Active Sensing for Two-Sided Beam Alignment Using Ping-Pong Pilots

Tao Jiang, Foad Sohrabi and Wei Yu
Electrical and Computer Engineering Department
University of Toronto, ON M5S 3G4, Canada
Emails: taoca.jiang@mail.utoronto.ca, {fsohrabi, weiyu}@ece.utoronto.ca

2022 56th Asilomar Conference on Signals, Systems, and Computers | 978-1-6654-5906-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/IEEECONF56349.2022.10051966

***Abstract*—Beam alignment is an important task for millimeter-wave (mmWave) communication, because constructing aligned narrow beams both at the transmitter (Tx) and the receiver (Rx) is crucial to compensate for the significant path loss in very high-frequency bands. However, beam alignment is also a highly nontrivial task, because the hybrid beamforming architecture typical of large antenna arrays allows only low-dimensional measurements of the high-dimensional channel. This paper considers a two-sided beam alignment problem based on an alternating ping-pong pilot scheme between Tx and Rx over multiple rounds without explicit feedback. We propose a deep active sensing framework in which two long short-term memory (LSTM) based neural networks are employed to learn the adaptive sensing strategies and to produce the final aligned beamformers at both sides. In the proposed ping-pong protocol, the Tx and the Rx alternatively send pilots so that both sides can leverage local observations to sequentially design their respective sensing and data transmission beamformers. Numerical experiments demonstrate significant and interpretable performance improvement. The proposed strategy works well even for the challenging multipath channel environments.**

I. Introduction

Millimeter-wave (mmWave) is envisioned to be a key enabler for high data rate transmission in the next generation of communication systems [1], [2]. In mmWave systems, it is crucial to deploy large antenna arrays in order to focus narrow beams to compensate for the considerable signal attenuation at high frequencies. However, due to the cost and energy consumption constraints, the mmWave transceivers are generally equipped with only a limited number of radio-frequency (RF) chains [3], which means only low-dimensional signals can be observed. This makes it difficult to estimate the high-dimensional channel and to find the aligned beams for data transmission so that the signal-to-noise ratio (SNR) is maximized. This paper aims to alleviate the pilot training overhead for such a two-sided beam alignment problem in which two multi-antenna transceivers need to find the optimal beamforming direction based on a limited number of low-dimensional measurements designed by the two transceivers.

The two-sided beam alignment problem is highly nontrivial for several reasons. First, the problem involves sensing a high-dimensional channel via the low-dimensional observations through the limited number of RF chains both at the Tx and Rx. The design of such analog sensing vectors is not an easy problem to solve analytically. Second, because the pilots are transmitted over multiple stages, the sensing vectors can be designed as functions of the observations in the previous rounds. Such an *active* sensing strategy can significantly improve the eventual beamforming gain [4]–[6], but the optimal design of the sensing strategy involves sequential exploration of the channel landscape and is extremely challenging. Third, this paper tackles the two-sided beam alignment problem with the Tx and the Rx actively designing their beamformers at the same time. Unlike the one-sided scenario [4]–[6], two-sided beam alignment typically requires feedback between the transceivers. But coordination and feedback are not easy to realize before the beam alignment is achieved.

The two-sided beam alignment problem has been investigated in [7]–[10] using nonadaptive sensing beamformers. To reduce the pilot training overhead further, iterative search algorithms have been proposed to actively design the sensing beamformers based on the historical observations [7], [11]–[14]. In particular, [13], [14] propose to construct sensing beamformers adaptively according to a hierarchical codebook, from which the next sensing beamformers are chosen, based on the current posterior distribution of the angle of arrivals (AoAs) and angle of departures (AoDs). For one-sided beam alignment, [5] proposes a codebook-free approach to map the posterior distribution to the next sensing vector by a fully connected deep neural network (DNN). However, calculating the posterior distribution is computationally feasible only for the single-path channel model. Instead of using the posterior distribution, a follow-up paper [6] proposes a deep learning framework based on the long short-term memory (LSTM) neural network architecture to summarize the information contained in the observations automatically, which further improves the one-sided beam alignment performance and is applicable to multipath channel models. The main goal of the current paper is to generalize the approach of [6] to two-sided beam alignment.

This paper proposes a novel deep active sensing framework to tackle the two-sided beam alignment problem for a mmWave communication link. The proposed framework involves a ping-pong pilot transmission scheme, in which the Tx and the Rx alternatively transmit and receive pilots through the beamformers designed by their respective active sensing units based on the pilots received so far. The use of the ping-pong pilot strategy eliminates the need for feedback between the transceivers. To account for the sequential nature of the learning task, this paper proposes to utilize a recurrent

neural network (RNN) with long short-term memory (LSTM) as the active sensing units at both the Tx and Rx to efficiently extract the relevant information from the received pilots over multiple stages to design the Tx and Rx beamformers both for the sensing phase and for the final data transmission phase. Overall, the proposed design is shown to significantly reduce the pilot overhead and to enhance the overall SNR for the mmWave link, while producing interpretable beamforming patterns.

II. Two-Sided Beam Alignment

A. System Model

Consider a mmWave MIMO communication system consisting of a Tx, also called agent A, with $M_{\rm t}$ antennas configured as a uniform linear array (ULA), and a Rx, also called agent B, with $M_{\rm r}$ antennas as a ULA. The Tx and the Rx each have a single RF chain. Let $\boldsymbol{w}_{\rm t} \in \mathbb{C}^{M_{\rm t}}$ and $\boldsymbol{w}_{\rm r} \in \mathbb{C}^{M_{\rm r}}$ denote the beamforming vectors at the Tx and the Rx side, respectively. Without loss of generality, the transmitted signal is subject to a power constraint and the beamformers are normalized such that $\|\boldsymbol{w}_{\rm t}\|_2 = \|\boldsymbol{w}_{\rm r}\|_2 = 1$. To establish a reliable link between the Tx and the Rx, the beamforming vectors $\{\boldsymbol{w}_{\rm t}, \boldsymbol{w}_{\rm r}\}$ should be jointly optimized according to the channel state information (CSI) so that the achievable rate (or equivalently the SNR) of the communication link is maximized.

Let matrix $\boldsymbol{G} \in \mathbb{C}^{M_{\rm t} \times M_{\rm r}}$ denote the uplink channel matrix from the Rx to the Tx, then the downlink channel is denoted by $\boldsymbol{G}^{\sf H}$, where we assume the system operates in the time-division duplex (TDD) mode with channel reciprocity. We assume a block fading channel model, where the channel remains constant over a coherence interval but changes independently over different coherence intervals. To capture the intrinsic sparse nature of the mmWave propagation environments, a mmWave channel is typically modeled by a sparse multipath channel as follows [3]:

$$\boldsymbol{G} = \sum_{i=1}^{L_{\rm p}} \alpha_i \boldsymbol{a}_{\rm t}(\phi_{\rm t}^i)\boldsymbol{a}_{\rm r}^{\sf H}(\phi_{\rm v}^i), \tag{1}$$

where $L_{\rm p}$ denotes the number of paths, α_i denotes the complex fading coefficients of the i-th path, $\{\phi_{\rm t}^i, \phi_{\rm r}^i\}$ denotes the set of AoA and AoD corresponding to the i-th path, and $\boldsymbol{a}_{\rm t}(\cdot), \boldsymbol{a}_{\rm r}(\cdot)$ are the steering vectors as given by the following (assuming half-wavelength antenna spacing):

$$[\boldsymbol{a}_{\rm t}(\phi_{\rm t}^i)]_n = e^{j\pi(n-1)\sin(\phi_{\rm t}^i)}, \quad n = 1, \cdots, M_{\rm t}, \tag{2a}$$

$$[\boldsymbol{a}_{\rm r}(\phi_{\rm r}^i)]_n = e^{j\pi(n-1)\sin(\phi_{\rm r}^i)}, \quad n = 1, \cdots, M_{\rm r}. \tag{2b}$$

Let $x \in \mathbb{C}$ with $\mathbb{E}[|x|^2] = P$ denote the intended data symbol, then the received signal at the Rx can be expressed as

$$r = \boldsymbol{w}_{\rm r}^{\sf H}\boldsymbol{G}^{\sf H}\boldsymbol{w}_{\rm t}x + n, \tag{3}$$

downlink

where $n \sim \mathcal{CN}(0, \sigma_0^2)$ is the additive Gaussian noise. In order to maximize the transmission rate, the beamforming vectors should be designed to maximize the beamforming gain $|\boldsymbol{w}_{\rm r}^{\sf H}\boldsymbol{G}^{\sf H}\boldsymbol{w}_{\rm t}|^2$.

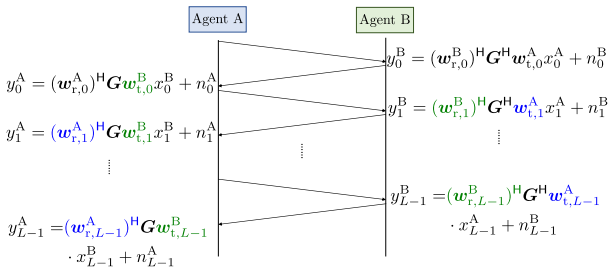

Fig. 1: Proposed ping-pong pilot training protocol. The sensing beamformers actively designed at agent A and agent B are highlighted as blue (e.g., $\boldsymbol{w}_{{\rm t},1}^{\rm A}$, $\boldsymbol{w}_{{\rm r},1}^{\rm A}$) and green (e.g., $\boldsymbol{w}_{{\rm t},0}^{\rm B}$, $\boldsymbol{w}_{{\rm r},1}^{\rm B}$), respectively. The initial sensing vectors $\boldsymbol{w}_{{\rm t},0}^{\rm A}$, $\boldsymbol{w}_{{\rm r},0}^{\rm A}$ and $\boldsymbol{w}_{{\rm r},0}^{\rm B}$ are fixed and can be learned from the channel and noise distributions in the proposed active sensing framework.

Given perfect CSI $\boldsymbol{G}$, the beamforming vectors $\{\boldsymbol{w}_{\rm t}^\star, \boldsymbol{w}_{\rm r}^\star\}$ that maximize the beamforming gain are given by

$$\boldsymbol{w}_{\rm t}^\star = \boldsymbol{u}_{\max}/\|\boldsymbol{u}_{\max}\|_2, \tag{4a}$$

$$\boldsymbol{w}_{\rm r}^\star = \boldsymbol{v}_{\max}/\|\boldsymbol{v}_{\max}\|_2, \tag{4b}$$

where $\boldsymbol{u}_{\max}$ and $\boldsymbol{v}_{\max}$ are respectively the left and the right singular vectors associated with the largest singular value of the matrix $\boldsymbol{G}$.

However, the channel matrix $\boldsymbol{G}$ is not known initially. The transceivers need to effectively obtain the CSI from a pilot training phase. To this end, this paper considers an adaptive sensing strategy without explicit CSI estimation. In particular, this paper proposes to use a deep active sensing approach to adaptively design the sensing beamformers at both sides during pilot training. Moreover, the subsequent beamformers for data transmission are also learned in an end-to-end fashion. This is all accomplished based on a novel ping-pong pilot protocol without explicit feedback.

B. Ping-Pong Pilot Training Protocol

The ping-pong pilot training protocol is illustrated in Fig. 1, in which the pilot symbols are sent back and forth between agent A (i.e., Tx) and agent B (i.e., Rx) such that each side gathers the required information to design its own beamformer for the data transmission phase.

In the ℓ-th transmission round, agent A first sends a pilot symbol $x_\ell^{\rm A}$ under a power constraint $\mathbb{E}[|x_\ell^{\rm A}|^2] \le P_1$ to agent B, then the received pilot symbol at agent B is given by

$$y_\ell^{\rm B} = (\boldsymbol{w}_{{\rm r},\ell}^{\rm B})^{\sf H}\boldsymbol{G}^{\sf H}\boldsymbol{w}_{{\rm t},\ell}^{\rm A}x_\ell^{\rm A} + n_\ell^{\rm B}, \quad \ell = 0, \cdots, L-1, \tag{5}$$

where the vectors $\boldsymbol{w}_{{\rm t},\ell}^{\rm A} \in \mathbb{C}^{M_{\rm t}}$ and $\boldsymbol{w}_{{\rm r},\ell}^{\rm B} \in \mathbb{C}^{M_{\rm r}}$ are respectively the transmit sensing beamforming vector at agent A and the receive sensing beamforming vector at agent B in the ℓ-th round of pilot transmission, and $n_\ell^{\rm B} \sim \mathcal{CN}(0, \sigma^2)$ is the additive Gaussian noise. The beamforming vectors in the pilot training phase are called sensing vectors to distinguish from the beamforming vectors in the data transmission phase. After receiving the pilot, as shown in Fig. 1, agent B sends back a pilot symbol $x_\ell^{\rm B}$ under a power constraint $\mathbb{E}[|x_\ell^{\rm B}|^2] \le P_2$ to agent A. Similarly, the received pilot at agent A is given by

$$y_\ell^{\rm A} = (\boldsymbol{w}_{{\rm r},\ell}^{\rm A})^{\sf H}\boldsymbol{G}\boldsymbol{w}_{{\rm t},\ell}^{\rm B}x_\ell^{\rm B} + n_\ell^{\rm A}, \quad \ell = 0, \cdots, L-1, \tag{6}$$

where the vectors $\boldsymbol{w}_{\mathrm{r},\ell}^{\mathrm{A}} \in \mathbb{C}^{M_{\mathrm{t}}}$ and $\boldsymbol{w}_{\mathrm{t},\ell}^{\mathrm{B}} \in \mathbb{C}^{M_{\mathrm{r}}}$ are the sensing vectors at agent A and agent B, respectively, and $n_\ell^{\mathrm{A}} \sim \mathcal{CN}(0, \sigma^2)$ is the additive Gaussian noise. Without loss of generality, we can set $x_\ell^{\mathrm{A}} = \sqrt{P_1}$ and $x_\ell^{\mathrm{B}} = \sqrt{P_2}$. After L rounds of pilot transmission, each of the transceivers obtains L measurements of the channel, which can be utilized to design their own beamforming vector for the data transmission phase. The overall pilot training overhead is $2L$ for L rounds of pilot transmission.

所提出的方案不需要feedback

Unlike the conventional schemes in which both the transmit and receive beamformers are designed at one node, the presented protocol designs the beamformers locally, hence it does not require a subsequent feedback procedure.

C. Active Sensing for Two-Sided Beam Alignment

The proposed active sensing approach is an adaptive sensing method because each sensing vector is designed based on the previously received pilot symbols. At the begining of the ℓ-th ping-pong round, as shown in Fig. 1, agent A sends a pilot to agent B. After receiving the observation y_ℓ^{B}, agent B utilizes all the historical observations $\{y_i^{\mathrm{B}}\}_{i=0}^{\ell}$ to design its next transmit sensing beamformer $\boldsymbol{w}_{\mathrm{t},\ell}^{\mathrm{B}}$, as well as the receive sensing beamformer $\boldsymbol{w}_{\mathrm{r},\ell+1}^{\mathrm{B}}$ in the next round, i.e.,

$$\boldsymbol{w}_{\mathrm{t},\ell}^{\mathrm{B}} = f_{\mathrm{t},\ell}^{\mathrm{B}}\left(\{y_i^{\mathrm{B}}\}_{i=0}^{\ell}\right), \ \ell = 0, \cdots, L-1, \tag{7a}$$

$$\boldsymbol{w}_{\mathrm{r},\ell+1}^{\mathrm{B}} = f_{\mathrm{r},\ell}^{\mathrm{B}}\left(\{y_i^{\mathrm{B}}\}_{i=0}^{\ell}\right), \ \ell = 0, \cdots, L-2, \tag{7b}$$

where $f_{\mathrm{t},\ell}^{\mathrm{B}} : \mathbb{C}^{\ell+1} \to \mathbb{C}^{M_{\mathrm{r}}}$ and $f_{\mathrm{r},\ell}^{\mathrm{B}} : \mathbb{C}^{\ell+1} \to \mathbb{C}^{M_{\mathrm{r}}}$ are the corresponding active sensing schemes at agent B. The outputs of the functions $f_{\mathrm{t},\ell}^{\mathrm{B}}$ and $f_{\mathrm{r},\ell}^{\mathrm{B}}$ should also satisfy the unit ℓ_2-norm constraints, i.e., $\|\boldsymbol{w}_{\mathrm{t},\ell}^{\mathrm{B}}\|_2 = \|\boldsymbol{w}_{\mathrm{r},\ell+1}^{\mathrm{B}}\|_2 = 1$.

At the end of the ℓ-th ping-pong round, agent A has made $\ell+1$ observations, i.e., $\{y_i^{\mathrm{A}}\}_{i=0}^{\ell}$, so its next transmit sensing beamforming vector $\boldsymbol{w}_{\mathrm{t},\ell+1}^{\mathrm{A}}$ and next receive sensing beamforming vector $\boldsymbol{w}_{\mathrm{r},\ell+1}^{\mathrm{A}}$ in the next round can be designed as functions of these observations, i.e.,

$$\boldsymbol{w}_{\mathrm{t},\ell+1}^{\mathrm{A}} = f_{\mathrm{t},\ell}^{\mathrm{A}}\left(\{y_i^{\mathrm{A}}\}_{i=0}^{\ell}\right), \ \ell = 0, \cdots, L-2, \tag{8a}$$

$$\boldsymbol{w}_{\mathrm{r},\ell+1}^{\mathrm{A}} = f_{\mathrm{r},\ell}^{\mathrm{A}}\left(\{y_i^{\mathrm{A}}\}_{i=0}^{\ell}\right), \ \ell = 0, \cdots, L-2, \tag{8b}$$

where $f_{\mathrm{t},\ell}^{\mathrm{A}} : \mathbb{C}^{\ell+1} \to \mathbb{C}^{M_{\mathrm{t}}}$ and $f_{\mathrm{r},\ell}^{\mathrm{A}} : \mathbb{C}^{\ell+1} \to \mathbb{C}^{M_{\mathrm{t}}}$ are respectively the transmit and receive active sensing strategies that map the historical observations to the sensing vectors in the next round, while satisfying the unit ℓ_2-norm constraints, $\|\boldsymbol{w}_{\mathrm{t},\ell+1}^{\mathrm{A}}\|_2 = \|\boldsymbol{w}_{\mathrm{r},\ell+1}^{\mathrm{A}}\|_2 = 1$. In the initial stage, since no prior observation is available, we fix the initial beamformers $\boldsymbol{w}_{\mathrm{t},0}^{\mathrm{A}}$, $\boldsymbol{w}_{\mathrm{r},0}^{\mathrm{B}}$, and $\boldsymbol{w}_{\mathrm{r},0}^{\mathrm{A}}$ to be some fixed unit-norm vectors, which can be designed based on channel statistics.

After L rounds of pilot transmission, the two agents design their respective beamformers for data transmission based on all the historical received pilots. In particular, the final beamforming vectors in the data transmission phase are given by

$$\boldsymbol{w}_{\mathrm{t}} = g_{\mathrm{t}}\left(\{y_i^{\mathrm{A}}\}_{i=0}^{L-1}\right), \tag{9a}$$

$$\boldsymbol{w}_{\mathrm{r}} = g_{\mathrm{r}}\left(\{y_i^{\mathrm{B}}\}_{i=0}^{L-1}\right), \tag{9b}$$

where $g_{\mathrm{t}} : \mathbb{C}^{L} \to \mathbb{C}^{M_{\mathrm{t}}}$ and $g_{\mathrm{r}} : \mathbb{C}^{L} \to \mathbb{C}^{M_{\mathrm{r}}}$ are functions that map the received pilots to the final beamforming vectors with unit-norm constraints at agent A (i.e., Tx) and agent B (i.e., Rx), respectively.

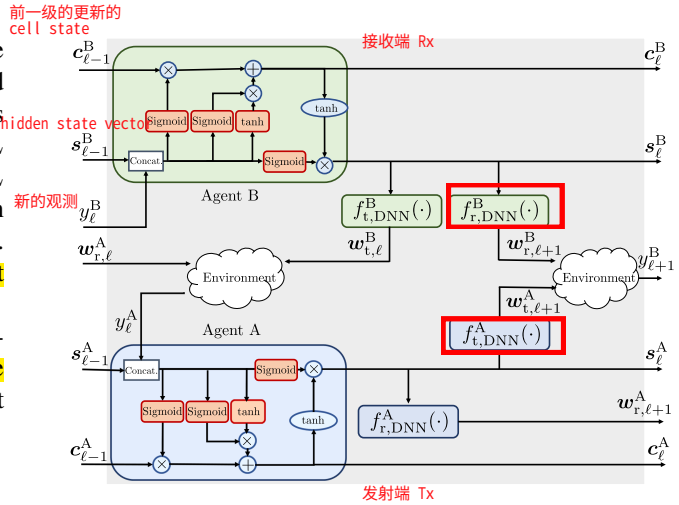

Fig. 2: Proposed active sensing unit for the two-sided beam alignment problem in the ℓ-th ping-pong pilot training round.

This paper aims to find the optimal active sensing strategies together with the mapping functions at the final stage so that the overall beamforming gain for data transmission is maximized. The overall problem can be formulated as

$$\underset{\mathcal{F}}{\text{maximize}} \quad \mathbb{E}\left[|\boldsymbol{w}_{\mathrm{r}}^{\mathsf{H}} \boldsymbol{G}^{\mathsf{H}} \boldsymbol{w}_{\mathrm{t}}|^2\right] \tag{10a}$$

$$\text{subject to} \quad (7), (8), \text{ and } (9), \tag{10b}$$

where the optimization variables are a set of functions

$$\mathcal{F} = \Big\{\{f_{\mathrm{t},\ell}^{\mathrm{A}}(\cdot)\}_{\ell=0}^{L-2}, \{f_{\mathrm{r},\ell}^{\mathrm{A}}(\cdot)\}_{\ell=0}^{L-2}, \{f_{\mathrm{t},\ell}^{\mathrm{B}}(\cdot)\}_{\ell=0}^{L-1}, \{f_{\mathrm{r},\ell}^{\mathrm{B}}(\cdot)\}_{\ell=0}^{L-2}, g_{\mathrm{t}}(\cdot), g_{\mathrm{r}}(\cdot)\Big\}, \tag{11}$$

and the expectation is taken over all the stochastic parameters in the system, i.e., the channels and the noise.

Solving the problem (10) is computationally challenging because the optimization variables are high-dimensional functions. Moreover, the input dimensions of the functions in $\mathcal{F}$ increases with the number of rounds, so the complexity also scales accordingly. The main idea of this paper is that an RNN can be used to solve this optimization problem efficiently.

III. Proposed Deep Learning Framework

In this section, we propose a deep learning framework to parameterize the mapping in $\mathcal{F}$ and to solve the optimization problem (10). In particular, the sequential nature of the active sensing problem motivates the use of RNN. Further, we propose to use two LSTM cells [15], one on each side, to automatically summarize historical observations into fixed-dimensional state vectors, which are used for designing the subsequent sensing vectors. This is motivated by the fact that LSTM can capture the correlations over long sequences. It has the ability to summarize sufficient information from historical observations, thereby preventing the dimension of the input from growing with the number of pilot transmission rounds.

Fig. 2 shows the architecture of the proposed active sensing unit in the ℓ-th pilot transmission round. In particular, the

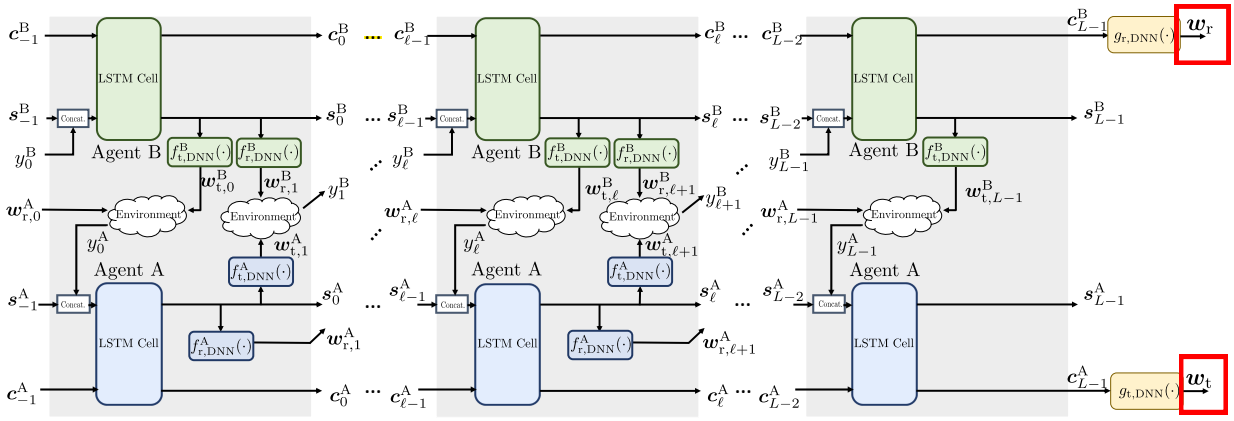

Fig. 3: Proposed overall active sensing framework for the two-sided beam alignment problem with L ping-pong pilot trainining rounds.

proposed active sensing unit consists of two LSTMs deployed at agent A and agent B, respectively.

At the Rx side, given the new observation y_ℓ^{B} at agent B, its LSTM cell outputs the updated cell state vector $\boldsymbol{c}_\ell^{\mathrm{B}}$ and hidden state vector $\boldsymbol{s}_\ell^{\mathrm{B}}$ according to the following equations [15]:

$$\boldsymbol{f}_\ell^{\mathrm{B}} = \mathrm{sigmoid}(\boldsymbol{W}_{\mathrm{f}}^{\mathrm{B}}\boldsymbol{y}_\ell^{\mathrm{B}} + \boldsymbol{U}_{\mathrm{f}}^{\mathrm{B}}\boldsymbol{s}_{\ell-1}^{\mathrm{B}} + \boldsymbol{b}_{\mathrm{f}}^{\mathrm{B}}), \tag{12a}$$
$$\boldsymbol{i}_\ell^{\mathrm{B}} = \mathrm{sigmoid}(\boldsymbol{W}_{\mathrm{i}}^{\mathrm{B}}\boldsymbol{y}_\ell^{\mathrm{B}} + \boldsymbol{U}_{\mathrm{i}}^{\mathrm{B}}\boldsymbol{s}_{\ell-1}^{\mathrm{B}} + \boldsymbol{b}_{\mathrm{i}}^{\mathrm{B}}), \tag{12b}$$
$$\boldsymbol{o}_\ell^{\mathrm{B}} = \mathrm{sigmoid}(\boldsymbol{W}_{\mathrm{o}}^{\mathrm{B}}\boldsymbol{y}_\ell^{\mathrm{B}} + \boldsymbol{U}_{\mathrm{o}}^{\mathrm{B}}\boldsymbol{s}_{\ell-1}^{\mathrm{B}} + \boldsymbol{b}_{\mathrm{o}}^{\mathrm{B}}), \tag{12c}$$
$$\boldsymbol{c}_\ell^{\mathrm{B}} = \boldsymbol{f}_\ell^{\mathrm{B}} \circ \boldsymbol{c}_{\ell-1}^{\mathrm{B}} + \boldsymbol{i}_\ell^{\mathrm{B}} \circ \tanh(\boldsymbol{W}_{\mathrm{c}}^{\mathrm{B}}\boldsymbol{y}_\ell^{\mathrm{B}} + \boldsymbol{U}_{\mathrm{c}}^{\mathrm{B}}\boldsymbol{s}_{\ell-1}^{\mathrm{B}} + \boldsymbol{b}_{\mathrm{c}}^{\mathrm{B}}), \tag{12d}$$
$$\boldsymbol{s}_\ell^{\mathrm{B}} = \boldsymbol{o}_\ell^{\mathrm{B}} \circ \tanh(\boldsymbol{c}_\ell^{\mathrm{B}}), \tag{12e}$$

where $\boldsymbol{y}_\ell^{\mathrm{B}} = [\Re(y_\ell^{\mathrm{B}}), \Im(y_\ell^{\mathrm{B}})]^\top$ is the concatenation of real and imaginary parts of y_ℓ^{B}, $\{\boldsymbol{W}_{\mathrm{f}}^{\mathrm{B}}, \boldsymbol{W}_{\mathrm{i}}^{\mathrm{B}}, \boldsymbol{W}_{\mathrm{o}}^{\mathrm{B}}, \boldsymbol{W}_{\mathrm{c}}^{\mathrm{B}}, \boldsymbol{U}_{\mathrm{f}}^{\mathrm{B}}, \boldsymbol{U}_{\mathrm{i}}^{\mathrm{B}}, \boldsymbol{U}_{\mathrm{o}}^{\mathrm{B}}, \boldsymbol{U}_{\mathrm{c}}^{\mathrm{B}}\}$ and $\{\boldsymbol{b}_{\mathrm{f}}^{\mathrm{B}}, \boldsymbol{b}_{\mathrm{i}}^{\mathrm{B}}, \boldsymbol{b}_{\mathrm{o}}^{\mathrm{B}}, \boldsymbol{b}_{\mathrm{c}}^{\mathrm{B}}\}$ are respectively the trainable weights and biases in the LSTM network. Moreover, the forget gate's activation vector $\boldsymbol{f}_\ell^{\mathrm{B}}$, input/update gate's activation vector $\boldsymbol{i}_\ell^{\mathrm{B}}$, and output gate's activation vector $\boldsymbol{o}_\ell^{\mathrm{B}}$ are intermediate vectors generated within the LSTM unit to update the cell state vector $\boldsymbol{c}_\ell^{\mathrm{B}}$ and the hidden state vector $\boldsymbol{s}_\ell^{\mathrm{B}}$.

Similarly, the Tx utilizes another LSTM with the same architecture but with different trainable parameters $\{\boldsymbol{W}_{\mathrm{f}}^{\mathrm{A}}, \boldsymbol{W}_{\mathrm{i}}^{\mathrm{A}}, \boldsymbol{W}_{\mathrm{o}}^{\mathrm{A}}, \boldsymbol{W}_{\mathrm{c}}^{\mathrm{A}}, \boldsymbol{U}_{\mathrm{f}}^{\mathrm{A}}, \boldsymbol{U}_{\mathrm{i}}^{\mathrm{A}}, \boldsymbol{U}_{\mathrm{o}}^{\mathrm{A}}, \boldsymbol{U}_{\mathrm{c}}^{\mathrm{A}}\}$ and $\{\boldsymbol{b}_{\mathrm{f}}^{\mathrm{A}}, \boldsymbol{b}_{\mathrm{i}}^{\mathrm{A}}, \boldsymbol{b}_{\mathrm{o}}^{\mathrm{A}}, \boldsymbol{b}_{\mathrm{c}}^{\mathrm{A}}\}$. Given the new observation y_ℓ^{A} at the Tx side, the LSTM updates its cell state vector $\boldsymbol{c}_\ell^{\mathrm{A}}$ and hidden state vector $\boldsymbol{s}_\ell^{\mathrm{A}}$ according to the following equations:

$$\boldsymbol{f}_\ell^{\mathrm{A}} = \mathrm{sigmoid}(\boldsymbol{W}_{\mathrm{f}}^{\mathrm{A}}\boldsymbol{y}_\ell^{\mathrm{A}} + \boldsymbol{U}_{\mathrm{f}}^{\mathrm{A}}\boldsymbol{s}_{\ell-1}^{\mathrm{A}} + \boldsymbol{b}_{\mathrm{f}}^{\mathrm{A}}), \tag{13a}$$
$$\boldsymbol{i}_\ell^{\mathrm{A}} = \mathrm{sigmoid}(\boldsymbol{W}_{\mathrm{i}}^{\mathrm{A}}\boldsymbol{y}_\ell^{\mathrm{A}} + \boldsymbol{U}_{\mathrm{i}}^{\mathrm{A}}\boldsymbol{s}_{\ell-1}^{\mathrm{A}} + \boldsymbol{b}_{\mathrm{i}}^{\mathrm{A}}), \tag{13b}$$
$$\boldsymbol{o}_\ell^{\mathrm{A}} = \mathrm{sigmoid}(\boldsymbol{W}_{\mathrm{o}}^{\mathrm{A}}\boldsymbol{y}_\ell^{\mathrm{A}} + \boldsymbol{U}_{\mathrm{o}}^{\mathrm{A}}\boldsymbol{s}_{\ell-1}^{\mathrm{A}} + \boldsymbol{b}_{\mathrm{o}}^{\mathrm{A}}), \tag{13c}$$
$$\boldsymbol{c}_\ell^{\mathrm{A}} = \boldsymbol{f}_\ell^{\mathrm{A}} \circ \boldsymbol{c}_{\ell-1}^{\mathrm{A}} + \boldsymbol{i}_\ell^{\mathrm{A}} \circ \tanh(\boldsymbol{W}_{\mathrm{c}}^{\mathrm{A}}\boldsymbol{y}_\ell^{\mathrm{A}} + \boldsymbol{U}_{\mathrm{c}}^{\mathrm{A}}\boldsymbol{s}_{\ell-1}^{\mathrm{A}} + \boldsymbol{b}_{\mathrm{c}}^{\mathrm{A}}), \tag{13d}$$
$$\boldsymbol{s}_t^{\mathrm{A}} = \boldsymbol{o}_\ell^{\mathrm{A}} \circ \tanh(\boldsymbol{c}_\ell^{\mathrm{A}}), \tag{13e}$$

where $\boldsymbol{y}_\ell^{\mathrm{A}} = [\Re(y_\ell^{\mathrm{A}}), \Im(y_\ell^{\mathrm{A}})]^\top$ is the real representation of the received pilot symbols at the Tx. To initialize the cell state and hidden state vectors, we set $\boldsymbol{c}_{-1}^{\mathrm{A}} = \boldsymbol{c}_{-1}^{\mathrm{B}} = \boldsymbol{0}$ and $\boldsymbol{s}_{-1}^{\mathrm{A}} = \boldsymbol{s}_{-1}^{\mathrm{B}} = \boldsymbol{0}$ following the convention in LSTM networks.

The idea is to train the LSTM to capture the useful information in the sequence of historical observations into its cell state. We then use the hidden state vectors $\boldsymbol{s}_\ell^{\mathrm{A}}$ and $\boldsymbol{s}_\ell^{\mathrm{B}}$ to design the corresponding sensing vectors. This is achieved by using fully connected DNNs. As shown in Fig. 2, at agent B, the hidden state vector $\boldsymbol{s}_\ell^{\mathrm{B}}$ is taken as input to two different fully connected DNNs to output the next transmit sensing beamforming vector $\boldsymbol{w}_{\mathrm{t},\ell}^{\mathrm{B}}$ and receive sensing beamforming vector $\boldsymbol{w}_{\mathrm{r},\ell+1}^{\mathrm{B}}$ as follows:

$$\boldsymbol{w}_{\mathrm{t},\ell}^{\mathrm{B}} = f_{\mathrm{t,DNN}}^{\mathrm{B}}(\boldsymbol{s}_\ell^{\mathrm{B}}), \tag{14a}$$
$$\boldsymbol{w}_{\mathrm{r},\ell+1}^{\mathrm{B}} = f_{\mathrm{r,DNN}}^{\mathrm{B}}(\boldsymbol{s}_\ell^{\mathrm{B}}). \tag{14b}$$

Analogously, the next transmit and receive sensing beamformers at agent A, i.e., $\boldsymbol{w}_{\mathrm{t},\ell+1}^{\mathrm{A}}$ and $\boldsymbol{w}_{\mathrm{r},\ell+1}^{\mathrm{A}}$, are designed by two different fully connected DNNs with the hidden state vector $\boldsymbol{s}_\ell^{\mathrm{A}}$ as input i.e.,

$$\boldsymbol{w}_{\mathrm{t},\ell+1}^{\mathrm{A}} = f_{\mathrm{t,DNN}}^{\mathrm{A}}(\boldsymbol{s}_\ell^{\mathrm{A}}), \tag{15a}$$
$$\boldsymbol{w}_{\mathrm{r},\ell+1}^{\mathrm{A}} = f_{\mathrm{r,DNN}}^{\mathrm{A}}(\boldsymbol{s}_\ell^{\mathrm{A}}). \tag{15b}$$

Fig. 2 illustrates the proposed active sensing unit in the ℓ-th pilot transmission round. To train the neural networks in the active sensing unit, we concatenate L active sensing units together to form a very deep neural network, corresponding to the L rounds of pilot transmission in Fig. 1. The overall deep active sensing architecture is shown in Fig. 3. The neural network parameters across different pilot transmission rounds can be tied together to reduce the training complexity. After L rounds of pilot transmission, the state vector $\{\boldsymbol{c}_{L-1}^{\mathrm{A}}, \boldsymbol{c}_{L-1}^{\mathrm{B}}\}$ of the LSTMs at both sides are respectively mapped to the final beamforming vectors $\{\boldsymbol{w}_{\mathrm{t}}, \boldsymbol{w}_{\mathrm{r}}\}$ for the data transmission phase. This is done by employing another two DNNs in the final stage as follows:

$$\boldsymbol{w}_{\mathrm{t}} = g_{\mathrm{t,DNN}}(\boldsymbol{c}_{L-1}^{\mathrm{A}}), \tag{16a}$$
$$\boldsymbol{w}_{\mathrm{r}} = g_{\mathrm{r,DNN}}(\boldsymbol{c}_{L-1}^{\mathrm{B}}). \tag{16b}$$

The overall neural network can be trained end-to-end in order to maximize the utility function $\mathbb{E}\left[|\boldsymbol{w}_{\mathrm{r}}^{\mathsf{H}}\boldsymbol{G}^{\mathsf{H}}\boldsymbol{w}_{\mathrm{t}}|^2\right]$ by employing stochastic gradient descent (SGD). In this way, the

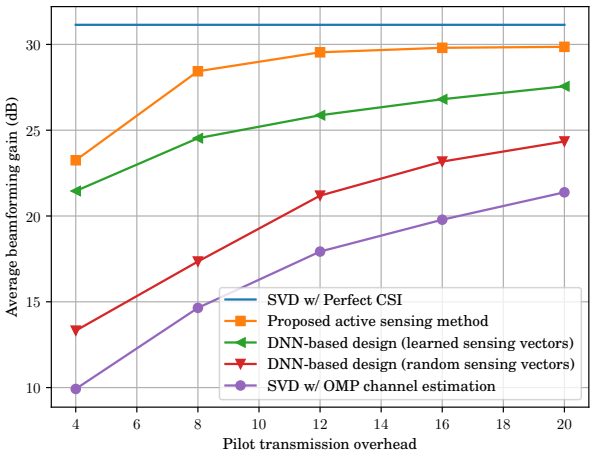

Fig. 4: Average beamforming gain vs. pilot training overhead.

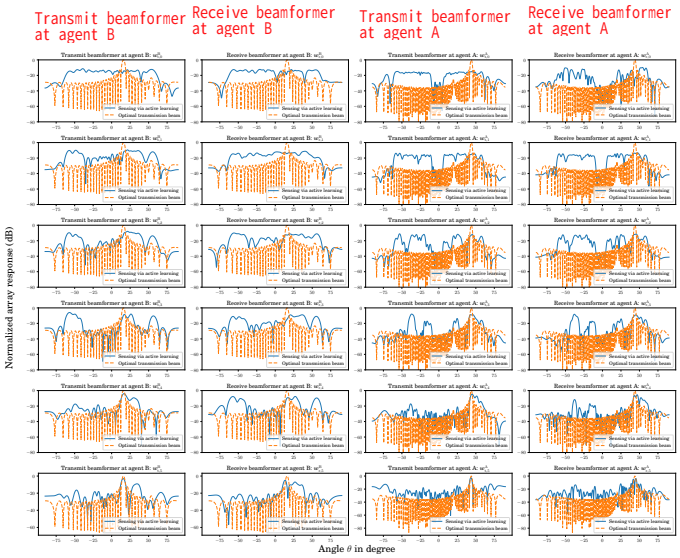

Fig. 5: Learned sensing beamforming patterns.

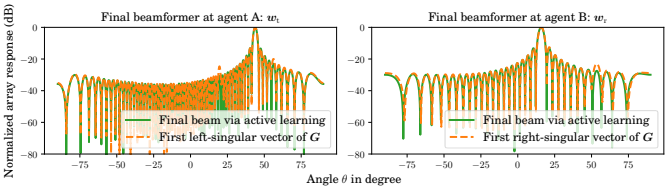

Fig. 6: Learned data transmission beamforming patterns.

active sensing strategies together with the final DNNs at both sides are jointly optimized. Once trained and deployed, no feedback is needed between the Tx and the Rx.

IV. Performance Evaluation

In simulations, we consider a system with $M_{\rm t} = 64$ antennas at the Tx and $M_{\rm r} = 32$ antennas at the Rx. The number of paths between the Tx and Rx is set to $L_p = 3$. For each channel realization, the AoAs/AoDs are uniformly generated from $[-60°, 60°]$, and the complex fading coefficients are randomly taken from the distribution $\mathcal{CN}(0, 1)$. In the pilot training phase, the raw SNRs in both directions, i.e., P_1/σ^2 and P_2/σ^2, are set to be 0dB.

For the proposed deep active sensing unit, the dimensions of hidden states and cell states are both set to be 512 for the Tx side and both set to be 256 for the Rx side. The DNNs in (15) are of size $[512, 512, 2M_{\rm t}]$, and the size of DNN $g_{\rm t,DNN}(\cdot)$ is $[1024, 1024, 2M_{\rm t}]$; the DNNs in (14) are of size $[512, 512, 2M_{\rm r}]$, and the size of DNN $g_{\rm r,DNN}(\cdot)$ is $[1024, 1024, 2M_{\rm r}]$. We adopt the rectified linear unit (ReLU) activation function in all the dense layers except the last layer that outputs the beamforming vector. To meet the unit ℓ_2-norm constraint on the beamforming vectors, the activation function for the last dense layer is chosen to be $\varrho(\cdot) = \cdot/\|\cdot\|_2$. A batch normalization layer is added between two dense layers in the fully connected neural networks to accelerate the training process [16]. The initial sensing vectors $\boldsymbol{w}_{\rm t,0}^{\rm A}$, $\boldsymbol{w}_{\rm r,0}^{\rm B}$, and $\boldsymbol{w}_{\rm r,0}^{\rm A}$ as shown in Fig. 1 are set as trainable parameters in the implementation, so that they are learned from the distribution of the training data. The parameters of the dense layers in the active sensing unit are tied together across different sensing stages to reduce the training complexity. The overall deep learning framework is implemented on Tensorflow [17] and trained by Adam optimizer [18] with a learning rate progressively decreasing from 10^{-4} to 10^{-5}. We compare the proposed method with channel estimatation based approach [7] using orthogonal matching pursuit (OMP) algorithm [19] and DNN-based approach [20] where the received pilots with fixed sensing vectors are directly mapped to data transmission beamformers by a DNN.

In Fig. 4, it can be seen that the data-driven approaches, which bypass channel estimation, significantly outperform the channel estimation based approach. The DNN-based approach with learned sensing vectors can achieve much better performance as compared to the DNN-based method with random sensing vectors. This implies that designing the sensing vectors is important for improving the quality of the received pilots. Further, the proposed active sensing approach, which utilizes pilot observations that are functions of instantaneous CSI, achieves better performance than the DNN-based approach that learns the sensing vectors based on channel statistics. Specifically, at raw SNR of 0dB, the proposed approach using 8 pilot transmission overhead can outperform all the other benchmarks with 20 pilot transmissions. This indicates that the proposed deep active sensing framework can efficiently design the sensing vectors from the observations, thereby reducing the pilot training overhead. Moreover, the proposed method does not need feedback and coordination between the Tx and the Rx in the pilot training stage.

V. Interpretation of Active Sensing Solution

We have shown that the proposed active sensing method can significantly outperform the state-of-the-art benchmarks for the two-sided beam alignment problem. It is important to interpret the solutions learned via the LSTM based neural networks and to understand where the gains come from. To this end, we examine the beamforming patterns of the designed

sensing and beamforming vectors using the normalized array response $f_{\text{beam}}(\boldsymbol{w},\theta) = |\boldsymbol{w}^{\text{H}}\boldsymbol{a}(\theta)|^2,\ \forall\theta \in [-\pi/2,\pi/2]$, where $\boldsymbol{w}\in\mathbb{C}^M$ is the designed sensing/beamforming vector and $\boldsymbol{a}(\theta)=[1,\cdots,e^{j\pi(M-1)\sin(\theta)}]^\top/\sqrt{M}$ is the normalized steering vector.

In Fig. 5, we plot the beamforming pattern of the sensing vectors $\boldsymbol{w}^{\text{B}}_{\text{r},\ell}$, $\boldsymbol{w}^{\text{B}}_{\text{t},\ell}$, $\boldsymbol{w}^{\text{A}}_{\text{t},\ell}$ and $\boldsymbol{w}^{\text{A}}_{\text{r},\ell}$ designed via the proposed active sensing framework for $\ell=0,\cdots,5$ for a randomly generated channel realization. The active sensing neural network is trained for $L=6$ ping-pong transmission rounds. As can be seen from Fig. 5, the sensing vectors designed by the active sensing unit have relatively uniform array response in the first two rounds, i.e., $\ell=0,1$. As the number of observations increases, the active sensing units at both sides gradually learn to narrow down the search directions and to focus more energy in the optimal direction, but they also try to explore other directions. This can be interpreted as a behavior of systematic exploration of channel landscape, learned by the proposed active sensing method for finding the optimal beamformers for the eventual data transmission.

In Fig. 6, we plot the beamforming pattern of the data transmission beamforming vectors $\{\boldsymbol{w}_{\text{t}},\boldsymbol{w}_{\text{r}}\}$, which are learned via the DNNs after the final sensing stage as in (16) for the channel realization used in Fig. 5. It can be seen that the final data transmission beamformers perfectly match the optimal beamforming vectors computed from (4) assuming perfect CSI, (i.e., the singular vectors corresponding on the largest singular value). This means that the proposed active sensing framework indeed learns an intelligent strategy to design the sensing vectors to collect the received pilots so that the optimal final data transmission beamformers can be found within a few pilot transmission rounds.

We observe through simulations over many random channel realizations that the proposed active sensing strategy learns to match the singular vector corresponding to the largest eigenvalue a majority of the time, but there are also cases in which the learned data transmission beamformers happen to match the second or third singular vectors. This phenomenon accounts the gap between the proposed active sensing method and the perfect-CSI benchmark seen in Fig. 4. This gap is approximately 1.58dB when the number of pilot transmission overhead is 12 (or $L=6$).

VI. Conclusions

This paper proposes an active sensing framework to sequentially design the sensing vectors from successive observations for the two-sided beam alignment problem in mmWave systems. The proposed approach is based on a novel ping-pong pilot training scheme, which eliminates the need for feedback between the Tx and the Rx at the sensing stage, and is data driven, so that it does not rely on mathematical models of the channel. Active sensing is a challenging problem, because of the sequential nature of the sensing operation, which allows successive exploration of the channel landscape, but also requires the learning agent to succinctly summarize the observations so far, which is a nontrivial task. To this end, this paper proposes to use an active sensing strategy based on LSTM neural networks to account for the sequential dependency of the sensing task. Simulation results verify the superior performance of the proposed method as compared to the previous state-of-the-art methods and that the learned beamforming patterns are meaningful and interpretable.

References

[1] T. S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. N. Wong, J. K. Schulz, M. Samimi, and F. Gutierrez, "Millimeter wave mobile communications for 5G cellular: It will work!" *IEEE Access*, vol. 1, pp. 335–349, May 2013.

[2] C.-X. Wang, F. Haider, X. Gao, X.-H. You, Y. Yang, D. Yuan, H. M. Aggoune, H. Haas, S. Fletcher, and E. Hepsaydir, "Cellular architecture and key technologies for 5G wireless communication networks," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 122–130, Feb. 2014.

[3] F. Sohrabi and W. Yu, "Hybrid digital and analog beamforming design for large-scale antenna arrays," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 3, pp. 501–513, Apr. 2016.

[4] S.-E. Chiu, N. Ronquillo, and T. Javidi, "Active learning and CSI acquisition for mmWave initial alignment," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 11, pp. 2474–2489, Nov. 2019.

[5] F. Sohrabi, Z. Chen, and W. Yu, "Deep active learning approach to adaptive beamforming for mmWave initial alignment," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2347–2360, Aug. 2021.

[6] F. Sohrabi, T. Jiang, W. Cui, and W. Yu, "Active sensing for communications by learning," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 6, pp. 1780–1794, 2022.

[7] A. Alkhateeb, O. El Ayach, G. Leus, and R. W. Heath, "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 831–846, Oct. 2014.

[8] X. Song, S. Haghighatshoar, and G. Caire, "A scalable and statistically robust beam alignment technique for millimeter-wave systems," *IEEE Trans. Wireless Commun.*, vol. 17, no. 7, pp. 4792–4805, July 2018.

[9] ——, "Efficient beam alignment for millimeter wave single-carrier systems with hybrid MIMO transceivers," *IEEE Trans. Wireless Commun.*, vol. 18, no. 3, pp. 1518–1533, Mar. 2019.

[10] I. Aykin and M. Krunz, "Efficient beam sweeping algorithms and initial access protocols for millimeter-wave networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2504–2514, Apr. 2020.

[11] C. N. Manchón, E. de Carvalho, and J. B. Andersen, "Ping-pong beam training with hybrid digital-analog antenna arrays," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Paris, France, May 2017.

[12] N. Akdim, C. N. Manchón, M. Benjillali, and E. de Carvalho, "Ping pong beam training for multi stream MIMO communications with hybrid antenna arrays," in *Proc. IEEE Global Commun. Workshops (GLOBECOM Workshops)*, Abu Dhabi, UAE, Dec. 2018.

[13] F. Pedraza and G. Caire, "Adaptive two-sided beam alignment in mmWave via posterior matching," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Riva del Garda, Italy, Apr. 2021.

[14] N. Akdim, C. N. Manchón, M. Benjillali, and P. Duhamel, "Variational hierarchical posterior matching for mmWave wireless channels online learning," in *Proc. IEEE Int. Workshop Signal Process. Advances Wireless Commun. (SPAWC)*, Atlanta, GA, USA, May 2020.

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[16] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Lille, France, July 2015.

[17] M. Abadi *et al.*, "TensorFlow: A system for large-scale machine learning," in *USENIX Symp. Operating Syst. Des. Implementation (OSDI)*, Savannah, USA, Nov. 2016, pp. 265–283.

[18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations (ICLR)*, San Diego, CA, USA, May 2015.

[19] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[20] T. Jiang, H. V. Cheng, and W. Yu, "Learning to reflect and to beamform for intelligent reflecting surface with implicit channel estimation," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 7, pp. 1931–1945, July 2021.