in regions with a low probability density.

The recent work in [30] introduces an unsupervised learning approach that jointly learns the best hyper-parameters and a refinement model for optimal precoding matrices. This is similar to the iterative denoising approach done with Langevin dynamics. Conditional generative models have been used to learn the effects of unknown channels [31], where a deep model is trained to map transmitted symbols to received symbols. The goal of this approach is to model transmitter and receiver effects (e.g., non-linear power amplification or analog-to-digital conversion) alongside channel effects, which is useful for learning black-box approximations to entire communication chains [32].

Supervised end-to-end training of deep learning based methods has been successfully used for wireless MIMO channel estimation. The algorithm proposed in [33] introduces a two-stage deep learning approach for two-dimensional channel estimation: in the first stage, a super-resolution network is used, followed by a denoising stage. While originally introduced for time-frequency channels, this approach is applicable to MIMO channels as well, where an initial estimate of the channel matrix is treated as an image [34]. The work in [3] introduces a powerful and robust deep learning algorithm in the form of the learned denoising approximate message passing (L-DAMP) algorithm [35]. In L-DAMP, differentiable optimization steps are interleaved with forward passes of a deep neural network, and the entire method is trained end-to-end using back-propagation.

Score-based generative models – closely related to diffusion models [36] – are introduced in [18], where the authors also introduce annealed Langevin dynamics and how to efficiently sample from a distribution of interest using a learned score model. These models have been recognized to surpass generative adversarial networks (GANs) at modeling real-world distributions, such as natural images [19]. The work in [37] introduces practical techniques for improving the training of score-based models, which we leverage and adapt for wireless channels. The work in [38] theoretically proves the in-distribution optimality of posterior sampling and links this to the full-dimensional support coverage capability of a model.

B. Contributions

Our contributions in this work are the following:

- We propose a posterior sampling based solution for MIMO channel estimation. Given received pilots, at test (inference) time the algorithm samples from the posterior distribution of channel state information conditioned on the received pilots. In the training phase, a denoising score matching framework learns the score (the gradient of log-prior distribution) of high-dimensional MIMO channels. During inference, we use the learned score-based model in conjunction with the received pilots to iteratively update the channel estimate and perform posterior sampling (Algorithm 1).

- We derive a closed-form expression that describes robustness guarantees for multi-tap, single-input single-output (SISO) channel estimation with posterior sampling in an out-of-distribution setting, under assumptions of mutual
independence between the tap gains and locations. The derived expression bounds the probability of successful recovery using the mismatch-to-noise ratio. This predicts the behaviour of the proposed method as a function of the SNR and the distributional mismatch between training and testing environments.

- We perform numerical simulations showing the performance of score-based models and a set of diverse baselines on the CDL family of channel models [23] at mmWave frequencies. Our main findings are that score-based generative models are near-optimal in-distribution, many baselines on the CDL family of channel models [23] at mmWave frequencies. Our main findings are that score-based generative models are near-optimal in-distribution, and is constrained to have unit amplitude and low-resolution phase — each entry of $\mathbf P$ is a randomly chosen (fixed for all test samples) quadrature phase shift keying (QPSK) symbol. Assuming that the channel state information is constant across $N_p$ pilot transmissions, we obtain the matrix model:

$$\mathbf Y = \mathbf H \mathbf P + \mathbf N, \quad (2)$$

Channel estimation requires estimating the channel state information matrix $\mathbf H$ using the received pilot matrix $\mathbf Y$, while having knowledge of the transmitted pilot matrix $\mathbf P$. The latter part is common in communication standards, where pilot sequences are pre-specified [39]. Let $\alpha = N_p/N_t$ denote the pilot density. When $\alpha < 1$, then there are $N_rN_t < N_pN_t$ received pilots, and channel estimation is an under-determined inverse problem.

### C. Notation and Organization

We use boldfaced letters $\mathbf x, \mathbf X$ to denote vectors and matrices, respectively, and $x_{i,j}$ to denote the $(i, j)$-th element of a matrix. We use italic letters to denote their random variable counterparts. The notation $p_X(\mathbf X)$ represents the continuous probability distribution function of the matrix random variable $\mathbf X$, evaluated at the matrix $\mathbf X$. We use $\mathbf X^H$ to denote the Hermitian (conjugate transpose) of $\mathbf X$. The matrix $\mathbf I$ denotes the identity matrix of appropriate size. We use $\| \cdot \|_F$ to denote the Frobenius norm of a matrix, $\| \cdot \|_{1,1}$ to denote the $\ell_1$-norm of a vectorized matrix, $\mathcal O$ to denote proportionality up to a multiplicative constant, and $\log$ to denote the natural logarithm. We use $\nabla f(\mathbf X)$ to denote the derivative of a function $f$ with respect to a value $\mathbf X$, as the matrix with the $(i, j)$-th entry given by $\partial f(\mathbf X)/\partial x_{i,j}$. We use $d\mathbf X$ to denote an infinitesimal change in a random process $\mathbf X$.

The remainder of the paper is structured as follows: Section II introduces the MIMO system model and defines the score of a distribution. Section III introduces the training and inference (testing) stages of the proposed approach, while Section IV introduces our theoretical derivations that characterize the probability of successful estimation. Section V presents numerical simulation results and discussions, and Section VI concludes the paper.

### II. Preliminaries

#### A. Wireless System Model

We consider a narrowband, point-to-point MIMO communication scenario between a transmitter and receiver with $N_t$ and $N_r$ antennas, respectively. Propagation in this scenario is characterized by the channel state information matrix $\mathbf H \in \mathbb C^{N_r \times N_t}$. We let $\mathbf p_i \in \mathbb C^{N_t}$ denote the $i$-th pilot symbol chosen from a pre-designed codebook of $N_p$ pilots, and $\mathbf W \in \mathbb C^{N_r \times N_t}$ be the receive beamforming matrix. The received signal vector for the $i$-th beamforming matrix.

$$y_i = \mathbf W^H(\mathbf H p_i + \mathbf n_i), \quad (1)$$

where $\mathbf n_i$ is complex additive white Gaussian noise with zero mean and power covariance matrix $\sigma^2_{p_i} \mathbf I$, and $s$ is a complex-valued scalar. In practice, the pilot vectors are selected as entries from a beamforming codebook with structural constraints [39]. For the remainder of the paper, we make no assumptions on the structure of $\mathbf p_i$. We assume that $s = 1$, $\mathbf W = \mathbf I$ (fully digital receiver), and that $\mathbf p_i$ is constrained to have unit amplitude and low-resolution phase — each entry of $\mathbf P$ is a randomly chosen (fixed for all test samples) quadrature phase shift keying (QPSK) symbol. Assuming that the channel state information is constant across $N_p$ pilot transmissions, we obtain the matrix model:

$$\mathbf Y = \mathbf H \mathbf P + \mathbf N, \quad (2)$$

Channel estimation requires estimating the channel state information matrix $\mathbf H$ using the received pilot matrix $\mathbf Y$, while having knowledge of the transmitted pilot matrix $\mathbf P$. The latter part is common in communication standards, where pilot sequences are pre-specified [39]. Let $\alpha = N_p/N_t$ denote the pilot density. When $\alpha < 1$, then there are $N_rN_t < N_pN_t$ received pilots, and channel estimation is an under-determined inverse problem.

#### B. Langevin Dynamics

Langevin dynamics are a process for sampling from arbitrary distributions. Let $p_H$ be an arbitrary, high-dimensional probability distribution and $H_t$ be a Langevin diffusion process on the same probability space as $p_H$, defined by the dynamics [40]:

$$dH_t = dW_t + \frac{1}{2} \nabla \log p_H(H_t), \quad (3)$$

where $W_t$ is an i.i.d. zero mean and unit variance Gaussian random process. A remarkable property of the Langevin diffusion process is that the stationary distribution of $H_t$ converges in probability to $p_H$ as $t \to \infty$, regardless of the initial value $H_0$, assuming that certain smoothness conditions for $p_H$ are met [40]. Running the process in (3) for a given initial value is thus a way to sample from the distribution $p_H$. In practice, numerical machine computations require finite precision and updates to $\mathbf H$. The discretized Langevin dynamics update equation with step size $\epsilon$ is given by [40]:

$$H_{t+1} \leftarrow H_t + \epsilon \cdot \nabla \log p_H(H_t) + \sqrt{2\epsilon} \cdot \zeta_t, \quad (4)$$

where $\zeta_t$ is i.i.d. Gaussian noise with zero mean and unit variance. The work [40] analyzes the convergence rate of (4) and finds that convergence (also termed mixing) can be slower than the continuous process in (3). To alleviate slow mixing times, practical modifications have been proposed to (4). In this work, we leverage annealed Langevin dynamics [18], where time-varying hyper-parameters $\alpha_t$ and $\beta_t$ are introduced to improve convergence and yield the update rule:

$$H_{t+1} \leftarrow H_t + \alpha_t \cdot \nabla \log p_H(H_t) + \beta_t \cdot \zeta_t. \quad (5)$$

1 https://github.com/utcsilab/score-based-channels

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
The above consists of two additive update terms to the current iterate $H_i$:

- $\alpha_i \cdot \nabla \log p_H(H_i)$ increases the likelihood of the current sample. Because the gradient is the local direction of steepest ascent, this update term guides $H_i$ to a more plausible sample under the distribution $p_H$, i.e., towards a point of the distribution with higher density.

- $\beta_i \cdot \zeta_i$ represents a perturbation to the above process. If the hyper-parameter $\beta_i$ is chosen correctly, this perturbation allows for sample diversity and prevents always sampling the mode of the distribution [38].

Given a target criterion and a set of validation samples, these hyper-parameters can be tuned using an approximate grid search approach. The only issue that remains is how can one obtain query access to $\nabla \log p_H(H)$, for arbitrary $H$. This is the central purpose of score-based generative modeling and described next.

C. The Score of a Distribution

An important assumption of this paper is that wireless channels are sampled from a distribution. This is a realistic assumption for real-world wireless environments, that has been extensively used in both theoretical [41] and practical works [31]. Let $p_H$ denote the distribution of complex-valued MIMO channels for an arbitrary, stochastic environment introduced in Section II-A. The score of $p_H$ at $H$ is defined as [42]:

$$\psi_H(H) = \nabla \log p_H(H),$$

where $\psi_H(H) \in \mathbb{C}^{N_r \times N_t}$. While the score function can be used to sample from $p_H$ using annealed Langevin dynamics, it is generally intractable for non-trivial distributions, including realistic models of wireless channels. The result in [42] shows that it is possible to learn an approximation to the score function using an unsupervised formulation. For the remainder of this work, we do not make any explicit assumptions on $p_H$ or $\psi_H$. That is, we do not assume the existence of a low-dimensional (sparse) representation of MIMO channels, and the proposed recovery approach does not use explicit information about $p_H$ (the channel statistics in a given environment) or $\psi_H$, instead fully relying on a data-driven approach.

D. Score-Based Generative Modeling

The goal of score-based generative models is to learn the score function $\psi_H$ at all input points, for a channel distribution $p_H$. When training samples \{${H_i}$\}_i=1^N and their corresponding scores \{${\psi_H(H_i)}$\}_i=1^N are available, explicit score matching uses the following loss function to train a model $s_\theta$ [42]:

$$L_{\text{ESM}, p_H}(\theta) = \mathbb{E}_{H \sim p_H} \left[ ||s_\theta(H) - \psi_H(H)||_2^2 \right].$$

The above can be minimized using any gradient descent method. However, in general, $\psi_H$ is intractable, and thus the loss objective in (7) cannot be used to learn a model $s_\theta$. Furthermore, if $\psi_H$ were tractable and computationally feasible to evaluate, a deep learning model that approximates it would have limited practical use. To circumvent this problem, the work in [42] proposes to use denoising score matching by synthesizing corrupted data samples $\tilde{H}$ and learning the score of the conditional distribution $p_{H|\tilde{H}}$, using the following objective:

$$L_{\text{DSM}}(\theta) = \mathbb{E}_{H \sim p_H, \tilde{H} \sim p_{\tilde{H}}}[||s_\theta(\tilde{H}) - \nabla \log p_{H|\tilde{H}}(H|\tilde{H})||_2^2].$$

The work in [42] proves the following theorem:

Theorem 1 (Appendix from [42]): Assuming that $\log p_{H|\tilde{H}}(H|\tilde{H})$ is differentiable with respect to $H$, then the losses $L_{\text{ESM}, p_H}$ and $L_{\text{DSM}}$ are equivalent.

Theorem 1 implies that the score of the perturbed distribution $p_{\tilde{H}}$ can be learned by using the objective in (8), as long as its probability distribution is differentiable. In particular, this allows the use of arbitrary noise distributions for training, and learning the score at arbitrarily perturbed inputs, when using a continuum of noise levels. When the perturbation $Z$ is i.i.d. Gaussian, with zero mean and covariance matrix $\sigma^2 I$, we obtain that [42]:

$$\nabla \log p_{\tilde{H}|H}(H|\tilde{H}) = -Z_i/\sigma_{2i}^2.$$ (9)

The work in [18] proposes to train a single score-based model using a weighted version of the loss in (8) at multiple noise levels, as well as a learnable model (in practice, a deep neural network) $s_\theta$ with parameters (weights) $\theta$. The loss function for this model is given by [18]:

$$L_{\text{score}}(\theta) = \mathbb{E}_{\tilde{H}, H \sim p_H, Z \sim p_Z}[\frac{\sigma_{2j}^2}{\sigma_{2i}^2} ||s_\theta(H + Z_j) + Z_j/\sigma_{2j}^2||_1^2].$$ (10)

The decision to weigh the predicted score at each noise level comes from formulating denoising score-matching as a variance-exploding (VE) diffusion process [36]. As $-Z_i/\sigma_{2i}^2$ tends towards infinity magnitude for small $\sigma$, weighting with $\sigma^2$ compensates for this and stabilizes learning. Finally, the learned score function $\psi_{H|\tilde{H}}$ is used to draw samples from the posterior distribution in the sequel.

E. Posterior Sampling Using Score Functions

The formulation in Section II-D does not require any assumptions on $P$, $Y$ or $\sigma_{\text{pilot}}$, and makes learning the score function an unsupervised task. To perform channel estimation with a learned model, we use posterior sampling via annealed Langevin dynamics, with the update given by:

$$H \leftarrow H + \alpha \cdot \psi_{H|Y}(H|Y) + \beta \cdot \zeta,$$ (11)

where $\alpha$ and $\beta$ are decaying step sizes, potentially different than the ones in (5). Using Bayes rule for $p_{H|Y}(H|Y) = \frac{p_Y(Y|H)p_H(H)}{p_Y(Y)}$ and expanding the logarithm yields:

$$\log p_{H|Y}(Y|H) = \log p_Y(Y|H) + \log p_H(H) - \log p_Y(Y).$$ (12)

Taking the gradient with respect to $H$ on both sides, we have that $\nabla \log p_Y(Y) = 0$ for all $Y$, and $\psi_{H|Y}(Y|H) = \psi_{Y|H}(Y|H) + \psi_H(H)$. Replacing $\psi_{H|Y}(Y|H)$ in (11) yields:

$$H \leftarrow H + \alpha \cdot \psi_{Y|H}(Y|H) + \alpha \cdot \psi_H(H) + \beta \cdot \zeta.$$ (13)
Compared to (5), the additional term $\alpha \cdot \psi_{Y|H}$ updates the current estimate of $H$ in a direction where it becomes more consistent with the received pilots $Y$. Taken together, the three updates in (13) represent the core routine of the proposed channel estimation algorithm.

III. METHODS

There are two optimization problems that must be solved in order to estimate channels using score-based generative models. These are mapped to training and inference stages, respectively. The decoupling of these two stages is a key difference compared to supervised channel estimation methods, where information about pilots is assumed during training:

1) During the training stage, a score-based generative model $s_\theta$ is trained by minimizing the loss function described in Section III-A. This stage only takes place once in the lifetime of a wireless device, e.g., offline, using a powerful computational server and a dataset of accurate channel measurements or simulated channel realizations.

2) During the testing (inference) stage, channel estimation is formulated as an optimization problem and the iterative algorithm in Section III-B is used to solve it. This stage uses the pretrained score-based model in conjunction with the received pilots to recover CSI. The formulation in Algorithm 1 is independent from the first stage, and could accommodate other impairments such as interference scenarios or few-bit quantization of the received pilots.

A. Training a Score-Based Model for Wireless Channels

In practice, a deep neural network with weights $\theta$ is used to learn a model $s_\theta$. We use the RefineNet architecture [43], that operates at multiple resolution levels in parallel, and consists of a series of RefineNet blocks, with residual skips between them and a number of down- and up-sampling operations, respectively. This allows the model to learn structural relations in the CSI at multiple resolution levels, and to efficiently predict the score function. Furthermore, as the model is fully convolutional, it can handle channel matrices of dynamic input size, both during training and testing. The architecture of a RefineNet block is shown in detail in Figure 3. Further details about the architectural details of RefineNet are available in the original paper [43], as well as in our source code repository.

The loss used to train $s_\theta$ with a batch size of $B$ samples is given by the finite-sample version of (10) as:

$$
L_{\text{train}}(\theta) = \frac{1}{B} \sum_{i=1}^{B} \sigma^2_{z_i} \left\| s_\theta(H_i + Z_i) + \frac{Z_i}{\sigma^2_{z_i}} \right\|^2_2, \quad (14)
$$

where, at each training step, and for each sample in the batch we sample $Z_i \sim \mathcal{CN}(0, \sigma^2_{z_i} I)$, where $\sigma_{z_i}$ is a noise level selected uniformly at random from a set of values $\{\sigma_{z_i}\}_{i=1,...,L}$. That is, for each sample in a batch, noise at a randomly chosen noise level is added to the clean CSI matrix and the model is trained to predict the scaled negative noise that points away from the noisy sample $H_i$ and towards the clean sample $H_i$: this represents the score of the distribution of perturbed wireless channels. Figure 1 illustrates this effect for a perturbed sample, starting from the ideal channel matrix $H_i$, and in which $H_i$ is used as input to the deep neural network trained to predict $-Z_i/\sigma^2_{z_i}$. Finally, note that (14) does not use any information about (2), the power $\sigma^2_{\text{pilot}}$ of the noise affecting the received pilots, or the pilot matrix $P$ itself. The majority of existing deep learning approaches for MIMO channel estimation are supervised and use this information explicitly during training, which leads to overfitting to a specific measurement or noise distribution. In contrast, score-based generative modeling is unsupervised and does not require this information for training, making inference robust and usable across a wide range of SNR values and number of pilot symbols.

B. MIMO Channel Estimation via Posterior Sampling

To perform channel estimation using score-based models, we resort to posterior sampling: our solution comes in the form of a single sample from the posterior distribution $p_{H|Y}(\cdot|Y)$. To sample from the posterior, we use annealed Langevin dynamics [18]. This is an iterative algorithm, that takes the following form at the $i$-th step:

$$
\begin{align*}
H_{\text{est},i+1} & = H_{\text{est},i} + \alpha_i \cdot \nabla \log p_{H|Y}(H_{\text{est},i}|Y) \\
& \quad + \sqrt{2\beta_i} \cdot \alpha_i \cdot \sigma_{z_i} \cdot \zeta_i,
\end{align*}
$$

(15)
where $\zeta \sim \mathcal{CN}(0, \mathbf{I})$ is randomly sampled at every update step, and $\alpha_1 = \alpha_0 + r^t$. The scalars $\alpha_0$, $\beta$ and $r$ are hyper-parameters discussed in Section V. Using the expansion in (13) yields:

$$H_{\text{est}, i+1} = H_{\text{est}, i} + \alpha_i \cdot \left( \nabla \log p_Y(Y | H_{\text{est}, i}) + \nabla \log p_H(H_{\text{est}, i}) \right) + 2\beta \cdot \alpha_0 \cdot \sigma_{z_i} \cdot \zeta,$$  

(16)

where the term involving $\log p_Y(Y)$ does not depend on the current $H_{\text{est}, i}$ and has zero gradient. Using (2) and the assumption that the pilot noise $N$ is Gaussian, we obtain that the conditional distribution of $Y | H$ is Gaussian with mean $H_{\text{est}, i}^T \mathbf{P}$ and covariance matrix $\sigma_{\text{pilot}}^2 \mathbf{I}$, and its score can be derived in closed-form as:

$$\nabla \log p_Y(Y | H_{\text{est}, i}) = \frac{(H_{\text{est}, i}^T \mathbf{P} - Y) \mathbf{P}^H}{\sigma_{\text{pilot}}^2}.$$  

(17)

In practice, we also include an annealing term in the denominator of the above, as shown in Algorithm 1. Finally, the only unknown component in (16) is now the score of $p_H(H)$, evaluated at the current estimate. While this does not have a closed-form expression, we can leverage the score-based model $s_\theta$ trained using the procedure described in Section III-A. This leads to the final channel estimation procedure in Algorithm 1 and illustrated in Figure 2, where, additionally, at each noise level we perform $M = 3$ updates [37]. The update in (16) can also be interpreted as a form of noisy, regularized gradient descent on the pilot consistency loss.

**Algorithm 1 MIMO Channel Estimation With Score-Based Generative Models**

**Inputs:** Pilot matrix $\mathbf{P}$, received pilots $\mathbf{Y}$, pretrained score-based model $s_\theta$, received noise power $\sigma_{\text{pilot}}^2$, inference noise levels $\sigma_{z_i}^2$ (same as what $s_\theta$ was trained with), hyper-parameters $L, \alpha_0, \beta$ and $r < 1$.

**Generate random initial estimate:** $H_{\text{est}} \sim \mathcal{CN}(0, \mathbf{I})$

for $i = 1, \ldots, L$

Set annealed noise level $\sigma = \sigma_{z_i}$.

for $m = 1, \ldots, M$ 

Generate annealing noise $\zeta \sim \mathcal{CN}(0, \mathbf{I})$.

$H_{\text{est}} \leftarrow H_{\text{est}} + \alpha_0 \cdot r^t \cdot (H_{\text{est}, m}^T \mathbf{P} - Y) \mathbf{P}^H + \alpha_0 \cdot r^t \cdot s_\theta(H_{\text{est}}) + 2\beta \cdot \alpha_0 \cdot r^t \cdot \sigma \cdot \zeta$.

**Output:** Estimated channel matrix $H_{\text{est}}$.

**IV. THEORETICAL RESULTS**

An important aspect of using posterior sampling is that previous work has derived theoretical guarantees for recovering the correct estimate (up to the ambient noise level), assuming that a sufficient number of measurements are available [16], [38], even when posterior sampling is performed with respect to a mismatched distribution in terms of the $2$-Wasserstein distance. In this section we derive an expression for the effects of train-test distributional mismatch for a simplified class of tapped delay line SISO channels.

Let $p$ and $r$ be two probability distributions, and $W_2^2(p, r) = \inf_{f: X \sim p, Y \sim r} \mathbb{E} \|X - Y\|_2^2$ be the squared $2$-Wasserstein distance between the two probability distributions [44]. The infimum is taken over pairs of random variables with arbitrary joint distribution and marginals given by $p$ and $r$. If $W_2^2(p, r) > 0$, then the two distributions are mismatched.

We define the following family of sparse channels with complex-valued gains $\{g_i\}_{i=1, \ldots, K}$ and real-valued delays $\{\tau_i\}_{i=1, \ldots, K}$, and the channel gain at an arbitrary delay $t$ given by:

$$h(t) = \sum_{i=1}^{K} g_i \cdot \delta(t - \tau_i).$$  

(18)

To obtain a vector channel, we sample $h(t)$ along the delay dimension, using a pre-determined, fixed waveform $w(t)$ at $N$ equally spaced points as $h[i] = (w(t) * g(t))(iT)$, where $1/T$ is the sampling resolution. In practice, this corresponds to the sampled impulse response of a SISO channel. We make the following additional assumptions:

- $\tau_i = -\alpha_i \log X_i$, where $\alpha_i$ is a constant and $X_i \sim \mathcal{U}(0, 1), \forall i$. This is a realistic model for tap delays, and is used in the standardized CDL models [23].
- $g_i \sim \mathcal{CN}(0, \sigma_i^2)$, $\forall i$, where $\sigma_i$ is a constant.
- $\sum_{i=1}^{K} \sigma_i^2 = 1$, i.e., the taps are normalized to average unit total power.

The above assumptions imply that a distribution of normalized channels $p_H$ is parameterized by the $2K$ real-valued degrees of freedom $\{\sigma_i\}_{i=1, \ldots, K}$ and $\{\alpha_i\}_{i=1, \ldots, K}$. For example, a line-of-sight (LOS) family of channels could exhibit $\alpha_1 < \alpha_i, \forall i > 1$, as well as $\sigma_1 \gg \sigma_i, \forall i > 1$, meaning that the channel profile contains a large magnitude path with a small delay, followed by delayed rays with much lower gains. Note that these assumptions only concern the marginal distributions of the taps and delays, while placing no restrictions on their joint distribution. Let $h_1$ and $h_2$ be two distributions following the previous assumptions, with degrees of freedom $\{\sigma_i^{(1)}, \alpha_i^{(1)}\}_{i=1, \ldots, K}$ and $\{\sigma_i^{(2)}, \alpha_i^{(2)}\}_{i=1, \ldots, K}$, respectively. The main theorem is given below.

**Theorem 1:** Let $h^*$ be a vector channel sampled from $h_2$. Then, posterior sampling with respect to $p_H$, using a number of $\mathcal{O}(1/\delta_{\text{MNR}})$ linear, Gaussian pilot measurements, at a noise level $\sigma_{\text{pilot}}$, recovers $h_{\text{est}}$ such that:

$$\|h^* - h_{\text{est}}\|_2 \leq C\sigma_{\text{pilot}} \; \text{w.p.} \; 1 - \mathcal{O}(\delta_{\text{MNR}}),$$  

(19)

where:

$$\delta_{\text{MNR}}^2 = \frac{W_2^2(h_1, h_2)}{\sigma_{\text{pilot}}^2} \leq \sum_{i=1}^{K} \frac{(\sigma_i^{(1)} - \sigma_i^{(2)})^2 + 2(\alpha_i^{(1)} - \alpha_i^{(2)})^2}{\sigma_{\text{pilot}}^2},$$  

(20)

and $C$ is a constant.

**Proof:** For the complete proof, see the Appendix. At a high-level, it involves two stages, in which the main novelty lies in the second stage: applying Theorem 1.1 from [38] for the $2$-Wasserstein distance, followed by deriving an upper bound for $\delta_{\text{MNR}}^2$ given the previous assumptions. \(\square\)

The quantity $\delta_{\text{MNR}}^2$ represents the mismatch-to-noise ratio, i.e., the ratio between the distributional mismatch of $h_1$ and $h_2$, and the pilot noise power. In general, $\delta_{\text{MNR}}^2$ is a function
of the two distributions. Theorem 1 states that the probability of successful channel estimation with posterior sampling is inversely proportional to $\delta_{\text{MNR}}$, with a smaller $\delta_{\text{MNR}}$ leading to higher probability of correct channel estimation. In general, this happens in two conditions: a very small mismatch $\mathcal{W}_2^2(h_1, h_2)$ exists between the train and test distributions, or the noise power $\sigma_p^2$ is very large. The above leads to the following performance analysis for channel estimation with posterior sampling:

(i) When there is a train-test match, we have $\mathcal{W}_2^2(h_1, h_2) = 0$ and channel estimation with posterior sampling is optimal up to the noise level if using sufficient measurements.

(ii) The probability of successful channel estimation decreases as the distributional distance between the training and test environments increases, at a fixed noise level $\sigma_p^2$.

(iii) Assuming a fixed $\mathcal{W}_2^2(h_1, h_2)$, the probability of successful channel estimation increases as the ambient noise level $\sigma_p^2$ increases. While this implies favourable scaling in very low SNR, the score-based estimator in this regime is limited by the large noise power.

**Applicability to MIMO Channels.** While the theory is derived for vector channels, the implications of Theorem 1 are verified in Section V for MIMO channels, and are successful in predicting the performance of the proposed approach in varying test-time environments. In particular, we verify that channel estimation with posterior sampling, under moderate distributional shifts and with realistic, non-Gaussian transmitted pilot matrices verifies the three behaviours outlined in the previous paragraph.

As an alternative to delay-domain SISO channels, it is possible to interpret a vector channel as either a single-input multiple-output (SIMO) or multiple-input single-output (MISO) channel sampled along the spatial angular direction, using an array with a finite number of elements. As long as the distributions of the angles of arrival and departure allow for tractable computation of the 2-Wasserstein distance, a similar result could be derived for MIMO channels.

**V. Simulation Results and Discussion**

We evaluate Algorithm 1 in simulated settings, across a wide range of SNR values, using channel estimation fidelity, end-to-end error rates in a simulated coded system and training and inference complexity to draw conclusions. To emulate deployment in a novel wireless propagation environment, Sections V-C and V-D consider scenarios where models are tested on different distributions than the training one, without any prior knowledge about the test distribution or adaptation, and without any additional retraining.

**A. Data and Training**

We use the CDL family of channel models to generate training, validation and test data. CDL-D channels are LOS, and generally the easiest to estimate due to their very sparse structure in the beamspace representation. CDL-B and CDL-C channels are NLOS, while CDL-A channels have both components. For training, we use 10000 channel realizations from a specific CDL model. To generate training samples, we increment the seed of the CDL generator, and pick the first subcarrier of the first symbol in each generated channel. Details about the used parameters are given in Table I. For tuning the hyper-parameter of CS methods we use a set of 100 channel realization from the training distribution. Exact details on the training parameters (optimizer, learning rate, batch size) are available in the source code repository.
For testing, we generate a new set of 100 channel realizations from each target distribution, using different random seeding than training and validation. For pilots $P$, we use matrices of size $N_t \times N_p$ with randomly chosen QPSK elements (unit-power, two-bit phase-quantized random beamforming). We normalize all channels using the average channel power from the training set taken across all training samples and entries, and define the average SNR as $\frac{N_t}{\sigma^2_{\text{pilot}}}$.

**B. Baselines**

The following baselines are evaluated and are available in the source code repository:

- **Maximum Likelihood (ML)** [45]: This approach does not assume any prior information about the structure of $H$ and aims to maximize the log-likelihood $p(Y|H)$. Using knowledge that the noise $N$ in (1) is Gaussian with zero mean and known variance $\sigma^2_{\text{pilot}}$, and that the pilot entries are chosen i.i.d., yields the closed form solution via the regularized pseudo-inverse as [45]: $H_{\text{ML}} = Y^H (PP^H + \sigma^2_{\text{pilot}} I)^{-1}$.

- **Lasso** [24]: This is a CS-based approach that uses $\ell_1$-norm element-wise regularization in the two-dimensional Fourier (beamspace) domain. Channel estimation is formulated as the solution to the following optimization problem:

$$\arg\min_{H} \frac{1}{2} \| Y - HP \|_F^2 + \lambda \left\| F_{\text{left}} H F_{\text{right}}^H \right\|_{1,1} , \quad (21)$$

where $F_{\text{left}}$ and $F_{\text{right}}$ are square, discrete Fourier matrices of size $N_t \times N_r$ and $N_r \times N_t$, respectively. We use gradient descent with momentum to solve (21) and we tune the step size, number of optimization steps, and the value of $\lambda$ using the validation set.

- **EM-GM-AMP** [25]: This approach uses a Gaussian mixture (GM) prior in the beamspace domain of $H$, justified by the sparse angular nature of mmWave propagation channels [39]. We use the publicly available Matlab implementation provided in [46] and tune the number of internal expectation maximization (EM) steps, as well as the stopping condition using the validation set.

- **fsAD** [26], [27]: This represents a classical CS-based approach for recovering channel matrices assumed sparse in the continuous spatial frequency domain, similar to Newtonized OMP (Orthogonal Matching Pursuit) [47]. In scenarios with uniform linear arrays (ULA) at both the receiver and transmitter, we use a formulation that assumes sparsity in the over-sampled beamspace domain [27]:

$$\arg\min_{H} \frac{1}{2} \| Y - HP \|_F^2 + \lambda \left\| W_{\text{left}} H W_{\text{right}}^H \right\|_{1,1} , \quad (22)$$

where $W_{\text{left}}$ and $W_{\text{right}}$ represent $l$ times over-sampled DFT matrices. We use $l = 4$, and tune the parameter $\lambda$, step size and number of steps using the validation set.

- **WGAN** [4]: This represents an unsupervised framework for using generative adversarial networks in channel estimation. A generative model $g_{\theta}$ is first trained to map low-dimensional vectors $z$ to channel matrices $H$, using an adversarial loss and regularization [48]. During inference, a similar optimization problem to ours is formulated and solved iteratively, by inverting the generative model. We tune the hyper-parameters using the validation set.

- **L-DAMP** [3], [35]: This represents a powerful data-driven algorithm that uses deep unrolling and end-to-end learning. We use a denoising convolutional neural network (DnCNN) [49] backbone, and train a separate L-DAMP model for each value of $\alpha$.

- **Approximate MMSE**: By definition, the minimum mean squared error (MMSE) estimator is given by $E[H|Y]$. Because the log-prior for realistic channel models, such as CDL channels, is generally intractable and does not have a closed form expression (e.g., the tap locations themselves are stochastic, which would require additional assumptions and factorization of the prior to become tractable [50]), we leverage score-based models to obtain an empirical upper performance bound by averaging a number of samples from the estimated posterior (given by running Algorithm 1) $p(H|Y)$ [16]. For the remaining simulations, we use 50 posterior samples, each obtained from a different run of Langevin dynamics with different initial estimates.

**C. Robust Estimation Performance**

We train score models on $N = 10000$ samples of CDL-B and CDL-C channels separately, and test them in CDL-{A, B, C, D} environments, without fine-tuning or adaptation. As a reference, we also train a Mixed score-based model, which uses 10000 training samples each from all four CDL-{A, B, C, D} models. We also train score-based models on their matching test-time distribution to evaluate the potential gains of adapting models. In practice, this adaptation would be done at the base station or user equipment, after a training set from an environment is collected. We measure estimation quality using the normalized mean squared error (NMSE), defined as:

$$\text{NMSE} [\text{dB}] = 10 \log_{10} \frac{\| H_{\text{est}} - H \|_F^2}{\| H \|_F^2} . \quad (23)$$

Figure 4 shows estimation results when using models trained on CDL-B channels. In the in-distribution setting score-based models recover channels up to the noise level, and surpass prior work by at least 3 dB in NMSE for SNR values between $-5$ and 30 dB. In all test conditions, the Mixed, in-distribution and approximate MMSE methods scale favourably.

---

**Table I**

PARAMETERS USED TO SIMULATE CDL CHANNEL REALIZATIONS

| Channel Model | Carrier Frequency | Antenna Arrays | Antenna Spacing | $(N_t, N_r)$ |
|---------------|------------------|----------------|----------------|-------------|
| Training: CDL-B, C | 40 GHz | ULA, UPA | $\lambda/2$ - ULA | (16, 64) |
| Testing: CDL-A, B, C, D | | | $(\lambda/4, \lambda/4)$ - UPA | (32, 128) |
| | | | (64, 256) | UPA |
with high SNR. ML, EM-GM-AMP and Lasso are competitive in the very low SNR regime, but saturate performance quickly. This makes score-based models an attractive candidate for scenarios where very accurate channel estimation is desired, such as fixed, broadband wireless access. The slight drop in performance between the Mixed model and the model trained only on CDL-B channels in Figure 4(b) is owed to the finite capacity of the deep network, performing slightly poorer on a specific distribution when data from other distributions is included in training.

The WGAN approach can competitively estimate channels in the very low SNR regime, but, in general, suffers from saturation of the performance at SNR values greater than −5 dB. This has been previously observed in [29] and is owed to the sub-optimal approach (in practice, using an Adam [51] optimizer) used to invert the latent representation, whereas a score-based model using Langevin dynamics converges to a near-optimal solution. Estimating CDL-A channels at a pilot density $\alpha = 0.6$ using a score-based model trained on CDL-B is difficult, as shown in Figure 4(a). Because of CDL-A channels including both LOS and non-LOS components, this leads to a large distributional distance between CDL-A and all other models, where a subset of propagation paths will always be missing. In accordance with Theorem 1, this causes an error floor in the high SNR regime. Figure 5 plots estimation performance with models trained only on CDL-C channels, and tested on all four CDL models. This generally corroborates the findings from Figure 4: score-based models recover channels up to the noise floor in-distribution (Figure 5(c)), and outperform all baselines for SNR values between −5 and 30 dB. A notable difference here is the much better generalization capability of score-based models from CDL-C to CDL-D environments, as observed in Figure 5(d). This is explained by the fact that CDL-C channels contain the least amount of scattering, and are most similar to CDL-D channels, leading to a lower $\delta_{SNR}(CDL-C, CDL-D)$, in line with the Theorem 1. Another difference that occurs when training on CDL-C channels, is that generalization to CDL-A models is also improved, matching the performance of the Lasso algorithm, as shown in Figure 5(a).

In both settings of Figure 4 and Figure 5, we also find that L-DAMP has similar scaling in terms of SNR, and is a reliable estimation approach, especially for the simpler CDL-C and CDL-D models, as highlighted in Figures 4(c) and 4(d).
Fig. 5. Channel estimation performance using methods trained/tuned on CDL-C channels in a 16 \times 64 \text{mmWave} MIMO scenario. Each of the four sub-figures shows performance in a different CDL environment, when \( \alpha = 0.6 \) (38 pilot vectors).

The Lasso and EM-GM-AMP approaches are also competitive, in general, and surpass score-based models (trained on CDL-B), WGAN, and L-DAMP when evaluated on CDL-A channels, as shown in Figure 4(a).

The findings in this section support the conclusions of Theorem 1:

(i) Posterior sampling with score-based models can recover channels up to the noise floor when the test distribution matches the training and the SNR is above 0 dB.

(ii) Channel estimation with posterior sampling outperforms the baselines in the low SNR regime (\(-5\) to 0 dB), under test-time distributional shifts.

(iii) Test channel distributions that are significantly more different than the training distributions lead to error floors in the high SNR regime (Figures 4(a) and 5(a)).

Finally, it is worth highlighting that CS approaches are overall robust and retain performance under distributional shifts, regardless of what environment the hyper-parameters are tuned on. While these algorithms are useful in idealized settings, the following limitations remain:

- When deriving the approximation in (22), we use knowledge about the shapes of the antenna arrays at the receiver and transmitter. In general settings, this information may not be available, or the array response may be intractable (e.g., for non-uniform arrays). In contrast, score-based generative models learn the distribution of the channels while implicitly learning the array configuration, with no external knowledge required.

- A key assumption made by CS approaches is that channels are exactly sparse (on a discrete grid for Lasso and EM-GM-AMP, or on the continuum of spatial frequencies for fSAD). While this holds for simulated channels using the CDL models, it is generally not true in practice, e.g., as shown in channel sounding experiments at mmWave [22], [28]. Score-based models are completely data-driven, with no assumptions required.

D. End-to-End Coded Performance

To consider a realistic performance metric, in this section we evaluate end-to-end coded bit error rates in a simulated downlink physical layer flow of cellular communication systems. A block diagram of the system is shown in Figure 7. The transmitter (base station) sends a pilot matrix \( \mathbf{P} \) across multiple channel uses, which is received by the user as \( \mathbf{Y} \),
Fig. 6. End-to-end coded bit error rate as a function of SNR in CDL-D out-of-distribution channels, with $\alpha = 0.6$. Methods are trained (Score-based, L-DAMP) or tuned (fsAD, Lasso) exclusively on CDL-C channels, and tested on CDL-D using: (a) 16-QAM and (b) 64-QAM data modulation.

using the model in (2). Channel estimation is used to obtain $H_{\text{est}}$. For the data transmission, a payload of 324 bits is encoded using a rate 1/2 low-density parity-check (LDPC) code, followed by digital modulation. Symbols are split in groups of $N_s = 4$ data streams, followed by adaptive digital beamforming with the unit-power $V_{\text{est}}$ matrix to obtain a matrix of 64-dimensional column vectors. We assume that the channel is symmetric (e.g., time-division duplex), and that the transmitter obtains noiseless feedback about the estimated channel. The adaptive transmitter beamforming block uses the first four columns of $V_{\text{est}}$ from the singular value decomposition of $H_{\text{est}}$ as beamforming weights. The data symbols are observed at the receiver as $Y_d = H V_{\text{est}}$ and are succeeded by linear MMSE detection to yield the equalized data symbols:

$$X_{d,\text{eq}} = (H_{\text{est}}^H H_{\text{est}} + \sigma_{\text{pilot}}^2 I)^{-1} H_{\text{est}}^H Y_d.$$ (24)

The above is followed by entry-wise soft de-mapping, reshaping to a soft codeword, and LDPC decoding, to obtain the decoded bit stream $b_{\text{est}}$. This is repeated for two million codewords, each using a different CDL-D channel realization from a randomly generated test set, while using models trained on the CDL-C distribution for channel estimation, matching the setting of Figure 5(d). Figure 6 plots the performance results for four methods, as well as communication using ideally known channels, measured in terms of the energy per bit to noise ratio ($E_b/N_0$) of the corresponding channel estimation SNR. We omit WGAN, ML and EM-GM-AMP from this investigation due to their estimation error floor, which cannot support packet communication at the considered coding rate. We also omit the approximate MMSE due to its large computation complexity, and instead use ideal channel knowledge as an upper performance bound.

Figure 6(a) uses 16-QAM modulation. Even in this case, channel estimation errors lead to an error floor for the Lasso algorithm, while score-based models, fsAD, and L-DAMP manage to overcome this and decay the bit error rate at high $E_b/N_0$. In this case, an NMSE between $-20$ dB and $-15$ dB is required to avoid this error floor, that Lasso and other simple CS methods cannot achieve, as per Figure 5(d). Figure 6(a) also shows that both score-based models and fsAD are competitive when compared to L-DAMP: the same coded bit error rate can be achieved with an $E_b/N_0$ that is 5 dB lower than L-DAMP. In Figure 6(b), the modulation is 64-QAM and accurate channel estimation is required for equalization and precoding. The L-DAMP approach is not enough to avoid an end-to-end error floor, while score-based models and fsAD overcome this floor and decay the bit error rate at approximately the same rate as ideal channel knowledge. Finally, the benefits of extremely accurate channel estimation at high SNR values (e.g., estimation NMSE lower than $-30$ dB at pilot SNR larger than 25 dB) are also illustrated in Figure 6(b), where it can be seen that both methods depart from optimal performance, and score-based models improve performance by up to 0.5 dB in $E_b/N_0$. Overall, these results highlight the importance of accurate channel estimation and how score-based models are competitive for this purpose and trade off increased computational complexity for accuracy.

E. Scaling to Large Channel Sizes

We verify that channels can be estimated without error floors using a small score-based model, even for large channel
sizes. Figure 8(a) shows the results of the experiment when training and testing score-based models for three channel sizes, where for each size we train a separate model. It can be seen that, given a fixed pilot overhead \( \alpha \), there is a slight drop in performance at larger channel sizes – we attribute this to using a score-based model of the same size (depth \( D = 6 \), width \( W = 12 \)), regardless of channel size. To compare performance under a resource-limited scenario, we simulate larger sizes, where only 38 pilot vectors are allowed, leading to low values of \( \alpha \). In this case, estimation for larger sizes is competitive given sufficient pilots, but fails for \( 64 \times 256 \) with \( \alpha = 0.15 \).

Figure 8(b) investigates in more detail how performance scales with \( \alpha \) for all three sizes in the high SNR regime, and the same model size as in Figure 8(a): here, we consistently find that there is a breaking point for \( \alpha \approx 0.25 \), below which the estimation error increases rapidly. This is indicative of not having sufficient measurements to meet the conditions of Theorem 1.

### F. Complexity Analysis and Ablation

All RefineNet with \( W = 24 \) hidden channels in the first layer, and a depth \( D \) of six residual blocks in both the encoder and decoder paths. In Table II, we investigate validation performance at SNR 28 dB for nine model sizes that vary in depth and width. We chose this value as it corresponds to the high SNR regime, where score-based models have NMSE lower than \(-30 \) dB and performance is bounded by the quality of the learned prior, as indicated by Theorem 1. Our findings indicate that model performance is much more sensitive to depth rather than width, but even a shallow model (four blocks), has a performance loss of at most \( 1.5 \) dB. Shallower models improve performance in the out-of-distribution setting, indicating that they overfit less. Table II also measures per-step latency on a machine with an NVIDIA A100 GPU and Intel Xeon 6230 CPU, as well as the optimal number of inference steps at two SNR values and \( \alpha = 0.6 \). We find that depth is a main contributor to latency, whereas width is less impactful, due to the powerful parallelization capabilities of the GPU.

Figure 9(a) plots the convergence of the training loss in (14) for score-based models trained on all channel distributions (note that the optimal value of this loss is not zero [37]). It can be noticed that in all cases, convergence is stable and achieved in at most \( 60 \) epochs using \( N = 10000 \) training samples, with approximately one minute per epoch of training on the GPU. This is also a strong indicator that adapting a pre-trained score-based model would converge in at most the same time, and achieve near optimal performance in as few as ten epochs, given a large training set is available.

Figure 9(b) plots the convergence of the Langevin dynamics updates in (16) as a function of the number of steps, SNR, and the hyper-parameter \( \beta \), averaged over the validation set. It can be seen that convergence is stable for all SNR levels, and that robust early stopping points across all validation samples can
be found – these benefit both complexity (reduced number of inference steps and up to $6 \times$ reduced inference complexity) and performance. In general, lowering $\beta$ converges faster and to a better solution, but is also more sensitive to the stopping criterion. In all cases, the estimation loss during inference exhibits two stages: a fast convergence stage (in the first 200 steps, where an estimate with an error less than $-10$ dB can be obtained for all SNR values) in which the residual errors are non-Gaussian and the updates lead to higher density regions of the distribution, and a slower convergence stage, where finer structural details of the estimated channel are captured. Table III evaluates the computational complexity of score-based generative models, approximate MMSE, and the compressed sensing baselines. It can be seen that inference latency is comparable to fsAD, and much lower than that of approximate MMSE and EM-GM-AMP in large MIMO scenarios. For EM-GM-AMP, we do not include exact FLOP counts and memory footprints due to lack of support in the MATLAB profiler. The FLOP count of score-based models scales much more favourably compared to fsAD due to being a purely convolutional algorithm, and has a performance loss smaller than 0.1 dB in-distribution compared to the approximate MMSE algorithm, while requiring four times fewer FLOPs.

G. Performance in Interference Channels

We test estimation performance using pretrained score-based models on interference limited scenarios, which can become the primary bottleneck in mmWave cellular systems with dense deployments [52]. The system equation in (1) takes the form [53]:

$$Y = HP + \gamma H_I P_I + N,$$

where $H_I$ and $P_I$ are the interference channels and transmitted pilots, respectively, and $\gamma$ represents the power of the interference signal.

We investigate a $64 \times 16$ MIMO scenario where $H$ is sampled from CDL-C channels, and a score-based model trained on CDL-C models is used. $H_I$ and $P_I$ are sampled independently of $H$ and $P$, respectively, with $H_I$ sampled from either CDL-C or CDL-B channels. The value of $\gamma$ is varied from $-10$ dB to $-30$ dB. When running annealed Langevin dynamics, no knowledge about the distribution of interference or $\gamma$ is assumed, and the interference is treated as noise. The results in Figure 9(b) show that estimation performance is reliable in noise-limited scenarios (without departing from the performance in Figure 5(c)) and that score-based models can effectively accommodate non-Gaussian noise scenarios. At higher SNR values, the performance becomes interference...
limited, with the performance floor improving relative to $\gamma$
by at least 2 dB, experimentally showing that part of the
interference signal can be reliably eliminated using score-

based models.

VI. CONCLUSION
In this paper, we have introduced an unsupervised, prob-
abilistic approach for MIMO channel estimation using a
reduced number of pilot symbols. The approach leverages
score-based generative models to perform posterior sampling,
and represents a new research direction for MIMO channel
estimation. Our results on simulated mmWave channels show
that performance is favourable in-distribution, as well as in
out-of-distribution settings, where channel distributions not
seen during training are tested. Additionally, compared to prior
work, estimation with posterior sampling has the advantage of
avoiding error floors and achieving high-quality estimation in
the high SNR regime, which leads to improved end-to-end
performance, while trading off complexity.

A current limitation of score-based models is the high
inference complexity of posterior sampling with Langevin
dynamics. In Section V-F we have performed an initial inves-
tigation into improvements achievable through architectural
modifications (depth and width of the deep score-based model)
which was able to reduce inference latency down to 1.5 sec-
onds for channels of size $16 \times 64$. This is usable in scenarios
with low mobility, where long-term distributional shifts may
still occur, such as fixed access mmWave, reflective intelligent
surfaces, or mmWave backhaul. Based on the fast convergence
of training in ten epochs when learning from randomly initial-
ized models, future work that considers adapting pre-trained
models to new environments using a limited amount of training
channels is a promising direction. Recent work [54] has
investigated algorithmic improvements to posterior sampling
by choosing the hyper-parameters during inference to trade-off
performance for latency. This a promising direction of future
research that we aim to explore, alongside others such as lever-
aging pretrained score-based models for channel estimation in
few-bit quantization receivers and overcoming error floors in
interference scenarios.

APPENDIX

Theorem 2 (Theorem 1.1 from [38]): Let $h_1$ be a
high-dimensional arbitrary distribution over an $\ell_2$ ball
of radius $r$, and let $h^*$ be drawn from $h_1$. Let $h_2$ be any
distribution on the same probability space as $h_1$, such that

$$\mathcal{W}_2(h_1, h_2) \leq \sigma \delta^{1/2}. \quad (26)$$

Suppose there exists an algorithm that recovers an estimate $\hat{h}$
of $h^*$ using an arbitrary measurement matrix $A$ that gives $n$
measurements, and noise power $\sigma^2$, such that $\|h^* - \hat{h}\|_2 \leq \sigma$ with probability $1 - O(\delta)$. Then, posterior sampling with
respect to $h_2$ and with $n' \geq \Omega(n \log(1 + \frac{\alpha r^2\|A\|_2^2}{\sigma^2}) + \log 1/\delta)$
oisy, Gaussian measurements at noise level $\sigma$ will output $h_{est}$
such that:

$$\|h^* - h_{est}\|_2 \leq \sigma \quad \text{w.p.} \quad 1 - O(\delta). \quad (27)$$

Proof: We refer readers to [38] for a complete proof, omitted here due to lack of space.

From (26), there exists a unique value $\delta_{\min}$ that is a function
of $h_1$ and $h_2$, such that:

$$\delta_{\min}^2(h_1, h_2) = \frac{\mathcal{W}_2^2(h_1, h_2)}{\sigma_{\text{pilot}}}.$$ \quad (28)

Let $\delta_{\text{SNR}}(h_1, h_2) = \delta_{\min}(h_1, h_2)$. We have the following
lemmas.

Lemma 1 (Section 1.2, Page 4 From [44]): Let $x$ and $y$ be
two random variables with marginal distributions $p_x$ and $p_y$,
respectively, and arbitrary joint distribution. Let $a$ and $b$ be
two independent random variables with the same marginals.
We have that:

$$\mathcal{W}_2^2(x, y) \leq \mathcal{W}_2^2(a, b).$$

Lemma 2 (Section 1.2, Page 7 From [44]): For any two scalar random variables $a$ and $b$
with cumulative distribution functions $F_a$ and $F_b$, respectively, we have that:

$$\mathcal{W}_2^2(a, b) = \int_0^1 |F_a^{-1}(u) - F_b^{-1}(u)|^2 \, du.$$

Lemma 3 (Lemma 1 From [55]): For any two random vec-
tors $x$ and $y$ with mutually independent entries, we have that:

$$\mathcal{W}_2^2(x, y) = \sum_i \mathcal{W}_2^2(x_i, y_i).$$

Using Lemma 2 we compute the closed-form 2-Wasserstein
distance between any two scalar random variables. This leads
to the following corollaries.

Corollary 1: Let $h_1 \sim \mathcal{C}(0, \sigma_1^2)$ and $h_2 \sim \mathcal{C}(0, \sigma_2^2)$ be
two independent, complex, zero-mean and circularly symmet-
metric Gaussian random variables. We have that:

$$\mathcal{W}_2^2(h_1, h_2) = (\sigma_1 - \sigma_2)^2.$$

Proof: Let $h_i = x_i + jy_i$ for $i = 1, 2$. Using Lemma 2
we have that:

$$\mathcal{W}_2^2(h_1, h_2) = \inf_{\psi_1 \sim h_1, \psi_2 \sim h_2} \mathbb{E} \left( \|\psi_1 - \psi_2\|_2^2 \right)$$

$$= \inf_{\psi_1 \sim h_1, \psi_2 \sim h_2} \mathbb{E} \left( \|\text{Re}\{\psi_1\} - \text{Re}\{\psi_2\}\|_2^2 \right)$$

$$+ \mathbb{E} \left( \|\text{Im}\{\psi_1\} - \text{Im}\{\psi_2\}\|_2^2 \right)$$

$$= \mathcal{W}_2^2(x_1, x_2) + \mathcal{W}_2^2(y_1, y_2)$$

$$= \frac{(\sigma_1 - \sigma_2)^2}{2} + \frac{(\sigma_1 - \sigma_2)^2}{2} = (\sigma_1 - \sigma_2)^2.$$

Corollary 2: Let $\tau_1 = -\alpha_1 \log X_1$ and $\tau_2 = -\alpha_2 \log X_2$,
where $X_1$ and $X_2$ are i.i.d. and distributed as $\mathcal{U}(0, 1)$. Then:

$$\mathcal{W}_2^2(\tau_1, \tau_2) = 2 (\alpha_1 - \alpha_2)^2.$$

Proof: Using the definition of the cumulative distribution function, we have:

$$F_{\tau}(u) = P(\tau \leq u) = P(-\alpha \log X \leq u)$$

$$= P \left( \log X \geq \frac{-u}{\alpha} \right) \leq P \left( X \geq \exp \left( \frac{-u}{\alpha} \right) \right)$$

$$= 1 - \exp \left( -\frac{u}{\alpha} \right).$$
where we use that $X \sim U(0,1)$ satisfies $F_X(u) = u$ for all $u \in [0,1]$. We have that $F_{\tau^{-1}}(v) = -\alpha \log (1-v)$. Using Lemma 2:

\[
\mathcal{V}_2^2 (\tau_1, \tau_2) = \int_0^1 |F_{\tau^{-1}}(v) - F_{\tau_2^{-1}}(v)|^2 dv \\
= \int_0^1 -\alpha_1 \log (1-v) + \alpha_2 \log (1-v) |^2 dv \\
= (\alpha_1 - \alpha_2)^2 \int_0^1 \log^2 (1-v) dv \\
= (\alpha_1 - \alpha_2)^2 \int_0^1 \log^2 (t) dt,
\]

where the last equality is by change of variable $t = 1 - v$. Using that $\int_0^1 \log^2 (t) dt = (t \log (t))^2 - 2 t \log |t| + 2t|^1_0 = 2$, the proof is completed. \(\square\)

Combining Corollary 1, Corollary 2, and Lemma 3, we obtain that:

\[
\delta_{\text{MINR}}(a, b) = \sum_{i=1}^K \left( (\sigma_i^{(1)} - \sigma_i^{(2)})^2 + 2(\alpha_i^{(1)} - \alpha_i^{(2)})^2 / \sigma_{\text{pilot}}^2 \right),
\]

to two channel distributions $a$ and $b$ that satisfy the assumptions as $h_1$ and $h_2$ in Section IV, as well as an additional mutual independence between all tap location and delays. Using Theorem 2 and Lemma 1, we obtain that $\delta_{\text{MINR}}(h_1, h_2) \leq \delta_{\text{MINR}}(a, b)$ and Theorem 1 is proved.

### References

[1] H. Viswanathan and P. E. Mogensen, “Communications in the 6G era,” *IEEE Access*, vol. 8, pp. 57063–57074, 2020.

[2] Y. Heng et al., “Six key challenges for beam management in 5.5G and 6G systems,” *IEEE Commun. Mag.*, vol. 59, no. 7, pp. 74–79, Jul. 2021.

[3] H. He, C.-K. Wen, S. Jin, and G. Y. Li, “Deep-learning-based channel estimation for beamspace mmWave massive MIMO systems,” *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 852–855, Oct. 2018.

[4] E. Balevi, A. Doshi, A. Jalal, A. Dimakis, and J. G. Andrews, “High dimensional channel estimation using deep generative networks,” *IEEE J. Sel. Areas Commun.*, vol. 39, no. 1, pp. 18–30, Jan. 2021.

[5] W. Hong et al., “The role of millimeter-wave technologies in 5G/6G wireless communications,” *IEEE J. Microw.*, vol. 1, no. 1, pp. 101–122, Jan. 2021.

[6] N. Shlezinger, G. C. Alexandropoulos, M. F. Imani, Y. C. Eldar, and D. R. Smith, “Dynamic metasurface antennas for 6G extreme massive MIMO communications,” *IEEE Wireless Commun.*, vol. 28, no. 2, pp. 106–113, Apr. 2021.

[7] B. Zheng, C. You, W. Mei, and R. Zhang, “A survey on channel estimation and practical passive beamforming design for intelligent reflecting surface aided wireless communications,” *IEEE Commun. Surveys Tuts.*, vol. 24, no. 2, pp. 1035–1071, 2nd Quart., 2022.

[8] 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. Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 5000–5004.

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

[10] Y.-C. Liang et al., “Reconfigurable intelligent surfaces for smart wireless environments: Channel estimation, system design and applications in 6G networks,” *Sci. China Inf. Sci.*, vol. 64, no. 10, pp. 1–21, 2021.

[11] K. B. Letaief, W. Chen, Y. Shi, J. Zhang, and Y. A. Zhang, “The roadmap to 6G: AI empowered wireless networks,” *IEEE Commun. Mag.*, vol. 57, no. 8, pp. 84–90, Aug. 2019.

[12] S. Han et al., “Artificial-intelligence-enabled air interface for 6G: Solutions, challenges, and standardization impacts,” *IEEE Commun. Mag.*, vol. 58, no. 10, pp. 73–79, Oct. 2020.

[13] P. W. Koh et al., “WILDS: A benchmark of in-the-wild distribution shifts,” in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 5637–5664.
[37] Y. Song and S. Ermon, “Improved techniques for training score-based generative models,” in Proc. Adv. Neural Inf. Process. Syst., vol. 33, 2020, pp. 12438–12448.

[38] A. Jalal, S. Karmalkar, A. Dimakis, and E. Price, “Instance-optimal compressed sensing via posterior sampling,” in Proc. Int. Conf. Mach. Learn., 2021, pp. 4709–4720.

[39] R. W. Heath, Jr., N. González-Prelcic, S. Rangan, W. Roh, and A. M. Sayeed, “An overview of signal processing techniques for millimeter wave MIMO systems,” IEEE J. Sel. Topics Signal Process., vol. 10, no. 3, pp. 436–453, Apr. 2017.

[40] G. O. Roberts and R. L. Tweedie, “Exponential convergence of Langevin distributions and their discrete approximations,” Bernoulli, vol. 2, pp. 341–363, Dec. 1996.

[45] E. Nayebi and B. D. Rao, “Semi-blind channel estimation for multiuser massive MIMO systems,” IEEE Trans. Signal Process., vol. 66, no. 2, pp. 540–553, Jan. 2018.

[50] D. Neumann, T. Wiese, and W. Utschick, “Learning the MMSE channel estimator,” IEEE Trans. Signal Process., vol. 66, no. 11, pp. 2905–2917, Jun. 2018.

[53] S. Akoum, O. E. Ayach, and R. W. Heath, Jr., “Coverage and capacity in mmWave cellular systems,” in Proc. Conf. Rec. 46th Asilomar Conf. Signals, Syst. Comput. (ASILOMAR), Nov. 2012, pp. 688–692.

[54] T. Salimans and J. Ho, “Progressive distillation for fast sampling of diffusion models,” in Proc. Int. Conf. Learn. Represent., 2022, pp. 1–21.

[55] G. Mordant and J. Segers, “Measuring dependence between random vectors via optimal transport,” J. Multivariate Anal., vol. 189, May 2022, Art. no. 104912.

Marius Arvinte (Member, IEEE) received the B.S. and M.S. degrees in electronics, telecommunications and information technology from the Politehnica University of Bucharest, Romania, in 2014 and 2016, respectively, and the Ph.D. degree in electrical and computer engineering from The University of Texas at Austin, USA, in 2022. Since 2022, he has been a Research Scientist with Intel Corporation, USA. His research interests include applied deep learning for signal processing and inverse problems, with a focus on digital communications.

Jonathan I. Tamir (Member, IEEE) received the B.S. degree in electrical and computer engineering (ECE) from The University of Texas at Austin in 2011 and the Ph.D. degree in electrical engineering and computer sciences from the University of California, Berkeley, in 2018, where he continued as a Research Associate. He joined the faculty of The University of Texas at Austin in 2020 as an Assistant Professor with joint appointments in ECE and the Dell Medical School. His research interests include computational magnetic resonance imaging (MRI), machine learning for inverse problems, and wireless communications.