Deep Reinforcement Learning for Intelligent Reflecting Surfaces: Towards Standalone Operation

Abdelrahman Taha*, Yu Zhang*, Faris B. Mismar†, and Ahmed Alkhateeb*

**Arizona State University*, Tempe, AZ, USA, Emails: {a.taha, y.zhang, aalkhateeb}@asu.edu

†*The University of Texas at Austin*, Austin, TX, USA, Email: faris.mismar@utexas.edu

Abstract—The promising coverage and spectral efficiency gains of intelligent reflecting surfaces (IRSs) are attracting increasing interest. To adopt these surfaces in practice, however, several challenges need to be addressed. One of these main challenges is how to configure the reflecting coefficients on these passive surfaces without requiring massive channel estimation or beam training overhead. Earlier work suggested leveraging supervised learning tools to predict the IRS reflection matrices. While this approach has the potential of reducing the beam training overhead, it requires collecting large datasets for training the neural network models. In this paper, we propose a novel deep reinforcement learning framework for predicting the IRS reflection matrices with minimal beam training overhead. Simulation results show that the proposed online learning framework can converge to the optimal rate that assumes perfect channel knowledge. This represents an important step towards realizing a *standalone IRS operation*, where the surface configures itself without any control from the infrastructure.

Index Terms—reconfigurable intelligent surface, large intelligent surface, intelligent reflecting surface, smart reflect-array, beamforming, deep reinforcement learning

I. INTRODUCTION

The increasing demand for data rates from the massive number of devices motivates the need to develop novel system architectures that are both energy and spectrally efficient. For the past years, state of the art research has focused on leveraging large-scale MIMO systems, such as massive and millimeter wave (mmWave) MIMO at the base stations (BSs) and mobile users. To further improve the coverage and the energy efficiency of these systems, intelligent reflecting surfaces (IRSs) have been recently proposed and attracted massive interest [1]–[6]. IRSs consist of a huge number of passive reflecting elements whose function is to reflect the incident signal *intelligently* into the desired directions, through software-controllable phase shifts. Since the IRS reflection beamforming prediction requires the perfect/imperfect channel knowledge, the channel estimation is a crucial aspect for predicting the IRS interaction matrices. The massive number of passive IRS elements, however, impose the main challenge on acquiring the channel estimates; traditional channel estimation solutions will lead to either huge training overhead or prohibitive hardware complexity for the IRS architectures. Given an end goal of achieving harmonic co-existence between all the heterogeneous wireless systems, setting an objective of developing *fully-standalone* IRS architectures appears like the next step forward for reaching that end goal.

Prior work focused on proposing solutions for both the channel estimation and the reflection beamforming prediction problems [2], [7]–[9]. The authors in [2] proposed solutions to the channel/beam training overhead challenge leveraging tools from both compressive sensing and supervised deep learning. In [7], [8], a supervised deep learning framework is used for channel estimation by mapping the received pilots to the direct and the cascaded channels. In [9], an IRS channel estimation scheme based on a minimum variance unbiased estimator is proposed. The solutions in [2], [7]–[9], however, either considered supervised deep learning which requires a large dataset collection phase before training or assumed that the IRS is assisted/controlled by another base station/access point, not operating on its own.

This work presents a *novel* application of deep reinforcement learning in predicting the reflection coefficients of the IRS surfaces without requiring any prior training overhead. The main contributions of this paper can be summarized as follows.

- *A novel deep reinforcement learning (DRL) based solution* is proposed for predicting the best IRS interaction, where the IRS learns how to reflect the incident signals in the best possible way by adjusting its reflection matrix. This solution eliminates the need for collecting large training dataset, hence requires almost no beam training overhead.
- The proposed framework is directed more towards *standalone IRS operation*, where the IRS architecture is not controlled/assisted by any base station, but rather operating on its own while interacting with the environment, and without any initial training phase requirement.

Simulation results based on accurate 3D ray-tracing datasets show that the achievable rates of the proposed DRL based solution can converge close to the upper bound with an added value of almost no beam training overhead, as opposed to supervised learning based solutions.

Notation: $\mathbf{A}$ is a matrix, $\mathbf{a}$ is a vector, a is a scalar, and $\mathcal{A}$ is a set of vectors. $\mathrm{diag}(\mathbf{a})$ is a diagonal matrix with entries of $\mathbf{a}$ on its diagonal. $|\mathbf{A}|$ is the determinant of $\mathbf{A}$, $\mathbf{A}^T$ is its transpose, $[\mathbf{A}]_{r,:}$ is the rth row of $\mathbf{A}$, and $\mathbf{vec}(\mathbf{A})$ is a vector whose elements are the stacked columns of $\mathbf{A}$. $\mathbf{I}$ is the identity matrix. $\mathbf{A} \odot \mathbf{B}$ is the Hadamard product of $\mathbf{A}$ and $\mathbf{B}$. $\mathcal{N}(\mathbf{m}, \mathbf{R})$ is a complex Gaussian random vector with mean $\mathbf{m}$ and covariance $\mathbf{R}$. $\mathbb{E}[\cdot]$ is for expectation.
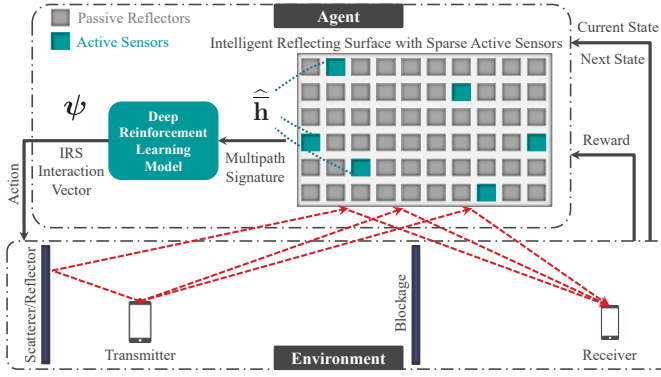
Fig. 1. The transmitter-receiver communication is assisted by an intelligent reflecting surface (IRS). The IRS is interacting with the incident signal through an interaction vector $\boldsymbol{\psi}$. Active channel sensors are randomly distributed over the IRS. These active elements have two modes of operation (i) a channel sensing mode where it is connected to the baseband to estimate the channels and (ii) a reflection mode where it just reflects the incident signal by applying a phase shift. The rest of the IRS elements are passive reflectors. The environment is represented by various scatterers, user locations, etc. The IRS acts as a reinforcement learning agent by acquiring a state and a reward from the environment and exerting an action back on the environment.

II. SYSTEM AND CHANNEL MODELS

A. System Model

Consider an OFDM-based system of K subcarriers where a single-antenna transmitter is communicating with a single-antenna receiver due to the assistance of an M-elements intelligent reflecting surface (IRS), as in Fig. 1. Let $\mathbf{h}_{\mathrm{T},k}, \mathbf{h}_{\mathrm{R},k} \in \mathbb{C}^{M \times 1}$ denote the channels from the transmitter/receiver to the IRS at the kth subcarrier. s_k is the transmit signal, where $\mathbb{E}[|s_k|^2] = \frac{P_{\mathrm{T}}}{K}$. P_{T} is the total transmit power. $\boldsymbol{\Psi}$ denotes the IRS interaction diagonal matrix. $n_k \sim \mathcal{N}_{\mathbb{C}}(0, \sigma_n^2)$ is the receive noise. The receive signal at the receiver can be expressed as

$$y_k = \mathbf{h}_{\mathrm{R},k}^T \boldsymbol{\Psi} \, \mathbf{h}_{\mathrm{T},k} s_k + n_k, \tag{1}$$

$$\overset{(a)}{=} (\mathbf{h}_{\mathrm{R},k} \odot \mathbf{h}_{\mathrm{T},k})^T \boldsymbol{\psi} \, s_k + n_k, \tag{2}$$

where $\boldsymbol{\psi}$ is the IRS interaction vector, such that $\boldsymbol{\Psi} = \mathrm{diag}(\boldsymbol{\psi})$. Assume an IRS architecture of RF phase shifters, every interaction factor can be represented as $[\boldsymbol{\psi}]_m = e^{j\phi_m}$, hence the choice of an interaction vector is constrained to a predefined codebook $\mathcal{P}$. Adopting the IRS architecture proposed in [2] and illustrated in Fig. 1, *active* elements are randomly distributed over the IRS. The *sampled* channel vector from the transmitter/receiver to the IRS active elements, $\overline{\mathbf{h}}_{\mathrm{T},k}, \overline{\mathbf{h}}_{\mathrm{R},k} \in \mathbb{C}^{\overline{M} \times 1}$, can be expressed as $\overline{\mathbf{h}}_{\mathrm{T},k} = \mathbf{G}_{\mathrm{IRS}}\mathbf{h}_{\mathrm{T},k}$ and $\overline{\mathbf{h}}_{\mathrm{R},k} = \mathbf{G}_{\mathrm{IRS}}\mathbf{h}_{\mathrm{R},k}$, where $\mathbf{G}_{\mathrm{IRS}}$ is an $\overline{M} \times M$ selection matrix that selects the entries corresponding to the active IRS elements. Finally, the overall IRS *sampled* channel vector can be expressed as $\overline{\mathbf{h}}_k = \overline{\mathbf{h}}_{\mathrm{T},k} \odot \overline{\mathbf{h}}_{\mathrm{R},k}$.

B. Channel Model

A wideband geometric channel model is adopted [10]. Consider a transmitter-IRS channel, $\mathbf{h}_{\mathrm{T},k}$, (and similarly for the IRS-receiver channel) consisting of L clusters. Each cluster contributes with one ray from the transmitter to the IRS.

The ray parameters are: azimuth/elevation angles of arrival, $\theta_\ell, \phi_\ell \in [0, 2\pi)$; complex coefficient $\alpha_\ell \in \mathbb{C}$; time delay $\tau_\ell \in \mathbb{R}$. The transmitter-IRS path loss is denoted by ρ_{T}. The pulse shaping function, with T_S-spaced signaling, is defined as $p(\tau)$ at τ seconds. The frequency-domain channel vector, $\mathbf{h}_{\mathrm{T},k}$, can then be defined as

$$\mathbf{h}_{\mathrm{T},k} = \sqrt{\frac{M}{\rho_{\mathrm{T}}}} \sum_{d=0}^{D-1} \sum_{\ell=1}^{L} \alpha_\ell \, \mathbf{a}(\theta_\ell, \phi_\ell) \, p(dT_S - \tau_\ell) \, e^{-j\frac{2\pi k}{K}d}, \tag{3}$$

where $\mathbf{a}(\theta_\ell, \phi_\ell) \in \mathbb{C}^{M \times 1}$ is the IRS array response vector. Assume a block-fading channel model, where $\mathbf{h}_{\mathrm{T},k}$ and $\mathbf{h}_{\mathrm{R},k}$ are assumed to stay constant over the channel coherence time.

III. PROBLEM FORMULATION

Given the objective of maximizing the achievable rate at the receiver, our problem is then to find the optimal interaction vector, $\boldsymbol{\psi}^\star$, that solves

$$\boldsymbol{\psi}^\star = \arg\max_{\boldsymbol{\psi} \in \mathcal{P}} \sum_{k=1}^{K} \log_2\left(1 + \mathsf{SNR} \left|(\mathbf{h}_{\mathrm{T},k} \odot \mathbf{h}_{\mathrm{R},k})^T \boldsymbol{\psi}\right|^2\right), \tag{4}$$

to achieve the optimal rate $R^\star$ defined as

$$R^\star = \max_{\boldsymbol{\psi} \in \mathcal{P}} \frac{1}{K} \sum_{k=1}^{K} \log_2\left(1 + \mathsf{SNR} \left|(\mathbf{h}_{\mathrm{T},k} \odot \mathbf{h}_{\mathrm{R},k})^T \boldsymbol{\psi}\right|^2\right). \tag{5}$$

Unfortunately, there is no closed-form solution for the optimization problem in (4) due to the quantized codebook constraint and the use of one interaction vector $\boldsymbol{\psi}$ fixed over all subcarriers. Accordingly, finding the optimal interaction vector for the IRS, $\boldsymbol{\psi}^\star$, requires an exhaustive search over the codebook $\mathcal{P}$. This search, however, leads either to prohibitive training overheard, hardware complexity, or power consumption, as detailed in [2]. Our objective is then to find an efficient solution for the IRS systems capable of approaching the optimal rate in (5) with **almost no beam training overhead** and with an **energy-efficient hardware**. In the next section, we propose a *novel* application of deep reinforcement learning in the interaction prediction problem of intelligent reflecting surfaces. This solution eliminates the need for collecting large training datasets as opposed to the supervised learning solution proposed in [2]. The supervised learning solution, however, approaches the optimal rate with fewer iterations.

IV. DEEP REINFORCEMENT LEARNING BASED IRS INTERACTION PREDICTION

A. Key Idea

From (4), the optimal interaction vector is a function of the channels between the two communication ends and the IRS. To avoid the prohibitive overhead of estimating the full IRS channels, the optimal interaction vector choice can be mapped to the surrounding *environment*, which the full IRS channels inherently describe. Modeling the various elements of

the environment, mathematically, is notoriously complicated. In contrast, leveraging an awareness of the environment using a multipath signature [10] can be sufficient. In such case, deep reinforcement learning models can be adopted to learn the mapping function from multipath signatures to the optimal interaction vectors as illustrated in Fig. 1. The IRS active elements play a crucial role in capturing one form of multipath signatures: the *sampled* channels, $\overline{\mathbf{h}}_{\mathrm{T},k}, \overline{\mathbf{h}}_{\mathrm{R},k}$. Fortunately, estimating the sampled channel vectors can be accomplished with a few pilot signals; i.e., negligible training overhead. This solution also involves energy-efficient low-complexity hardware architectures (few sparse active IRS elements) [2].

B. Proposed Solution

The proposed deep reinforcement learning (DRL) based IRS interaction prediction approach operates in two modes: (I) the agent interaction and (II) the agent learning, as in Algorithm 1. The IRS interchanges between these two tasks continuously; this cycle repeats over time and across multiple users.

Task I: Agent Interaction: The IRS interaction with the *environment* can be outlined as follows: the IRS observes the current *state*, s, of the environment and takes an *action*, a, predicated upon the observed state. The IRS then receives a *reward*, r, for the action taken and a new *state* observation, s', from the environment. Once the experience is acquired, $\langle s, a, r, s' \rangle$, the IRS trains the DRL model using current and past experiences, in the second task.

Let the term "experience" indicates the information captured in one learning episode, and define the concatenated *sampled* channel vector as

$$\overline{\mathbf{h}} = \mathbf{vec}\left(\left[\overline{\mathbf{h}}_1, \overline{\mathbf{h}}_2, \ldots, \overline{\mathbf{h}}_K\right]\right). \tag{6}$$

Assume that the one learning episode occurs every coherence block and let T be the maximum number of episodes, $\overline{\mathbf{h}}(t)$ denotes the concatenated *sampled* channel vector at the tth episode, where $t = 1, \ldots, T$. Task I steps are summarized as follows.

1. Sampled channel estimation (lines 3,13): The transmitter and receiver transmit two orthogonal uplink pilots. The IRS active elements will receive these pilots and estimate the *sampled* channel vectors to construct the multipath signature.

$$\widehat{\overline{\mathbf{h}}}_{\mathrm{T},k}(t) = \overline{\mathbf{h}}_{\mathrm{T},k}(t) + \mathbf{v}_k, \widehat{\overline{\mathbf{h}}}_{\mathrm{R},k}(t) = \overline{\mathbf{h}}_{\mathrm{R},k}(t) + \mathbf{w}_k, \tag{7}$$

$$\widehat{\overline{\mathbf{h}}}_k(t) = \widehat{\overline{\mathbf{h}}}_{\mathrm{T},k}(t) \odot \widehat{\overline{\mathbf{h}}}_{\mathrm{R},k}(t), \tag{8}$$

$$\widehat{\overline{\mathbf{h}}}(t) = \mathbf{vec}\left(\left[\widehat{\overline{\mathbf{h}}}_1(t), \widehat{\overline{\mathbf{h}}}_2(t), \ldots, \widehat{\overline{\mathbf{h}}}_K(t)\right]\right). \tag{9}$$

where $\mathbf{v}_k, \mathbf{w}_k \sim \mathcal{N}_{\mathbb{C}}\left(\mathbf{0}, \sigma_n^2 \mathbf{I}\right)$ are the receive noise vectors.

2. Interaction prediction (lines 5-10): The multipath signature is used to predict the interaction vector. To account for exploration (i.e., randomly sampling from the action space) besides exploitation (i.e., using prior learning experience), the factor ϵ is introduced such that an interaction vector can be randomly chosen out of the codebook $\mathcal{P}$ with ϵ probability.

Otherwise, the interaction vector is predicted from the current network. After that, the interaction vector chosen reflects the transmitted data from the transmitter.

3. Feedback reception (lines 11,12): The IRS receives feedback from the receiver indicating the achievable rate, $R(t)$, attained by using the interaction vector, which is defined as

$$R(t) =$$
$$\frac{1}{K} \sum_{k=1}^K \log_2 \left(1 + \mathsf{SNR} \left| (\mathbf{h}_{\mathrm{T},k}(t) \odot \mathbf{h}_{\mathrm{R},k}(t))^T \boldsymbol{\psi}_a \right|^2 \right). \tag{10}$$

After that, the rate is quantized based on a threshold level, such that $R_{\mathrm{Q}}(t) = 1$ if $R(t) > R^{\mathrm{TH}}$; otherwise, $R_{\mathrm{Q}}(t) = -1$. Reward clipping is substantial for learning convergence [11].

Algorithm 1 Deep Reinforcement Learning Based IRS Interaction Prediction

---

Input: Reflection beamforming codebook $\mathcal{P}$.
Output: Trained network $Q(s, a|\boldsymbol{\theta})$.
1: **Initialization:** Network $Q(s, a|\boldsymbol{\theta})$, replay buffer $\mathcal{D}$.
2: **repeat**
3:      IRS receives two pilots to estimate $\widehat{\overline{\mathbf{h}}}(1)$. ▷ Current state
4:      **for** episode $t = 1$ **to** T **do**   ▷ For every episode
         Task I: Agent Interaction
5:          Sample $\xi \sim \mathrm{Uniform}\,(0, 1)$
6:          **if** $\xi \leq \epsilon$ **then**   ▷ Select action
7:              Select interaction vector, $\boldsymbol{\psi}(t) \in \mathcal{P}$ at random.
8:          **else**
9:              Select interaction vector, $\boldsymbol{\psi}(t) = \arg\max_{a'} Q(s, a'|\boldsymbol{\theta})$.
10:          IRS reflects using $\boldsymbol{\psi}(t)$ beam. ▷ Carry out action
11:          IRS receives the feedback $R(t)$. ▷ Observe reward
12:          IRS quantizes the reward, $R_{\mathrm{Q}}(t) \in \{\pm 1\}$.
13:          IRS receives two pilots to estimate $\widehat{\overline{\mathbf{h}}}(t+1)$ ▷ Next state
         Task II: Agent Learning
14:          $\langle s, a, r, s' \rangle \leftarrow \left\langle \widehat{\overline{\mathbf{h}}}(t), \boldsymbol{\psi}(t), R_{\mathrm{Q}}(t), \widehat{\overline{\mathbf{h}}}(t+1) \right\rangle$.
15:          Store the experience $\langle s, a, r, s' \rangle$ in $\mathcal{D}$.
16:          Minibatch experiences from $\mathcal{D}$ for training.
17:          feedforward s to calculate $\widehat{\mathbf{R}}(t) \leftarrow Q(s, a|\boldsymbol{\theta})\,\forall a$.
18:          feedforward s' to calculate $\Gamma \leftarrow \max_{a'} Q(s', a'|\boldsymbol{\theta})$
         and calculate $a^\star \leftarrow \arg\max_{a'} Q(s', a'|\boldsymbol{\theta})$.
19:          Construct the target vector, $\overline{\mathbf{R}}(t)$:
20:          $[\overline{\mathbf{R}}(t)]_{a^\star} \leftarrow R_{\mathrm{Q}}(t) + \gamma\Gamma$,
21:          $[\overline{\mathbf{R}}(t)]_{a' \neq a^\star} \leftarrow \left[\widehat{\mathbf{R}}(t)\right]_{a' \neq a^\star}, a' \in \{1, \ldots, |\mathcal{P}|\}$.
22:          Perform SGD on $\mathsf{MSE}\left(\overline{\mathbf{R}}(t), \widehat{\mathbf{R}}(t)\right)$ to find $\boldsymbol{\theta}^\star$.
23:          Update network weights $\boldsymbol{\theta}(t) \leftarrow \boldsymbol{\theta}^\star$.
24:          Decrease ϵ gradually.
25:          $s \leftarrow s'$. ▷ Assign next state to current state
26: **until** reaching a terminal goal

Task II: Agent Learning: The IRS leverages the acquired experiences to train the DRL model. Task II steps are summarized as follows.

1. Constructing a new experience (lines 14,15): The new experience acquired is now stored in the experience replay buffer $\mathcal{D}$ for the training of the deep Q-network [12].

2. Model training (lines 16-23): The deep Q-network is now trained to minimize the prediction loss. To do so, we use the stochastic gradient descent algorithm (SGD). The training operates sequentially using minibatchs from the replay buffer
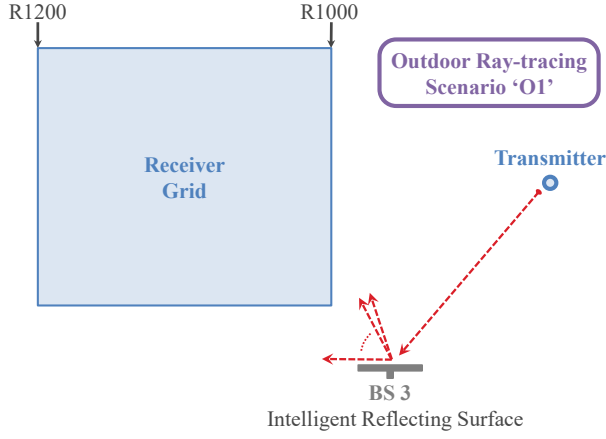
Fig. 2. The adopted ray-tracing scenario where an IRS is reflecting the signal received from one fixed transmitter to a receiver. The receiver is selected from a grid of candidate locations. This scenario is generated using Remcom Wireless InSite [15], and is available on the DeepMIMO dataset [16].

TABLE I
THE ADOPTED DEEPMIMO DATASET PARAMETERS

DeepMIMO Dataset Parameter	Value
Frequency band	3.5 GHz
Active BSs	3
Active users (receivers)	From row R1000 to row R1200
Active user (transmitter)	row R850 column 90
Number of BS Antennas	$(M_x, M_y, M_z) = (1, 40, 10)$
Antenna spacing	0.5λ
System bandwidth	100 MHz
Number of OFDM subcarriers	512
OFDM sampling factor	1
OFDM limit	64
Number of paths	$1, 15$

$\mathcal{D}$. It learns how to map an input state (*sampled* channel vector) to an output action (interaction vector).

C. Machine Learning Design

Input Representation: the concatenated *sampled* channel vector, $\widehat{\mathbf{h}}$, is the input to the deep Q-network. The normalization method used is a simple per-dataset scaling [13]; all samples are normalized by the maximum absolute value over the whole input data. This method preserves distance information encoded in the multipath signatures. Each complex entry of the input data is split into real and imaginary values, doubling the dimensionality of each input vector to $2K\overline{M}$.

Q-Network Architecture: The Q-network is designed as a Multi-Layer Perceptron network of U layers. The first $U-1$ of them alternate between fully-connected and rectified linear unit layers and the last one (output layer) is a fully-connected layer. The uth layer in the network has a stack of A_u neurons. Two deep Q-networks are used for training stability [14].

Training Loss Function: Given the objective of predicting the best interaction vector (with the highest achievable rate), the model is trained using a regression loss function. At the tth episode, the training is guided through minimizing the loss function, $\text{MSE}\left(\overline{\mathbf{R}}(t), \widehat{\mathbf{R}}(t)\right)$, which is the mean-squared-error between the desired and the predicted output, $\overline{\mathbf{R}}(t)$ and $\widehat{\mathbf{R}}(t)$.

V. SIMULATION RESULTS

A. Simulation Setup

The DeepMIMO dataset in [16] is adopted to generate the channels based on the outdoor ray-tracing scenario 'O1'. The dataset parameters are summarized in Table I. The transmitter's position is fixed while the receiver can take any random position in a specified x-y grid, as illustrated in Fig. 2. We select BS 3 to be the IRS. For a detailed description of the simulation setup, please refer to the simulation setup in [2].

Deep reinforcement learning parameters: We adopt the DRL model described in Section IV-C. States are represented by the normalized concatenated *sampled* channel of each user pair, and actions are represented by each candidate interaction vector, $\psi \in \mathcal{P}$. To reduce the Q-network complexity, we input the normalized *sampled* channels only at the first 64 subcarriers. The neural network architecture consists of four fully-connected layers of $4096, 16384, 16384, 4096$ nodes, respectively. Given the size of the receiver x-y grid, the DRL dataset has 36200 data points. We split this dataset into two sets: training and testing sets, with 70% and 30% of the points, respectively. We consider a replay buffer of 8192 samples and a batch size of 512 samples. ϵ starts from 0.99 and decreases gradually by a factor of 0.5% every 40 training iterations till it reaches 0.1. $\gamma = 0$. $R^{\text{TH}} = 8.9$ bps/Hz is set to the min-max rate of the dataset.

B. Achievable Rates with Deep Reinforcement Learning

Fig. 3 illustrates the achievable rate of both the proposed DRL based solution and the supervised deep learning (SL) based solution in [2], using 4 active elements with $L \in \{1, 15\}$ channel paths. Their performances are compared to the upper bound with perfect full channel knowledge, calculated according to (5). As shown, the proposed DRL solution is capable of approaching the optimal rate with more training samples than the one needed by the SL solution. **In contrast, the proposed DRL solution uses only one beam for each training episode, which constitute almost 0.3% of the beams used by the SL solution in the training phase (400 beams).** This emphasizes the efficiency of the DRL solution in operating with almost no beam training overhead.

Another candidate approach for refining the DRL prediction is to use the trained DRL model in predicting the most promising k_{B} beams. Then, these beams are used for beam training to identify the best beam that will be utilized for the rest of the coherence block. Fig. 4 illustrates the achievable rate of the proposed DRL based solution compared to the upper bound, at different values of k_{B} $(1, 3)$, using 4 active elements. As demonstrated, the beam training of the promising k_{B} beams achieves better performance than just relying on the best network-predicted beam to reflect the incident signals. To test the effectiveness of the proposed framework, we examined another variant of the algorithm by updating its reward policy such that $R_{\text{Q}}(t) = 1$ if $R(t) = R^{\star}(t)$; otherwise, $R_{\text{Q}}(t) = -1$,
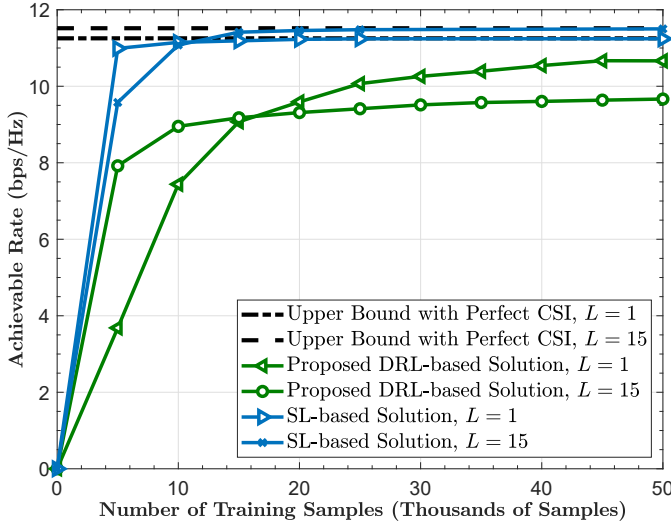
Fig. 3. The achievable rates of both the proposed deep reinforcement learning (DRL) solution and the supervised deep learning (SL) solution in [2], are compared to the upper bound, using $\overline{M} = 4$ active elements for a 3.5GHz scenario with $L \in \{1, 15\}$ channel path(/s). The upper bound, $R^\star$ in (5), assumes perfect channel knowledge. The figure shows the potential of the proposed DRL solution in approaching the optimal rate with almost no beam training overhead and a small fraction of the IRS elements to be active.

Fig. 4. The achievable rate of the proposed DRL based approach is compared to the upper bound $R^\star$, using $\overline{M} = 4$ active elements with $L = 15$ channel paths. The figure illustrates the achievable rate gain when the beams selected by the deep reinforcement learning model are further refined through beam training over k_B beams.

as illustrated in Fig. 4. The proposed DRL solution under this ideal rewarding assumption can converge to the optimal rate. This indicates that the small gap between the performance of the proposed solution and the upper bound can be explained by the practical assumptions of using threshold-based rewards and operating in an environment with 15 channel paths. These results show the gains from exploring deep reinforcement learning frameworks to develop *standalone* IRS architectures.

VI. CONCLUSION

For an IRS-assisted wireless communication systems, we developed an efficient solution for predicting the IRS inter-action matrices. Given the objective of developing standalone IRS architectures, the proposed solution exploits deep rein-forcement learning frameworks for the IRS to learn how to predict, on its own, the optimal interaction matrices directly from the sampled channel knowledge. This solution does not require an initial dataset collection phase as opposed to the supervised learning based solutions. Simulation results based on accurate ray-tracing channels showed that the proposed solution can converge near the optimal data rates with almost no beam training overhead and with few active elements.

REFERENCES

[1] C. Huang, S. Hu, G. C. Alexandropoulos, A. Zappone, C. Yuen, R. Zhang, M. Di Renzo, and M. Debbah, "Holographic MIMO Surfaces for 6G Wireless Networks: Opportunities, Challenges, and Trends," *arXiv preprint arXiv:1911.12296*, 2019.

[2] A. Taha, M. Alrabeiah, and A. Alkhateeb, "Enabling Large Intelligent Surfaces with Compressive Sensing and Deep Learning," *arXiv preprint arXiv:1904.10136*, Apr 2019.

[3] E. Basar, M. Di Renzo, J. De Rosny, M. Debbah, M.-S. Alouini, and R. Zhang, "Wireless Communications Through Reconfigurable Intelligent Surfaces," *IEEE Access*, vol. 7, pp. 116 753–116 773, 2019.
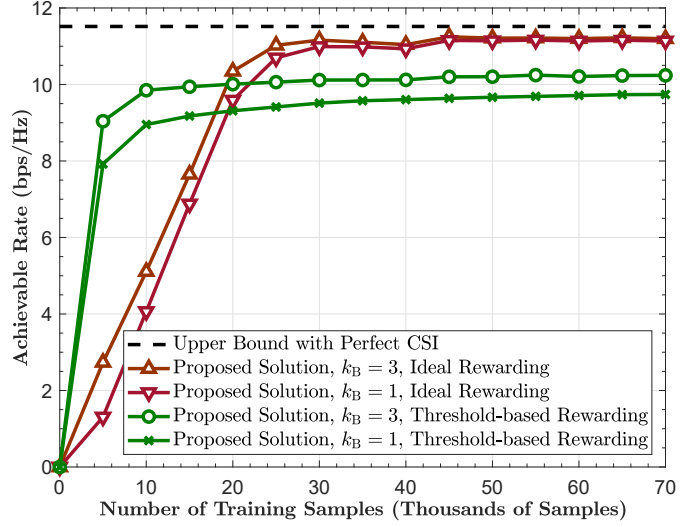
[4] Y.-C. Liang, R. Long, Q. Zhang, J. Chen, H. V. Cheng, and H. Guo, "Large Intelligent Surface/Antennas (LISA): Making Reflective Radios Smart," *arXiv preprint arXiv:1906.06578*, 2019.

[5] X. Yuan, Y.-J. Zhang, Y. Shi, W. Yan, and H. Liu, "Reconfigurable-Intelligent-Surface Empowered 6G Wireless Communications: Challenges and Opportunities," *arXiv preprint arXiv:2001.00364*, 2020.

[6] J. Zhao, "A Survey of Reconfigurable Intelligent Surfaces: Towards 6G Wireless Communication Networks with Massive MIMO 2.0," *arXiv preprint arXiv:1907.04789*, 2019.

[7] C. Huang, G. C. Alexandropoulos, C. Yuen, and M. Debbah, "Indoor Signal Focusing with Deep Learning Designed Reconfigurable Intelligent Surfaces," in *2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. IEEE, 2019, pp. 1–5.

[8] A. M. Elbir, A. Papazafeiropoulos, P. Kourtessis, and S. Chatzinotas, "Deep Channel Learning For Large Intelligent Surfaces Aided mm-Wave Massive MIMO Systems," *arXiv preprint arXiv:2001.11085*, 2020.

[9] T. L. Jensen and E. De Carvalho, "An Optimal Channel Estimation Scheme for Intelligent Reflecting Surfaces Based on a Minimum Variance Unbiased Estimator," *arXiv preprint arXiv:1909.09440*, 2019.

[10] A. Alkhateeb, S. Alex, P. Varkey, Y. Li, Q. Qu, and D. Tujkovic, "Deep Learning Coordinated Beamforming for Highly-Mobile Millimeter Wave Systems," *IEEE Access*, vol. 6, pp. 37 328–37 348, 2018.

[11] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level Control through Deep Reinforcement Learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[12] F. B. Mismar, B. L. Evans, and A. Alkhateeb, "Deep Reinforcement Learning for 5G Networks: Joint Beamforming, Power Control, and Interference Coordination," *IEEE Transactions on Communications*, 2019.

[13] Y. Zhang, M. Alrabeiah, and A. Alkhateeb, "Deep Learning for Massive MIMO with 1-Bit ADCs: When More Antennas Need Fewer Pilots," *arXiv preprint arXiv:1910.06960*, 2019.

[14] H. Van Hasselt, A. Guez, and D. Silver, "Deep Reinforcement Learning with Double Q-learning," in *Proc. of AAAI conference on artificial intelligence*, 2016.

[15] Remcom, "Wireless InSite," http://www.remcom.com/wireless-insite.

[16] A. Alkhateeb, "DeepMIMO: A Generic Deep Learning Dataset for Millimeter Wave and Massive MIMO Applications," in *Proc. of Information Theory and Applications Workshop (ITA)*, San Diego, CA, Feb 2019, pp. 1–8. [Online]. Available: https://www.deepmimo.net/