Physics-based Meta Learning for Channel Transformation

Satyavrat Wagle
Elmore School of Electrical and Computer Engineering, Purdue University
wagles@purdue.edu

Akshay Malhotra, Shahab Hamidi-Rad, Aditya Sant
InterDigital Communications
{firstname.lastname}@interdigital.com

Christopher G. Brinton
Elmore School of Electrical and Computer Engineering, Purdue University
cgb@purdue.edu

Abstract

Adapting machine learning (ML) solutions in the PHY layer to new wireless configurations is hindered by the prohibitive cost of data collection for each unique configuration. While existing methods attempt to transform data from a reference to a target configuration, their reliance on large, hard-to-obtain paired datasets is a significant bottleneck. We propose a physics-based transformation framework that leverages a parametric latent space in which channel transformations can be mapped to relatively simple translations. Our model pre-trains on readily available, unpaired data to learn this space, then fine-tunes with only a fraction of paired labeled data to learn the required translation. We demonstrate the effectiveness of our framework through experiments on different modalities of channel transformation and through improvements in downstream tasks relevant to wireless communication.

1 Introduction

Machine learning (ML) solutions for challenges in wireless communication such as, channel compression, estimation, equalization, and beamforming have seen extensive interest in the past few years Guo et al. [2022], Kassir et al. [2022], Mao et al. [2018], Sant et al. [2022]. A fundamental challenge for machine learning (ML) in the wireless PHY layer is that models are highly sensitive to the specific physical configuration of the environment. An ML model trained for one setup defined by parameters such as the carrier frequency, system bandwidth, or antenna geometry may not be directly usable or often fails to generalize to other settings. This lack of transferability implies that for any change in the operational environment, the entire data collection process must be repeated in order to retrain the model to adapt to the new configuration. This presents a significant practical barrier, as acquiring the necessary over-the-air (OTA) channel data is a manual, time-consuming, and extremely resource-intensive endeavor Ju and Rappaport [2023], Ju et al. [2022], Kumar et al. [2024].

Prior methods address this issue by training black-box deep learning models to directly map a channel matrix from a source configuration to the associated channel in the target configuration Alrabeiah and Alkhateeb [2019], Liang and Li [2024], Liu et al. [2019] . However, such methods require paired channel measurements from both source and target environments for training. This requirement of paired data is often impractical and contradicts the very goal of reducing the burden of data collection.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: AI and ML for Next-Generation Wireless Communications and Networking (AI4NextG).

In contrast, the key insight of this work is that channel transformations can be modeled more efficiently as translations of physical parameters. We introduce a physics-based meta-learning framework that operationalizes this idea. First, the model is pre-trained on abundant, unpaired data from the source configuration to learn an effective mapping from the complex channel space into the structured parameter space. Next, using only a small number of paired samples, the model is fine-tuned to learn the precise, and often sparse, translation from source to target parameters. As this translation is much simpler to learn than a direct, high-dimensional channel transformation, our method drastically reduces the requirement for paired data, providing significant savings in time and resources.

2 System Model and Approach

We consider a wireless communication system with N_t transmit and N_r receive antennas in a Uniform Linear Array (ULA) pattern. For a given carrier frequency f, the associated channel model defined by $\mathcal{M} : \mathbb{R}^{4P} \rightarrow \mathbb{C}^{N_t \times N_r}$ Alkhateeb et al. [2014], maps a set of parameters $\mathbf{s} \in \mathbb{R}^{4P}$ to a matrix $\mathbf{H}_f \in \mathbb{C}^{N_t \times N_r}$, where P is the number of constituent channel multipaths. The model $\mathcal{M}$ considers $\mathbf{H}_f$ to be a superposition of the channel responses for each of the P paths, and can be described as

$$\mathbf{H}_f = \mathcal{M}(\underbrace{\{\alpha_p, \theta_a^p, \theta_d^p, \tau_p\}_{p=1}^P}_{\mathbf{s}}, f) = \sum_{p=1}^{P} \alpha_p e^{2\pi j f \tau_p} \mathbf{a}_r(\theta_a^p) \mathbf{a}_t(\theta_d^p)^H. \tag{1}$$

Where α_p is the gain, τ_p is the delay, and θ_d^p and θ_a^p are the angles of departure and arrival of the p-th path respectively. $\mathbf{a}_t(\cdot), \mathbf{a}_r(\cdot)$ are the array response vectors of the transmit and receive antennas.

In this work we consider two modalities of channel transformation, namely (i) transformation due to a change in carrier frequency from f_1 to f_2, and (ii) uplink-downlink (UL-DL) transformation, from the uplink subband F_{UL} to the downlink subband F_{DL}, both of which are described next.

2.1 Physics-based Channel Transformation

Based on the channel model in (1), a channel transformation can be interpreted as a translation of the parameters $\mathbf{s}$ used to describe the channel. Such translations are less complex mappings than transformations in the channel space $\mathbb{C}^{N_t \times N_r}$, and are thus easier to learn. To learn these mappings, we leverage the the discretized, ML-compatible reformulation of the physics-based channel model as described in Wagle et al. [2025], wherein the channel in (1) is expressed as $\mathbf{H}_f = \mathbf{W} \odot \mathbf{D}_f$, where $\mathbf{D}_f$ is an array response dictionary with $R^{|\Gamma|}$ entries, each being a complex matrix of size $N_r \times N_t$, R is the user-defined discretization resolution, Γ is the set of discretized parameters and $\mathbf{W} \in \mathbb{R}^{R^{|\Gamma|}}$ is a sparse weight tensor. We adapt Γ and $\mathbf{D}_f$ for the channel transformation task as discussed next.

2.1.1 Frequency Transformation

For channel transformations due to a shift in the transmission frequency, we consider $\Gamma = \{\tau, \theta_a, \theta_d\}$ and dictionary $\mathbf{D}_f$ with R^3 entries, where $\mathbf{D}_f[l, m, n] = e^{2\pi j f \tau_l} \mathbf{a}_r(\theta_m) \mathbf{a}_t(\theta_n)^H$, where $\tau_l, \theta_m, \theta_n$ are discretized as per Wagle et al. [2025]. Now, for such a transformation, the geometric propagation paths are considered largely invariant and a significant part of the transformation is caused by changes in the gain α_p, frequency f and the path delay τ_p. Thus, given a change in frequency from f_1 to f_2, we can construct the corresponding dictionaries $\mathbf{D}_{f_1}, \mathbf{D}_{f_2}$. Now, the overall channel transformation can be interpreted as a change in α_p and τ_p in (1).

Thus, given a channel $\mathbf{H}_{f_1}$ in the reference frequency f_1, we train a physics-based transformation model (PBTM) $g_\psi(\cdot) : \mathbb{R}^{2 \times N_r \times N_t} \rightarrow \mathbb{R}^{R^3}$ parametrized by ψ, to produce a weight tensor $\mathbf{W} = g_\psi(\mathbf{H}_{f_1})$ and calculate the target channel $\mathbf{H}_{f_2}$ in the target frequency f_2 as

$$\mathbf{H}_{f_2} = G_\psi(\mathbf{H}_{f_1}, \mathbf{D}_{f_2}) = \sum_{k,l,m=0}^{R} \mathbf{W}[k, l, m] \odot \mathbf{D}_{f_2}[k, l, m] = \sum_{k,l,m=0}^{R} g_\psi(\mathbf{H}_{f_1})[k, l, m] \odot \mathbf{D}_{f_2}[k, l, m]. \tag{2}$$

Thus, as the dictionary $\mathbf{D}_{f_2}$ in the transformation pipeline corresponds to the target frequency f_2, the PBTM $g_\psi(\cdot)$ learns the translation to f_2 by estimating the appropriate parameter discretizations.

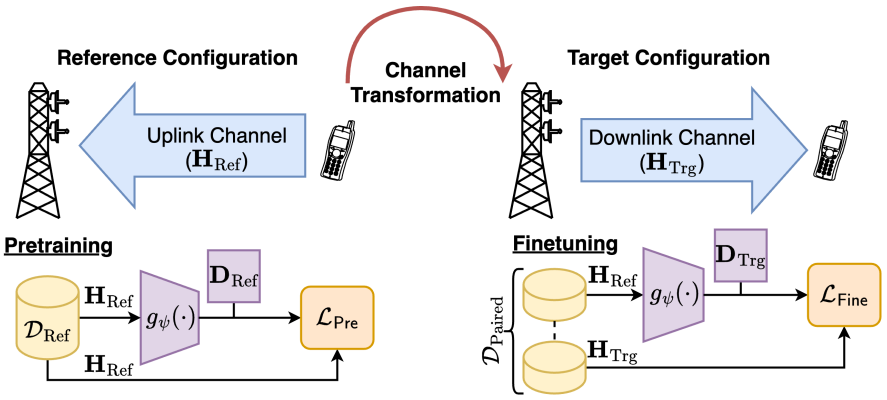


Figure 1: In the pre-training stage of our method, we leverage the physics-based latent space and abundant channel data from the reference configuration $\mathcal{D}_{\text{Ref}}$ to learn a suitable initialization. In the fine-tuning stage, we learn the channel transformations as translations in the consistent physics-based latent space, which requires fewer paired channel data $\mathcal{D}_{\text{Paired}}$ from both configurations.

2.1.2 Uplink to Downlink Transformation

For UL-DL transformations, to ensure tractability of models, we consider a MISO system, with receive antennas $N_r = 1$ and $\theta_a = \frac{\pi}{2}$. For this modality, we consider $\Gamma = \{\tau, \theta_d\}$ and a dictionary $\mathbf{D}_f$ with R^2 entries, where $\mathbf{D}_f[l, m] = e^{2\pi j f \tau_l} \mathbf{a}_t(\theta_m)^H$, where τ_l, θ_m are discretized as per Wagle et al. [2025]. We consider the channels for each subcarrier across an uplink subband $F_{\text{UL}} = \{f_{\text{UL}}^{\min} + k\Delta_f\}_{k=0}^{k=K-1}$, where Δ_f is the change in frequency between subcarriers, and K is the total number of subcarriers in the band. This results in the uplink channel $\mathbf{H}_{\text{UL}} \in \mathbb{R}^{2\times N_r \times N_t \times |F_{\text{UL}}|}$. Similarly, we consider a downlink channel $\mathbf{H}_{\text{DL}} \in \mathbb{R}^{2\times N_r \times N_t \times |F_{\text{DL}}|}$, where F_{DL} is the downlink subband.

Now, based on the characteristics of the channel model (1), the channel parameters $\{\alpha_p, \tau_p, \theta_a^p, \theta_d^p\}_{p=1}^P$ remain largely constant across the subband. Secondly, we observe that a channel transformation from the subcarrier $f_{\text{UL}}^{\min}$ to another subcarrier $f_{\text{UL}}^{\min} + k\Delta_f$ can be expressed as

$$\mathbf{H}_{f_{\text{UL}}^{\min}+k\Delta_f} = \sum_{m,n=1}^{R} \mathbf{W}[k,l,m] \odot \mathbf{D}_{f_{\text{UL}}^{\min}+k\Delta_f}[m,n] = \underbrace{\sum_{m,n=0}^{R} \mathbf{W}[m,n] \odot \mathbf{D}_{f_{\text{UL}}^{\min}}[m,n]}_{\mathbf{H}_{f_{\text{UL}}^{\min}}} \cdot e^{2\pi j \tau_m k \Delta_f}. \quad (3)$$

This indicates that, if path parameters are known, the channel $\mathbf{H}_{f_{\text{UL}}}^{\min}$ can be extrapolated across the remaining subbands by scaling it by $e^{2\pi j \tau_m k \Delta_f}$, where k, τ_m and Δ_f are known. Thus, similar to (2), a neural network can be trained to predict the downlink channel from the uplink channel as

$$\mathbf{H}_{\text{DL}} = G_\psi(\mathbf{H}_{\text{UL}}, \mathbf{D}_{\text{DL}}) = \left[\sum_{m,n=0}^{R} g_\psi(\mathbf{H}_{\text{UL}})[m,n] \odot \mathbf{D}_{f_{\text{UL}}^{\min}}[m,n] \cdot e^{2\pi j k \tau_m \Delta_f} \right]_{k=0}^{K-1}. \quad (4)$$

Similar to Sec. 2.1.1, as the dictionary $\mathbf{D}_{\text{DL}}$ in this pipeline corresponds to the downlink subband F_{UL}, the PBTM $g_\psi(\cdot)$ learns the translation to F_{UL} through suitable parameter estimation.

3 Training Pipeline

We consider a neural-network based PBTM $g_\psi(\cdot)$, unpaired channel data belonging to the reference configuration, given by $\mathcal{D}_{\text{Ref}}$ such that $\mathbf{H}_{\text{Ref}} \in \mathcal{D}_{\text{Ref}}$, and paired data from the reference and target configurations, given by $\mathcal{D}_{\text{Paired}}$ such that $\{\mathbf{H}_{\text{Ref}}, \mathbf{H}_{\text{Trg}}\} \in \mathcal{D}_{\text{Paired}}$, and the associated dictionaries for both configurations $\mathbf{D}_{\text{Ref}}, \mathbf{D}_{\text{Trg}}$. As mentioned in Sec. 1, collecting and annotating paired data is extremely resource intensive in practical scenarios. Thus, we assume that $|\mathcal{D}_{\text{Paired}}| << |\mathcal{D}_{\text{Ref}}|$.

3.1 Pre-Training on Reference Configuration Data

First, we pre-train the PBTM $g_\psi(\cdot)$ purely on reference data $\mathcal{D}_{\text{Ref}}$ using the loss function

Algorithm 1 Training Pipeline for Physics-based Channel Transformation

Pre-training on Reference Data

Given: PBTM $g_\psi(\cdot)$, Dataset of reference configuration channels $\mathcal{D}_{\text{Ref}}$, Reference configuration array response dictionary $\mathbf{D}_{\text{Ref}}$.

1: Sample $\mathbf{H}_{\text{Ref}} \in \mathcal{D}_{\text{Ref}}$ and produce the weight tensor $\mathbf{W}$ as $\mathbf{W} = g_\psi(\mathbf{H}_{\text{Ref}})$.
2: Construct reference channel using $G_\psi(\mathbf{H}_{\text{Ref}}, \mathbf{D}_{\text{Ref}})$ via (2) or (4) depending on modality.
3: Calculate the loss using (5) and update ψ.

Output: Pretrained PBTM $g_{\psi'}(\cdot)$.

Fine-tuning on Paired Data

Given: Pretrained PBTM $g_{\psi'}(\cdot)$, Dataset of paired channels $\mathcal{D}_{\text{Paired}}$, Target configuration array response dictionary $\mathbf{D}_{\text{Trg}}$.

1: Sample $\mathbf{H}_{\text{Ref}}, \mathbf{H}_{\text{Trg}} \in \mathcal{D}_{\text{Paired}}$ and produce the weight tensor $\mathbf{W}$ as $\mathbf{W} = g_{\psi'}(\mathbf{H}_{\text{Ref}})$.
2: Construct target channel using $G_{\psi'}(\mathbf{H}_{\text{Ref}}, \mathbf{D}_{\text{Trg}})$ via (2) or (4) depending on modality.
3: Calculate the loss using (6) and update ψ'.

Output: Trained PBTM $g_{\tilde{\psi}}(\cdot)$

$$\mathcal{L}_{\text{Pre}} = \frac{1}{|\mathcal{D}_{\text{Ref}}|} \sum_{\mathbf{H}_{\text{Ref}} \in \mathcal{D}_{\text{Ref}}} || \sum G_\psi(\mathbf{H}_{\text{Ref}}, \mathbf{D}_{\text{Ref}}) - \mathbf{H}_{\text{Ref}} ||_F^2, \tag{5}$$

where $\mathbf{D}_{\text{Ref}}$ is the array response dictionary associated with the reference configuration. For frequency transformation, $\mathbf{D}_{\text{Ref}} = \mathbf{D}_{f_1}$, and for UL-DL transformation, $\mathbf{D}_{\text{Ref}} = \mathbf{D}_{\text{UL}}$.

The purpose of the pre-training stage is to have the PBTM $g_\psi(\cdot)$ converge to a suitable initialization $g_{\psi'}(\cdot)$ for the channel transformation task. The pre-training stage enables this as the parameters of a target configuration channel are largely similar to those of the corresponding reference configuration channel. For example, from (1), we can observe that for a transformation due to shift in carrier frequency, the angles of arrival and departure, $\{\theta_a^p, \theta_p^d\}_{p=1}^p$ remain largely constant. The pre-training procedure on unpaired data from the reference domain implicitly learns these consistencies in the parameter space, resulting in a suitable initialization point for fine-tuning, described next.

3.2 Fine-tuning on Paired Reference-Target Data

The next step of the training pipeline involves fine-tuning the pretrained PBTM $g_{\psi'}(\cdot)$ using the paired data $\mathcal{D}_{\text{Paired}}$ from both reference and target configurations using the loss function

$$\mathcal{L}_{\text{Fine}} = \frac{1}{|\mathcal{D}_{\text{Paired}}|} \sum_{\mathbf{H}_{\text{Ref}}, \mathbf{H}_{\text{Trg}} \in \mathcal{D}_{\text{Paired}}} || \sum G_{\psi'}(\mathbf{H}_{\text{Ref}}, \mathbf{D}_{\text{Trg}}) - \mathbf{H}_{\text{Trg}} ||_F^2, \tag{6}$$

where $\mathbf{H}_{\text{Ref}}, \mathbf{H}_{\text{Trg}}$ are paired channel samples from the reference and target domain respectively, and $\mathbf{D}_{\text{Trg}}$ is the array response dictionary associated with the target configuration. For frequency transformation, $\mathbf{D}_{\text{Trg}} = \mathbf{D}_{f_2}$, and for UL-DL transformation, $\mathbf{D}_{\text{Trg}} = \mathbf{D}_{\text{DL}}$.

As the PBTM $g_{\psi'}(\cdot)$ has been pretrained to a suitable initialization, only a few paired datapoints $\mathcal{D}_{\text{Paired}}$ are needed to learn useful translations in the parameter space, as it is an easier task. Thus, the pre-training stage of the two-stage channel transformation pipeline extracts transformation-critical information from the unpaired reference data, and uses it to converge to a suitable initialization for the fine-tuning task in the second stage. A summary of the training pipeline is given in Algo. 1.

4 Experimental Results

For our experiments, we use the DeepMIMO framework Alkhateeb [2019] to generate ray-tracing datasets for two real-life scenarios, namely (i) the indoor office scenario (Indoor), and (ii) a street crossing with blocking and reflecting surfaces (Outdoor). For the PBTM, we use a transformer architecture with a convolutional encoder, and compare it against a black-box FSR-CNN model (*FSRCNN*) Dong et al. [2016] with approximately the same number of parameters. *FSRCNN* does not use the physics-based latent space, and can only be trained with paired datapoints. We consider a MIMO system for frequency transformation, with $N_t = N_r = 8$, and a MISO system for UL-DL transformation, with $N_t = 8, N_r = 1$ with at most 8 channel multipaths. For frequency

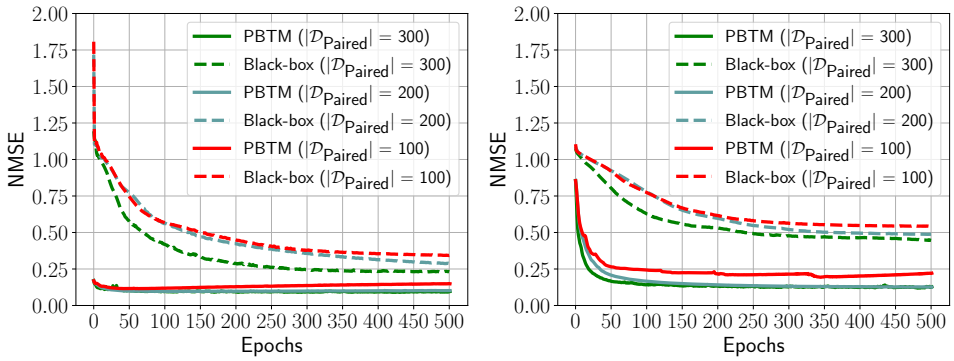


Figure 2: Our method outperforms similar black-box architectures for both frequency (Left) and UL-DL transformation (Right) for various amounts of paired samples $\mathcal{D}_{\text{Paired}}$. In both cases, the pre-training stage of our training pipeline allows the PBTM to learn transformation-critical information, thus resulting in improved performance despite the small amount of paired data.

Testing Dataset	*Ref. Only*	*Ref+Paired*	*Best Case*	*Transformed* (Ours)
$\mathcal{D}_{\text{Ref}}$	0.018	0.02	0.012	**0.014**
$\mathcal{D}_{\text{Trg}}$	0.312	0.1	0.018	**0.038**

Table 1: In the context of a downstream compression task, given a small number of paired datapoints, using them to train a PBTM (*Transformed*) rather than directly training a compression model (*Ref + Paired*) results in performance similar to when all paired datapoints are available (*Best Case*).

transformation, we consider reference frequency $f_1 = 3.5$ GHz and target frequency $f_2 = 28$ GHz. For UL-DL transformation, we consider a central frequency of 2.4 GHz, with 128 subcarriers of width 40 Mhz spread uniformly around it. We consider the lower 64 subcarriers for the uplink channel, and the upper 64 subcarriers for the downlink channel. For the PBTM, we use resolution $R = 64$.

4.1 Generalization to Various Transformation Modalities

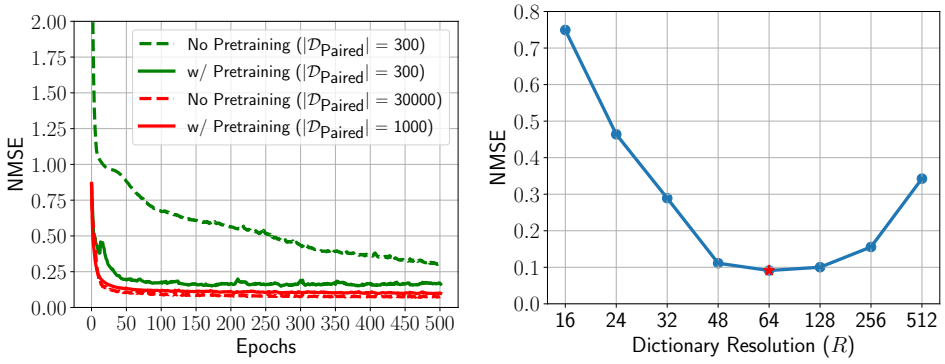
In Fig. 2, we illustrate the ability of our framework to adapt to different modalities of transformation, namely, transformations caused by a change in frequency and the UL-DL channel. We show our results for the Indoor scenario using different sizes of paired dataset $\mathcal{D}_{\text{Paired}}$.

We observe that for both modalities, our method significantly outperforms similarly sized black-box models, with the performance advantage increasing as the number of paired datapoints decrease. Our model also converges faster than the black-box models. This is because, firstly, the pre-training stage of our framework allows the model to converge to a suitable initialization by extracting relevant information from the reference configuration data, and secondly, the transformation model learns the translations in the parameter space, which is easier than learning the changes in the channel itself, thus requiring fewer datapoints to train. This illustrates that our method can effectively learn the channel transformation for different modalities while requiring a much smaller amount of paired data.

4.2 Impact on Downstream Compression Tasks

In Table 1, we illustrate the efficacy of our transformation framework in the context of a downstream compression task. We consider a set of $30,000$ channels from the reference configuration ($\mathcal{D}_{\text{Ref}}$), and a smaller set of 300 paired channels from both the reference and target configurations ($\mathcal{D}_{\text{Paired}}$). We consider a CSINet compression model Wen et al. [2018] in four training scenarios, namely, (i) training CSINet using only $\mathcal{D}_{\text{Ref}}$ (*Ref. Only*), (ii) training CSINet using $\mathcal{D}_{\text{Ref}}$ as well as $\mathcal{D}_{\text{Paired}}$ (*Ref + Paired*) , (iii) using the $\mathcal{D}_{\text{Ref}}$ and $\mathcal{D}_{\text{Paired}}$ to train a transformation model, which then transforms $\mathcal{D}_{\text{Ref}}$ to $\hat{\mathcal{D}}_{\text{Paired}}$, and use $\mathcal{D}_{\text{Ref}}$ and $\hat{\mathcal{D}}_{\text{Paired}}$ to train CSINet (*Transformed*), and finally (iv) assuming the best case scenario, where $|\mathcal{D}_{\text{Paired}}| = 30,000$, and use it to train CSINet (*Best Case*). We evaluate the trained CSINet model on ground truth channels from the target configuration.

We observe that, in the *Ref. Only* case, CSINet cannot generalize to the target configuration $\mathcal{D}_{\text{Trg}}$, as no data from the target distribution is available during training. In the *Ref + Paired* case, CSINet



(a) (b)

Figure 3: The pre-training stage of our training pipeline extracts transformation-critical information from source configuration data, resulting in a better initialization for the fine-tuning task. It also requires fewer paired datapoints to achieve comparable performance with the best-case scenario of having a large amount of paired data from both environment configurations (a). The ideal operating resolution R for the array response dictionary $\mathbf{D}$ balances the angular resolution of the dictionary $\mathbf{D}$ and the expressiveness of the learnable parameters of the transformation model (b).

performs better on channels from $\mathcal{D}_{\text{Trg}}$, but the accuracy is still significantly poorer than for the reference configuration $\mathcal{D}_{\text{Ref}}$, as the number of datapoints available from $\mathcal{D}_{\text{Trg}}$ is small. For the *Transformed* case, we observe that CSINet shows significant improvements in performance on $\mathcal{D}_{\text{Trg}}$, indicating that the transformed channels produced by our method are very similar to the ground truth data from the target configuration. Secondly, comparing *Transformed* and *Best Case*, we observe that the difference between them is not significant, indicating that by using our transformation framework, even a small number of paired datapoints can be leveraged to approach the best case performance.

4.3 Effect of pre-training

In Fig. 3(a), we study the impact of the pre-training stage in terms of (i) the NMSE loss for fine-tuning with and without pre-training for a given number of paired datapoints (green), and (ii) the number of paired datapoints required to match the performance for an ideal scenario where all datapoints are paired ($|\mathcal{D}_{\text{Paired}}| = 30,000$) (red). We consider UL-DL transformation for the Outdoor scenario.

We observe that for a given number of paired datapoints, the pretrained model converges significantly faster than the non-pretrained model, with a lower NMSE loss at the end of training. This is because pre-training enables the model to learn transformation critical information which allows it to converge faster in the fine-tuning stage. We also observe that for an ideal scenario with $30,000$ paired datapoints, our pretrained model requires only 1000 paired datapoints to achieve performance parity. This is because the pre-training stage leverages the abundant information in the reference data $\mathcal{D}_{\text{Ref}}$, thus requiring fewer paired datapoints in the fine-tuning stage to achieve similar performance.

4.4 Effect of Increasing Dictionary Size

In Fig. 3(b), we observe the impact of changing the resolution R of the array response dictionary $\mathbf{D}$. We keep the number of parameters in the transformation model constant by changing the number of convolutional kernels. We consider carrier frequency transformation for the Outdoor scenario.

We observe a clear optimal operating point at a resolution of $R = 64$. This is because, for larger R, fewer learnable parameters limit the representational variety that the model is able to learn, while for smaller R, the coarseness of the dictionary limits the range of channels that can be expressed.

5 Conclusion

In this paper, we propose a physics-based channel transformation model to learn channel transformations between environment configurations. We leverage the physics-based latent spaces learned by the model in conjunction with the large number of unpaired data from the reference configuration to

develop a two-stage training process that learns a suitable initialization, thus requiring fewer paired datapoints to achieve performance parity with state-of-the-art methods.

References

A. Alkhateeb. DeepMIMO: A generic deep learning dataset for millimeter wave and massive MIMO applications. In *Proc. of Info. Theory and App. Wkshp.*, pages 1–8, San Diego, CA, Feb 2019.

Ahmed Alkhateeb, Omar El Ayach, Geert Leus, and Robert W. Heath. Channel estimation and hybrid precoding for millimeter wave cellular systems. *IEEE Jrnl. of Selected Topics in Sig. Proc.*, 8(5): 831–846, 2014. doi: 10.1109/JSTSP.2014.2334278.

Muhammad Alrabeiah and Ahmed Alkhateeb. Deep learning for tdd and fdd massive mimo: Mapping channels in space and frequency. In *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, pages 1465–1470, 2019. doi: 10.1109/IEEECONF44664.2019.9048929.

Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. *CoRR*, abs/1608.00367, 2016. URL `http://arxiv.org/abs/1608.00367`.

Jiajia Guo, Chao-Kai Wen, Shi Jin, and Geoffrey Ye Li. Overview of deep learning-based csi feedback in massive mimo systems. *IEEE Transactions on Communications*, 70(12):8017–8045, 2022. doi: 10.1109/TCOMM.2022.3217777.

Shihao Ju and Theodore S. Rappaport. 142 ghz multipath propagation measurements and path loss channel modeling in factory buildings. In *ICC 2023 - IEEE Inter. Conf. on Commun.*, pages 5048–5053, 2023. doi: 10.1109/ICC45041.2023.10278815.

Shihao Ju, Yunchou Xing, Ojas Kanhere, and Theodore S. Rappaport. Sub-terahertz channel measurements and characterization in a factory building. In *ICC 2022 - IEEE Inter. Conf. on Commun.*, pages 2882–2887, 2022. doi: 10.1109/ICC45855.2022.9838910.

Haya Al Kassir, Zaharias D. Zaharis, Pavlos I. Lazaridis, Nikolaos V. Kantartzis, Traianos V. Yioultsis, and Thomas D. Xenos. A review of the state of the art and future challenges of deep learning-based beamforming. *IEEE Access*, 10:80869–80882, 2022. doi: 10.1109/ACCESS.2022.3195299.

Satyam Kumar, Akshay Malhotra, and Shahab Hamidi-Rad. Channel parameter estimation in wireless communication: A deep learning perspective. In *2024 IEEE Inter. Conf. on Mach. Learn. for Communication and Networking (ICMLCN)*, pages 511–516, 2024. doi: 10.1109/ICMLCN59089.2024.10624782.

Mu Liang and Ang Li. Deep learning-based channel extrapolation for pattern reconfigurable massive mimo. *IEEE Transactions on Vehicular Technology*, 73(3):4395–4400, 2024. doi: 10.1109/TVT.2023.3323803.

Zhenyu Liu, Lin Zhang, and Zhi Ding. Exploiting bi-directional channel reciprocity in deep learning for low rate massive mimo csi feedback. *IEEE Wireless Communications Letters*, 8(3):889–892, 2019. doi: 10.1109/LWC.2019.2898662.

Qian Mao, Fei Hu, and Qi Hao. Deep learning for intelligent wireless networks: A comprehensive survey. *IEEE Commun. Surveys & Tutorials*, 20(4):2595–2621, 2018.

Aditya Sant, Afshin Abdi, and Joseph Soriaga. Deep sequential beamformer learning for multipath channels in mmwave communication systems. In *2022 IEEE Inter. Conf. on Acoustics, Speech and Sig. Proc. (ICASSP)*, pages 5198–5202. IEEE, 2022.

Satyavrat Wagle, Akshay Malhotra, Shahab Hamidi-Rad, Aditya Sant, David J. Love, and Christopher G. Brinton. Physics-based generative models for geometrically consistent and interpretable wireless channel synthesis, 2025. URL `https://arxiv.org/abs/2506.00374`.

Chao-Kai Wen, Wan-Ting Shih, and Shi Jin. Deep learning for massive mimo csi feedback. *IEEE Wireless Commun. Letters*, 7(5):748–751, 2018. doi: 10.1109/LWC.2018.2818160.