A Digital Twin Assisted Framework for Interference Nulling in Millimeter Wave MIMO Systems

Yu Zhang, Tawfik Osman, and Ahmed Alkhateeb

2023 IEEE International Conference on Communications Workshops (ICC Workshops) | 979-8-3503-3307-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICCWORKSHOPS57953.2023.10283591

***Abstract*—Millimeter wave (mmWave) and terahertz MIMO systems rely on pre-defined beamforming codebooks for both initial access and data transmission. However, most of the existing codebooks adopt pre-defined beams that focus mainly on improving the gain of their target users, without taking interference into account, which could incur critical performance degradation in the dense networks. To address this problem, in this paper, we propose a sample-efficient digital twin-assisted beam pattern design framework that learns how to form the beam pattern to reject the signals from the interfering directions. The proposed approach does not require any explicit channel knowledge or any coordination with the interferers. The adoption of the digital twin improves the sample efficiency by better leveraging the underlying signal relationship and by incorporating a demand-based data acquisition strategy. Simulation results show that the developed signal model-based learning framework can significantly reduce the actual interaction with the radio environment (i.e., number of measurements) compared to the model-unaware design, leading to a more practical and efficient interference-aware beam design approach.**

I. Introduction

Rejecting in-band interference in the RF domain of the hybrid or fully analog MIMO systems is possible owing to the extra degree of freedom brought by multiple antennas. However, realizing its full potential is challenging in practical systems due to the lack of channel knowledge as well as the additional hardware constraints. As a result, the existing approaches normally require large beam learning overhead in order to form well-shaped analog beams. Moreover, when the codebook design problem is considered, such learning overhead gets magnified and increases linearly with respect to the codebook size, since the learning experience of one beam is not transferable to the others. Such prohibitive overhead downplays the potential of the system, and it motivates the development of sample efficient interference-aware beam codebook design framework, which is the focus of this paper.

Contributions: In this paper, we develop a model-based digital twin-assisted learning framework that achieves higher sample efficiency by better leveraging the underlying signal model. The improvement on the sample efficiency has the potential of reducing the beam learning overhead as well as shortening the convergence time of the proposed solution. The proposed digital twin-assisted learning framework also provides flexibility to the practical deployment in terms of enabling data sharing and cooperation (hence better learning) in complex scenarios. We extensively evaluate the proposed interference-aware beam learning framework using numerical simulations. This provides a comprehensive assessment of the capability of the proposed approach in nulling interference without requiring any knowledge about the channel, array geometry, or user location. The role of the digital twin model is also tested which empirically shows its efficacy in guiding the beam learning process and highlights its potential gain of reducing the overall learning overhead.

Prior work: The prior work of interference nulling beamforming design either focuses on MIMO transceivers with fully-digital architectures [1]–[3], or assumes some kind of channel information, such as covariance, of both the target and interferers [4], [5]. Some other approaches that do not rely on channel information are also proposed in the past [6], [7]. But they, in general, do not leverage underlying signal relationship which leads to large beam learning overhead and are not robust.

II. System and Channel Models

We consider the communication system where a mmWave MIMO base station (BS), equipped with M antennas, is communicating with a single-antenna user equipment (UE) in an uplink mode. Moreover, we assume that there exist K (≥ 1) non-cooperative interference transmitters[1] in the vicinity of the BS, operating at the same frequency bands and hence causing interference to the BS receiver. Therefore, if the UE transmits a symbol $x \in \mathbb{C}$ to the BS, and the other K interference transmitters also transmit symbols $x_k \in \mathbb{C}, k = 1, \ldots, K$ at the same time and frequency slot, such that all the transmitted symbols satisfy the same average power constraint, i.e., $\mathbb{E}[|x|^2] = P_x$ and $\mathbb{E}[|x_k|^2] = P_x, \forall k$, the received signal at the BS after combining can then be expressed as

$$y = \mathbf{w}^H \mathbf{h} x + \sum_{k=1}^{K} \mathbf{w}^H \mathbf{h}_k x_k + \mathbf{w}^H \mathbf{n}, \tag{1}$$

where $\mathbf{h} \in \mathbb{C}^{M \times 1}$ is the channel between the BS and the UE, $\mathbf{h}_k \in \mathbb{C}^{M \times 1}$ is the channel between the BS and the k-th interference transmitter. It is worth pointing out here that for clarity, we subsume the factors such as path-loss and transmission power into the channels. $\mathbf{n} \sim \mathcal{CN}(0, \sigma^2 \mathbf{I})$ is the receive noise vector at the BS with σ^2 being the noise power and $\mathbf{w} \in \mathbb{C}^{M \times 1}$ is the combining vector used by the BS. Furthermore, given the high cost and power consumption of the mixed-signal components, we consider a practical system

[1]For ease of exposition, each interference transmitter is also assumed to have a single antenna. This means that the interference signals are being transmitted omni-directionally.

Authorized licensed use limited to: ASU Library. Downloaded on March 24,2024 at 06:44:52 UTC from IEEE Xplore. Restrictions apply.

where the BS has only one radio frequency (RF) chain[2] and employs analog-only beamforming/combining using a network of r-bit quantized phase shifters. Therefore, the combining vector at the BS can be written as

$$\mathbf{w} = \frac{1}{\sqrt{M}} \left[e^{j\theta_1}, e^{j\theta_2}, \ldots, e^{j\theta_M}\right]^T, \tag{2}$$

where each phase shift $\theta_m, \forall m = 1, \ldots, M$ is selected from a finite set $\mathbf{\Psi}$ with 2^r possible discrete values drawn uniformly from $(-\pi, \pi]$. The normalization factor $M^{-1/2}$ is to make sure the combiner has unit power, i.e., $\|\mathbf{w}\|_2^2 = 1$.

We adopt a geometric channel model for the channel between BS and UE, as well as the interference channels between BS and any interfering transmitters. Hence, the channel between BS and its served UE takes the following form (the channel between BS and any interference transmitter takes similar form)

$$\mathbf{h} = \sum_{\ell=1}^{L} \alpha_\ell \mathbf{a}(\phi_\ell, \vartheta_\ell), \tag{3}$$

where L is the number of multi-paths. Each path ℓ has a complex gain α_ℓ, which includes the path-loss. The angles ϕ_ℓ and ϑ_ℓ represent the ℓ-th path's azimuth and elevation angles of arrival respectively, and $\mathbf{a}(\phi_\ell, \vartheta_\ell)$ is the BS array response vector. The exact expression of $\mathbf{a}(\phi_\ell, \vartheta_\ell)$ depends on the array geometry and possible hardware impairments.

III. Problem Formulation

In this paper, we investigate the design of the analog combining/precoding that achieves interference awareness (i.e., attempts to address the interference) without explicitly knowing any channel state information. Given the receive signal (1) at the BS, the achievable rate of its target user can be written as

$$R = \log_2\left(1 + \frac{|\mathbf{w}^H\mathbf{h}|^2 P_x}{\sum_{k=1}^{K} |\mathbf{w}^H\mathbf{h}_k|^2 P_x + \sigma^2}\right). \tag{4}$$

The objective is to design the combining vector $\mathbf{w}$ such that the achievable rate of the target user, i.e., (4), can be maximized. Given the monotonicity of the logarithm function, this is equivalent to maximize the SINR term in (4). Therefore, the problem can be cast as

$$\mathbf{w}^\star = \arg\max_{\mathbf{w}} \frac{|\mathbf{w}^H\mathbf{h}|^2 P_x}{\sum_{k=1}^{K} |\mathbf{w}^H\mathbf{h}_k|^2 P_x + \sigma^2}, \tag{5}$$

$$\text{s. t. } w_m = \frac{1}{\sqrt{M}} e^{j\theta_m}, \ \forall m = 1, ..., M, \tag{6}$$

$$\theta_m \in \mathbf{\Psi}, \ \forall m = 1, ..., M, \tag{7}$$

where w_m is the m-th element of the combining vector $\mathbf{w}$ [3]. Solving the interference-aware beam pattern design problem formulated in (5) is challenging due to the non-convex and discrete hardware constraints (6) and (7), as well as the unknown channels in the objective function (5). Therefore, it is hard to solve (5) using the conventional optimization methods [1]–[3]. An important observation, however, is that for a given combining beam $\mathbf{w}$, evaluating the SINR requires only the power values (after combining) of the desired and interference signals, and does not require explicit knowledge about the channel vectors. With this observation, we cast our problem as developing an online machine learning approach that learns how to design an interference-aware beam pattern $\mathbf{w}$ that optimizes (5), **given only the receive power measurements** for the signal plus interference and noise, $\left|\mathbf{w}^H\mathbf{h}\right|^2 P_x + \sum_{k=1}^{K} \left|\mathbf{w}^H\mathbf{h}_k\right|^2 P_x + \sigma^2$, and the interference plus noise, $\sum_{k=1}^{K} \left|\mathbf{w}^H\mathbf{h}_k\right|^2 P_x + \sigma^2$.

IV. Online Learning of Interference Aware Beam Pattern Design

In this section, we present the proposed online reinforcement learning based interference-aware beam pattern learning approach. To solve the problem with reinforcement learning, we first fit all the ingredients of problem (5) into a general reinforcement learning framework as follows:

- **State:** We define the state $\mathbf{s}_t$ as a vector that consists of the phases of all the phase shifters at the t-th iteration, that is, $\mathbf{s}_t = [\theta_1, \theta_2, \ldots, \theta_M]^T$.
- **Action:** We define the action $\mathbf{a}_t$ as the element-wise changes to all the phases in $\mathbf{s}_t$. Since the phases can only take values in $\mathbf{\Psi}$, a change of a phase represents the action that a phase shifter selects a value from $\mathbf{\Psi}$. Therefore, the action is directly specified as the next state, i.e., $\mathbf{a}_t = \mathbf{s}_{t+1}$, which can be viewed as a deterministic transition in the Markov Decision Process (MDP).
- **Reward:** We define a binary reward mechanism, i.e., the reward r_t takes values from $\{+1, -1\}$. Since the objective of (5) is to maximize the SINR performance, we compare the SINR achieved by the current combining vector, denoted as SINR_t, with the previous one, i.e., SINR_{t-1}. The reward is determined according to the following rule: $r_t = +1$, if $\text{SINR}_t > \text{SINR}_{t-1}$; $r_t = -1$, otherwise.

Deep Reinforcement Learning Architecture: Given the reinforcement learning formulation above for the interference-aware beam learning problem, we adopt an actor-critic based deep reinforcement learning architecture. This follows the learning framework that we proposed earlier in [8]. In summary, both the actor and critic networks are implemented using elegant fully-connected (FC) feed-forward neural networks. The input of the actor network is the state and the output is the action, while the critic network takes in the state-action pair and outputs the predicted Q value.[4] Moreover, to respect the discrete phase shifter hardware constraint (7), we perform an element-wise quantization to make the predicted action a

[2]It is very important to note that the RF precoder in a system with hybrid architecture is normally constructed using pre-defined codebooks that have pre-determined beams. Therefore, the learned beams in this paper can be included in such codebooks and be used in the hybrid analog/digital architectures as well.

[3]It is important to note that the proposed interference-aware beam learning approach can be straightforwardly extended to learning a codebook with multiple beams by, for example, using the user clustering and assignment algorithm proposed in [8].

[4]The detailed architectures and the parameters of the adopted neural networks are provided in Section VI-A.

Authorized licensed use limited to: ASU Library. Downloaded on March 24,2024 at 06:44:52 UTC from IEEE Xplore. Restrictions apply.

valid one. To be more specific, assume that $\widehat{\mathbf{a}}_t$ is the predicted action from the actor network at time t. Then, the action that finally gets implemented to the system is given by

$$[\mathbf{a}_t]_m = \arg\min_{\theta \in \mathbf{\Psi}} \left| [\widehat{\mathbf{a}}_t]_m - \theta \right|, \ \forall m = 1, \ldots, M. \tag{8}$$

It is worth emphasizing that such quantization operation is only activated when the system is actually implementing the predicted action by the actor network to obtain reward. It is not involved in the training process of the actor network due to its non-differentiability.

Despite its full compatibility with the considered system, the proposed interference-aware beam learning solution still has two drawbacks. First, it requires a relatively large number of iterations to find a qualified beam pattern, especially when the number of antennas is large. As a result, this incurs a large beam learning overhead, since these iterations are done over the air. Second, as indicated by the objective function of (5), the SINR performance of a given beam is determined by two factors: (i) The desired beamforming gain and (ii) The effectiveness of suppressing the undesired interference. However, the proposed solution does not fully leverage this information as it only focuses on the overall SINR performance. It turns out that the decomposition of these two factors, as will be further discussed in the next section, makes the data sharing among the learning processes of different beams possible, which has the potential of improving the convergence behavior of the beam/codebook learning algorithm.

V. Digital Twin Assisted Beam Learning Framework

In this section, we describe in detail the proposed digital twin assisted interference-aware beam pattern learning framework. The motivations of introducing the digital twin are mainly two-folds. First, it has the potential of improving the sample efficiency (i.e., reducing the number of interactions with the actual environment) of the learning process [9]. Second, it facilitates other more complex tasks (than learning a single beamforming vector), such as data sharing (which can be very useful in learning *interference-aware beam codebooks*) and cooperative learning[5] (among multiple BSs to avoid interfering each other).

A. Digital Twin for Beam Pattern Learning

In this subsection, we introduce the proposed digital twin that assists the learning of interference-aware beams. As mentioned before, in order to acquire the reward signal that is used for training the RL agent, the system needs to estimate two quantities, i.e., the signal power, $P_{\mathrm{S}} = \left|\mathbf{w}^H \mathbf{h}\right|^2 P_x$, and the interference plus noise power, $P_{\mathrm{I+N}} = \sum_{k=1}^{K} \left|\mathbf{w}^H \mathbf{h}_k\right|^2 P_x + \sigma^2$. Therefore, correspondingly, there are two major components in the considered digital twin that provide the agent with such information, i.e., an interference predictor and a signal predictor, as will be discussed in this subsection.

1) The key idea of digital twin: The machine learning model that virtually interacts with the agent can be considered as a **digital twin**. This model is used to *imitate* the behavior of the actual environment, aiming to reduce the expensive (sometimes, even impossible) actual evaluations of the design. In this paper, we design the digital twin with a particular emphasis on two important aspects. **First,** a digital twin should be able to accurately model the behavior of the actual environment, i.e., having accurate predictions. **Second,** training a digital twin should, in general, require less actual data samples than directly interacting with the actual environment, which yields a high sample efficiency. With these important criterions in mind, we next describe the adopted digital twin. As mentioned before, the considered digital twin consists of two major components, i.e., an interference prediction model and a signal prediction model. Formally, the interference prediction model predicts the interference plus noise power based on the configuration of the receive combining vector, which can be expressed as

$$\widehat{P}_{\mathrm{I+N}} = f_{\mathrm{in}}(\mathbf{w}; \mathbf{\Theta}_{\mathrm{in}}), \tag{9}$$

where $\mathbf{w} \in \mathbb{C}^{M \times 1}$ is the input of the model, representing the designed receive combining vector, and the output is the predicted interference plus noise power, i.e., $\widehat{P}_{\mathrm{I+N}} \in \mathbb{R}$. The model is parameterized by $\mathbf{\Theta}_{\mathrm{in}}$. Similarly, the signal prediction model predicts the signal power of a given receive combining vector, which can be written as

$$\widehat{P}_{\mathrm{S}} = f_{\mathrm{s}}(\mathbf{w}; \mathbf{\Theta}_{\mathrm{s}}), \tag{10}$$

where $\widehat{P}_{\mathrm{S}} \in \mathbb{R}$ is the predicted signal power value and $\mathbf{\Theta}_{\mathrm{s}}$ denotes the model parameters. It is worth mentioning that the architecture of f_{in} and f_{s} is not unique and is a design choice. Next, we present two candidates that could be used in the considered beam learning task.

2) Digital twin architecture: In this paper, we study two specific designs: (i) A model-based prediction architecture, and (ii) a fully-connected neural network based prediction architecture.

Model-based architecture: The model-based architecture, as its name suggests, is inspired by the underlying *signal model*. For instance, as can be seen from the expression of the interference plus noise power, i.e., $P_{\mathrm{I+N}} = \sum_{k=1}^{K} \left|\mathbf{w}^H \mathbf{h}_k\right|^2 P_x + \sigma^2$, it takes a quadratic form of the receive combining vector $\mathbf{w}$. To see this, by defining $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_K]$, $P_{\mathrm{I+N}}$ can be expressed as

$$P_{\mathrm{I+N}} = \left\|\mathbf{H}^H \mathbf{w}\right\|_2^2 P_x + \sigma^2, \tag{11}$$

$$= \mathbf{w}^H \left(P_x \mathbf{H}\mathbf{H}^H + \sigma^2 \mathbf{I}\right) \mathbf{w}, \tag{12}$$

$$= \mathbf{w}^H \mathbf{A} \mathbf{w}, \tag{13}$$

where $\mathbf{A} = P_x \mathbf{H}\mathbf{H}^H + \sigma^2 \mathbf{I}$. The signal power can be expressed in the similar form, i.e., $P_{\mathrm{S}} = \mathbf{w}^H P_x \mathbf{h}\mathbf{h}^H \mathbf{w}$. Therefore, the interference prediction network is essentially leveraged to learn the relationship (13). Inspired by this, we design

[5]For instance, as the system has full knowledge of its simulated environment, it can assign accurate reward to each agent. This has the potential of mitigating the non-stationary environment problem that exists in most of the multi-agent learning tasks.

Authorized licensed use limited to: ASU Library. Downloaded on March 24,2024 at 06:44:52 UTC from IEEE Xplore. Restrictions apply.

the interference prediction network with a focus on imitating the "behavior" of $\mathbf{A}$. Specifically, the interference prediction network is chosen to take the following form

$$f_{\text{in}}(\mathbf{w}) = \mathbf{w}^H \mathbf{Q}_{\text{in}} \mathbf{Q}_{\text{in}}^H \mathbf{w}, \tag{14}$$

where $\mathbf{Q}_{\text{in}} \in \mathbb{C}^{M \times r_{\text{in}}}$ with r_{in} being a hyperparameter. Therefore, the parameter of the interference prediction network is essentially $\mathbf{Q}_{\text{in}}$, i.e., $\mathbf{\Theta}_{\text{in}} = \{\mathbf{Q}_{\text{in}}\}$. The signal prediction network takes the similar form, i.e., $f_{\text{s}}(\mathbf{w}) = \mathbf{w}^H \mathbf{Q}_{\text{s}} \mathbf{Q}_{\text{s}}^H \mathbf{w}$, where $\mathbf{Q}_{\text{s}} \in \mathbb{C}^{M \times r_{\text{s}}}$ with r_{s} being a hyperparameter as well, which makes $\mathbf{\Theta}_{\text{s}} = \{\mathbf{Q}_{\text{s}}\}$.

Fully-connected neural network based architecture: Despite being lightweight and a better fit to the signal model, the model-based architecture, fundamentally, suffers from any mismatch between the assumed signal model and the actual signal relationship. For instance, there are normally unknown non-linearities in the practical hardware that undermine the validity of the assumed relationship between the receive combining vector and the interference plus noise power (similarly for the signal power). As a result, the signal model cannot always be met and the model-based architecture will show up certain level of residual prediction errors that are very hard to be eliminated given the less powerful expressive capability of its architecture. Motivated by this, we also investigate a more general architecture, which is built upon fully-connected neural network, given its powerful universal approximation capability [10]. Specifically, both f_{in} and f_{s} are modeled with feed-forward fully-connected neural networks. The detailed network parameters will be provided in Section VI-A.

3) Training dataset and loss function: We denote the training dataset of the interference prediction network as

$$\mathcal{D}_{\text{in}} = \left\{ \left(\mathbf{w}^{(n)}, P_{\text{I+N}}^{(n)} \right)_{n=1}^{N_{\text{in}}} \right\}, \tag{15}$$

where each data sample is comprised of a combining vector and its corresponding interference plus noise power value obtained from the actual environment, i.e., from the real measurement. N_{in} is the total number of data samples in the dataset, i.e., $|\mathcal{D}_{\text{in}}| = N_{\text{in}}$. And the dataset used for training the signal prediction network can be similarly denoted as

$$\mathcal{D}_{\text{s}} = \left\{ \left(\mathbf{w}^{(n)}, P_{\text{S}}^{(n)} \right)_{n=1}^{N_{\text{s}}} \right\}, \tag{16}$$

with N_{s} being its size. Since the target of these two networks is to predict the power values, we pose the learning problem as a regression problem conducted in a supervised fashion. Furthermore, we employ mean squared error (MSE) as the training loss function. Using the interference prediction network as an example, for the n-th data sample in $\mathcal{D}_{\text{in}}$, the loss function is defined as

$$\mathcal{L}\left(P_{\text{I+N}}^{(n)}, \widehat{P}_{\text{I+N}}^{(n)} \right) = \left| P_{\text{I+N}}^{(n)} - \widehat{P}_{\text{I+N}}^{(n)} \right|^2, \tag{17}$$

$$= \left| P_{\text{I+N}}^{(n)} - f_{\text{in}}(\mathbf{w}^{(n)}; \mathbf{\Theta}_{\text{in}}) \right|^2. \tag{18}$$

The loss function used for the signal prediction network is identical.

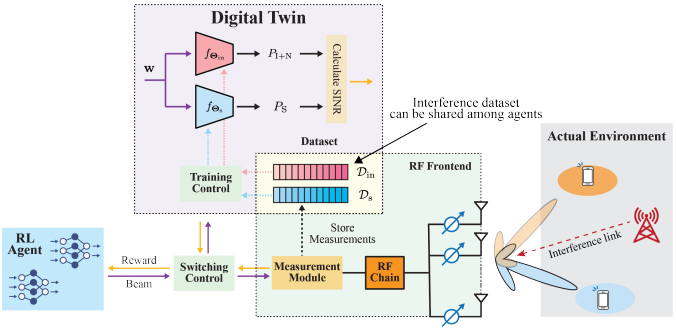


Fig. 1. An illustration of the proposed digital twin-assisted interference-aware beam pattern learning framework.

B. Digital Twin Assisted Learning

In this subsection, we discuss how to integrate the digital twin with the proposed RL based beam learning framework. Since the digital twin is essentially used to provide the RL agent with a simulated environment to interact with, it plays the same role as the actual environment. However, in order to provide high quality *synthetic* feedback, it requires training process that relies on the *authentic* data collected from the actual environment. Based on the trained digital twin, the system can virtually evaluate its designed beams without measuring the physical signals. Moreover, the system might require constantly switching between the digital twin and the actual environment, triggered by the demand for the authentic data. Next, we summarize the key components of the proposed digital twin assisted beam learning.

Initial interaction and data acquisition: The system starts with the normal interaction between the RL agent and the actual environment. To be more specific, upon forming a new beam $\tilde{\mathbf{w}}$, the BS estimates the interference plus noise power $P_{\text{I+N}}$ and the signal power P_{S}. The reward signal used for RL agent learning will then be generated. Moreover, these authentic power measurements together with the beam will be stored in the two datasets, i.e., $\mathcal{D}_{\text{in}}$ and $\mathcal{D}_{\text{s}}$, respectively. During this interaction process, two initial datasets are established.

Digital twin training: Based on the collected initial datasets $\mathcal{D}_{\text{in}}$ and $\mathcal{D}_{\text{s}}$, the two sub-networks of the digital twin, i.e., the interference prediction network f_{in} and the signal prediction network f_{s}, are trained in a *supervised* manner. After the training process saturates, the digital twin is ready to interact with the RL agent.

Environment switching and virtual interaction: The switching from the actual environment to the digital twin is triggered based on multiple factors, for example, when the interferers are not transmitting signal or when the digital twin can provide accurate predictions, etc. As a result, after the switching is finished, the reward signal required by the RL agent will be provided by the trained digital twin instead of the actual environment. The agent keeps interacting with the digital twin until it does not improve, which marks the saturation of the agent learning and the end of the virtual interaction process.

Demand based switching and active data acquisition: The

Authorized licensed use limited to: ASU Library. Downloaded on March 24,2024 at 06:44:52 UTC from IEEE Xplore. Restrictions apply.

system might require executing the above steps multiple times, based on the achieved performance. The motivation of such repetition can be summarized as follows. From the model training perspective, the quality of the collected datasets, i.e., $\mathcal{D}_{\text{in}}$ and $\mathcal{D}_{\text{s}}$, has significant influence on the prediction accuracy of the trained digital twin. To be more specific, during the initial interaction process, most of the beams tried out by the agent are relatively random and hence have relatively poor quality in terms of SINR performance. This means that the datasets are, intuitively speaking, biased towards the "poor-quality" beams. As a result, the trained digital twin will have relatively inaccurate predictions on the beams that actually have better performance. The incurred residual prediction error will in turn influence the learning of the agent, leading to unsatisfactory performance.

However, as the policy of the RL agent gets improved over time, the actions performed by the agent, i.e., the beams, are more likely to be in the beam space where the achieved SINR is high. Therefore, it is advisable to switching back to the actual environment to re-collect data (through agent-environment interaction). Such active data acquisition can enhance the training datasets with "high-quality" beams. Using those better data samples to refine the parameters of the digital twin can help achieve higher prediction accuracy in the interested beam space, which further helps the learning of the agent. By alternatingly performing these steps, the system has higher chance to collect data samples that are more useful for the agent learning, which has the potential of further enhancing both sample efficiency and learning convergence. We show such interplay between the RL agent, actual environment and the digital twin in Fig. 1.

VI. Simulation Results

A. Deep Learning Models and Training Procedures

1) DRL agent architecture: Since the input of the actor network is the state and the output is the action, the size of both the input and output of the actor network is M, i.e., the number of antennas. The critic network takes in the state-action pair and outputs the predicted Q value and hence it has an input size of $2M$ and an output size of 1. Both the actor and critic networks have two hidden layers in our proposed architecture, with the size of the first hidden layer being 16 times of the input size and the size of the second hidden layer being 16 times of the output size in both networks. All the hidden layers are followed by the batch normalization layer for an efficient training experience and the Rectified Linear Unit (ReLU) activation layer. The output layer of the actor network is followed by a Tanh activation layer scaled by π to make sure that the predicted phases are within $(-\pi, \pi]$ interval. The output layer of the critic network is a linear layer. Moreover, we adopt the same DRL architecture for both solutions, regardless of having digital twin or not.

2) Digital twin architecture: We describe the two different architectures of the digital twin studied in this paper. Also, as the signal prediction network and the interference prediction network have identical architecture in both solutions (i.e., model-based solution and fully-connected neural network based solution), for brevity, we only use the interference prediction network as an example.

Signal model-based prediction network: As mentioned before in (14), the interference prediction network is essentially devised to take a quadratic form of the combining vector determined by a positive semi-definite matrix $\mathbf{Q}_{\text{in}}\mathbf{Q}_{\text{in}}^H$, leaving the matrix $\mathbf{Q}_{\text{in}}$ to be the model parameter. Moreover, $\mathbf{Q}_{\text{in}}$ has a shape of $M \times r_{\text{in}}$ with M being the number of antennas and r_{in} being a hyper-parameter. The choice of r_{in} is empirically guided by the following rules: (i) r_{in} should not be too large as it will increase the model complexity and hence the required amount of training data; (ii) r_{in} should not be too small as it will limit the expressive capability of the model, leading to unsatisfactory prediction accuracy.

Fully-connected neural network based prediction network: We adopt the fully-connected neural network with two hidden layers to be the interference prediction network. The input layer of the network has M neurons, which is equal to the number of antennas. The output layer of the network has only one neuron with linear activation. Both hidden layers have M' neurons. Similar to r_{in} in the model-based architecture, the selection of M' needs to strike a balance between model complexity and model expressive capability. Moreover, all the hidden layers are followed by the batch normalization layer and ReLU activation layer.

3) Training parameters: As mentioned before, the digital twin is trained in a supervised fashion, based on the collected power datasets, i.e., $\mathcal{D}_{\text{in}}$ and $\mathcal{D}_{\text{s}}$. Moreover, the interference prediction network and the signal prediction network are independently trained. However, for the same type of digital twin, i.e., either model-based or fully-connected neural network based, we adopt the same training parameters for interference and signal prediction networks. We summarize the detailed hyper-parameters used for training the digital twins in Table I.

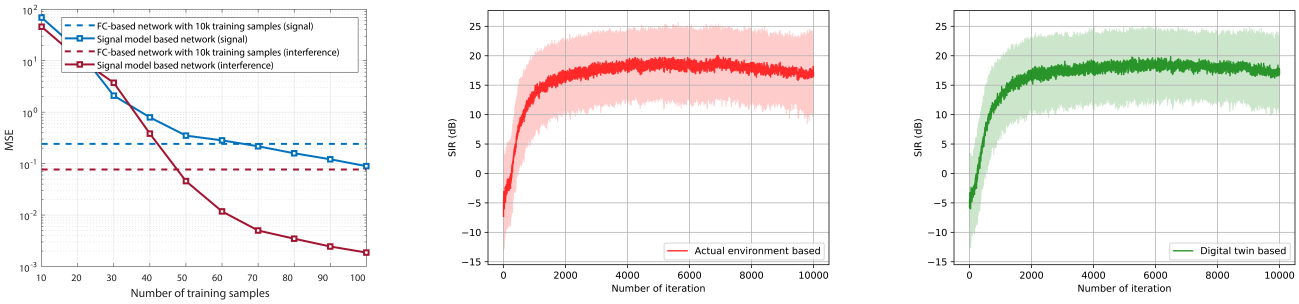
TABLE I
Hyper-parameters for digital twin training

Parameter	Model-based	FC-based
Batch size	512	512
Number of epochs	500	500
Optimizer	Adam	Adam
Initial learning rate	1×10^{-1}	1×10^{-2}
Learning rate schedule	$0.1@[50, 300, 400]$	$0.1@[100, 300, 400]$

B. Numerical Results

In this subsection, we provide the simulation results of the proposed digital twin-assisted interference-aware beam learning solutions. We first evaluate the prediction accuracy of the two proposed prediction network architectures, which provides insight on how much data samples are required in order to have a reasonable performance as well as the practicality of the solutions. We show the prediction accuracy of both the signal power and the interference power. As can be seen, the signal model-based architecture requires much less data samples to achieve higher prediction accuracy than the

Authorized licensed use limited to: ASU Library. Downloaded on March 24,2024 at 06:44:52 UTC from IEEE Xplore. Restrictions apply.



(a) Signal and interference prediction ($M = 8$) (b) Interaction with the actual environment (c) Interaction with the digital twin

Fig. 2. The performance of the proposed digital twin-assisted beam learning framework. In (a), we compare the proposed signal model-based digital twin design with the FC-based design to show the significant reduction on the required real measurements. In (b) and (c), we show the learning experience of the DRL agent when interacting with the actual environment and the trained digital twin, to highlight the efficacy of such "virtual" environment.

FC-based architecture trained with much more data samples. For instance, as indicated in Fig. 2(a), with only 50 samples, the signal model-based prediction architecture can achieve even more accurate interference prediction than the FC-based architecture trained with $10,000$ samples. **This saves almost 99.5% of the measurements, yielding a more sample-efficient solution for the practical system deployment.** Moreover, as there are more data samples, the prediction accuracy of the signal model-based architecture also gets improved quite significantly. Such performance is achieved by better leveraging the underlying signal relationships and hence the model parameters are essentially searched over a much smaller space. The trained digital twin is utilized to interact with the DRL agent, aiming to reduce the expensive actual measurements conducted by the hardware. In Fig. 2, we show the performance of the DRL agent when interacting with the actual environment as well as the digital twin. The training of the DRL agent is repeated for 100 times and the average performance as well as the standard deviation are reported in Fig. 2. We test the performance of a system with 8 antennas, and the digital twin is trained using $1,000$ data samples, i.e., $|\mathcal{D}_{\text{in}}| = |\mathcal{D}_{\text{s}}| = 1000$. As can be seen, the learning experience based on the digital twin is quite similar to that of the one based on the actual environment. This empirically shows the effectiveness of using the digital twin in training the DRL agent. As a result, although the DRL agent requires almost a total number of $5,000$ interactions with the environment to converge, **in the digital twin assisted learning framework, all these interactions are with the digital twin and hence the expensive evaluations on the real hardware are avoided.**

VII. Conclusion

In this paper, we developed a sample-efficient digital twin-assisted online interference-aware beam design framework. The proposed solution learns how to design beam patterns that can effectively manage interference, relying only on the power measurements and without any channel knowledge. The design of the digital twin leverages the underlying signal relationship, leading to a significant reduction on the required interactions with the actual environment. Moreover, it also facilitates other tasks such as interference-aware codebook learning, where the data sharing among different beam learning agents/engines becomes possible. The results highlight the efficacy of the trained digital twin in guiding the beam learning process.

VIII. Acknowledgment

This work is supported in part by the National Science Foundation under Grant No. 1923676 and by a U. S. Army research program under contract No. W911NF21C0015.

References

[1] R. Lorenz and S. Boyd, "Robust minimum variance beamforming," *IEEE Transactions on Signal Processing*, vol. 53, no. 5, pp. 1684–1696, 2005.
[2] D. Gesbert, S. Hanly, H. Huang, S. Shamai Shitz, O. Simeone, and W. Yu, "Multi-Cell MIMO Cooperative Networks: A New Look at Interference," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 9, pp. 1380–1408, 2010.
[3] H. Dahrouj and W. Yu, "Coordinated beamforming for the multicell multi-antenna wireless system," *IEEE Transactions on Wireless Communications*, vol. 9, no. 5, pp. 1748–1759, 2010.
[4] S. Smith, "Optimum phase-only adaptive nulling," *IEEE Transactions on Signal Processing*, vol. 47, no. 7, pp. 1835–1843, 1999.
[5] T. Van Luyen and T. Vu Bang Giang, "Interference Suppression of ULA Antennas by Phase-Only Control Using Bat Algorithm," *IEEE Antennas and Wireless Propagation Letters*, vol. 16, pp. 3038–3042, 2017.
[6] H. Steyskal, "Simple method for pattern nulling by phase perturbation," *IEEE Transactions on Antennas and Propagation*, vol. 31, no. 1, pp. 163–166, 1983.
[7] R. Davis, "Phase-only LMS and perturbation adaptive algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 1, pp. 169–178, 1998.
[8] Y. Zhang, M. Alrabeiah, and A. Alkhateeb, "Reinforcement Learning of Beam Codebooks in Millimeter Wave and Terahertz MIMO Systems," *IEEE Trans. on Communications*, vol. 70, no. 2, pp. 904–919, 2022.
[9] A. Alkhateeb, S. Jiang, and G. Charan, "Real-time digital twins: Vision and research directions for 6g and beyond," 2023. [Online]. Available: https://arxiv.org/abs/2301.11283
[10] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural networks*, vol. 2, no. 5, pp. 359–366, 1989.

Authorized licensed use limited to: ASU Library. Downloaded on March 24,2024 at 06:44:52 UTC from IEEE Xplore. Restrictions apply.