

Intelligent Analog Beam Selection and Beamspace Channel Tracking in THz Massive MIMO with Lens Antenna Array

Hosein Zarini, Mohammad Robot Mili,
Mehdi Rasti, *Senior Member, IEEE*, Sergey Andreev, *Senior Member, IEEE*,
P. H. J. Nardelli, *Senior Member, IEEE*, and Mehdi Bennis, *Fellow, IEEE*

Abstract—Beamspace multiple-input-multiple-output (MIMO) as a green technology can efficiently substitute for the conventional massive MIMO, provided that the beamspace channel is acquired precisely. The prior efforts in this area of study, especially the learning-driven ones, however, indicate remarkable performance losses owing to a lack of generalization. In this paper, we propose a modified non-linear auto-regressive exogenous (NARX) model for tracking and predicting the beamspace channel over the sequences of time. Benefiting from bounded generalization error, fast convergence, limited prediction variance, and negligible performance loss, the proposed scheme achieves up to 15% spectral efficiency (SE) gain over its counterparts. We further improve this performance by means of an ensemble learning technique for simultaneously training multiple NARX modules in parallel, thus leading to a 23% SE gain. Relying on the predicted beamspace channel, we propose a beamspace analog beam selection technique through fine-tuning the architecture of a pre-trained off-the-shelf GoogleNet, which brings up to 21% SE gain over similar baselines. With the aid of an ensemble learning technique, it is further indicated numerically that up to 34% SE improvement can be achieved, as compared to the counterparts.

Index Terms—Terahertz (THz) beamspace, non-linear auto-regressive exogenous (NARX) model, Bayesian optimizer, ensemble learning, GoogleNet, transfer learning.

I. INTRODUCTION

High-frequency spectrum with large free bandwidth is promising to compensate for the spectrum scarcity in future wireless communication systems. However, the environmental attenuation severely influences high-frequency communications such as those in the terahertz (THz) band in contrast to sub-6 GHz bands. As a consequence, only short-range links are feasible in the THz band. To facilitate high-frequency transmissions, massive multiple-input-multiple-output (MIMO) technology with large-scale antenna arrays is proposed [2].

This paper was presented in part at the IEEE 95th Vehicular Technology Conference (VTC2022-Spring) Workshops [1]. H. Zarini is with the Department of Computer Engineering at the Sharif University of Technology, Tehran, Iran (e-mail: hosein.zarini68@sharif.edu). M. Robot Mili is with the Pasargad Institute for Advanced Innovative Solutions (PIAIS), Tehran, Iran (email: mohammad.robotmili@piais.ir). M. Rasti, P.H.J. Nardelli and M. Bennis are with the Center for Wireless Communications at the University of Oulu, Oulu, Finland (email: mehdi.rasti@oulu.fi, pedro.nardelli@lut.fi and mehdi.bennis@oulu.fi). P.H.J. Nardelli is also with the Department of Electrical Engineering, School of Energy Systems, Lappeenranta-Lahti University of Technology (LUT), Lappeenranta, Finland. S. Andreev is with the faculty of information technology and communication sciences at Tampere university, Tampere, Finland (email: sergey.andreev@tuni.fi).

Generally, a conventional massive MIMO system is to be equipped with a large-scale antenna array to compensate for the significant attenuation in high-frequency bands. This system under a conventional implementation is, however, architecturally ill-suited for high-frequency bands. Conventionally, each antenna element in massive MIMO systems is associated with a unique radio-frequency (RF) chain, which is known as the dominant module for hardware cost, energy consumption, and system complexity [3]. Accordingly, the system needs an equivalent massive number of RF chains, which makes it inefficient and infeasible.

For a more efficient architecture, hybrid analog-digital beamspace MIMO [4] relies on a lens antenna array, where the focal surface of an energy-focusing electromagnetic lens is covered by a large-scale antenna array. This architecture is shown to work beneficially with a far fewer number of required RF chains. By employing the lens antenna array in a beamspace architecture, the system cost, energy, and complexity are reduced remarkably, as compared to the conventional massive MIMO architecture. At the cost of a negligible performance loss, the overall performance can be maintained at a satisfactory level. The lens antenna array in essence performs a discrete Fourier transformation (DFT) operation and concentrates the scattered signals of the divergent paths (beams) upon a limited number of antennas. Therefore, the angular space is thoroughly covered by the lens antenna array and the spatial domain is transformed into the beamspace domain via the lens antenna array¹ [5]. In the high-frequency spectrum, including the THz band, the transmission power should be highly concentrated on a limited number of dominant beams. Further, the number of effective line-of-sight (LoS) beams is far fewer and the THz beamspace channel is consequently quite sparse. Benefitting from this, one can conclude that the number of required RF chains is limited, hence making the lens-aided hybrid architecture a reasonable system design in terms of energy, cost, and complexity [6].

This architecture, despite its main advantage (i.e., sparse beamspace channel), still encounters important issues in system design. Primarily, the performance is highly affected

¹Note that from the functionality point of view, the lens antenna array works similar to the DFT concept on transforming the domains. The DFT transforms the signal from the time domain to the frequency domain, whereas the lens antenna array transforms the signal from the space domain to the beamspace domain.

in cases where the channel state information (CSI) is only partially available. Additionally, the optimal design of analog beam selection is computationally challenging [7]. Regarding these concerns, a category of solutions has been proposed in the literature aiming at performance optimization. Due to significant computational burden, earlier optimization-based schemes for beamspace channel estimation and analog beam selection cannot practically satisfy the real-time processing requirements of current applications².

In recent years, low-complexity real-time deep learning approaches are substituting in wireless network research the more complex optimization-based methods. For instance, the authors of [8] and [9] applied deep learning techniques to reduce the computational burden and improve the accuracy of the approximate message passing (AMP) [10] scheme for channel estimation. Accordingly, for vehicular communications, the authors of [11] investigated a long short-term memory (LSTM) scheme for efficient channel estimation. In [12], the authors analyzed the performance of the proposed deep learning techniques in predicting the beam angles on transceivers. The results of [12] reported non-negligible root mean square error (RMSE) values for the existing solutions.

On the other hand, learning-based approaches, such as support vector machine (SVM), k -nearest neighbors (k -NN), and multi-layer perceptron (MLP), were recently attempted for analog beam selection as a classification task [13]. Compared to the fully-digital systems, however, the accuracy loss in analog beam selection significantly degrades the performance of the beamspace MIMO system as a hybrid analog-digital architecture. According to the statistics in [12] (trained on environmental samples, e.g., LoS and non-LoS (NLoS) beams), two well-known classifiers, i.e., linear SVM [13] and decision tree [14] are only 33% and 55% accurate, respectively. This, in turn, leads to a considerable performance loss for the beamspace architecture, as compared to the fully-digital counterpart.

The main contribution of this paper is in the investigation of fine-tuned deep learning along with ensemble learning to perform beamspace channel tracking and analog beam selection more accurately. In detail, the primary claims of this work are outlined as follows.

- We propose a THz beamspace channel estimation/tracking mechanism using time-series prediction. To this end, a non-linear auto-regressive exogenous (NARX) model is trained by a Levenberg-Marquardt policy, and its learning rate hyperparameter is regularized via a Bayesian optimizer for faster convergence, as compared to the considered baseline in [15]. Further, we present a theoretical analysis of the generalization error. The proposed method exhibits remarkably lower RMSE and variance against its counterparts [8] and [9], especially in the validation and testing phases. Compared to the LSTM baseline [11] in terms of spectral efficiency (SE), the proposed

strategy demonstrates up to 15% improvement when the signal-to-noise ratio (SNR) is 15dB.

- Relying on the predicted beamspace channel, we present an analog beam selection strategy by fine-tuning the off-the-shelf pre-trained GoogleNet classifier based on transfer learning technique to learn the analog beam selection as a classification of the dominant beams into the RF chains. Further, we replace the conventional rectified linear unit (ReLU) activation function within the GoogleNet layers with the Swish. It is shown numerically that the Swish-driven and ReLU-driven GoogleNet schemes, respectively, on average achieve 86% and 83% accuracy for analog beam selection. Moreover, compared to MLP [13], leveraging ReLU-driven and Swish-driven GoogleNet for analog beam selection leads to 17% and 21% improvement in the achievable SE, respectively, when SNR = 30dB.
- We further improve the performance of the proposed strategies for beamspace channel tracking and analog beam selection by leveraging an ensemble learning technique that aggregates multiple predictions for higher precision and lower variance in predictions. The strong ensemble learner enhances the achievable SE of [11] by up to 23% for SNR = 15dB, while decreasing the standard deviation and the mean absolute deviation of an individual trained NARX module by up to 47% and 52%, respectively. From the analog beam selection perspective, a strong ensemble classifier is trained that accommodates multiple fine-tuned Swish-driven GoogleNet classifiers, which leads to 94% analog beam selection accuracy yielding, in turn, 34% achievable SE improvement at SNR = 30dB over the MLP scheme [13].

The remainder of this paper is organized as follows. Section II describes our system setup. Section III includes the beamspace channel tracking problem statement and the corresponding proposed solution. In Section IV, the analog beam selection problem and the proposed solution are elaborated. The ensemble learning technique and the complexity analysis are discussed in Sections V and VI, respectively. Finally, simulation results and conclusions are presented in Sections VII and VIII, correspondingly.

Notations: We use bold-face upper and lower case to indicate matrices and vectors, respectively. $(\cdot)^T$, $(\cdot)^H$, and $\|\cdot\|_2$ denote the transpose, conjugate transpose, and the second norm of a matrix, respectively. $|\cdot|$ represents the absolute operator, while \mathbf{I}_N denotes the identity matrix of size $N \times N$.

II. SYSTEM SETUP

A. Hybrid Analog-Digital Architecture

A hybrid analog-digital THz beamspace MIMO system in the downlink communication mode is assumed. A transmitter employing N_t transmit antennas and N_t^{RF} transmit RF chains ($N_t^{\text{RF}} \leq N_t$) serves a receiver equipped with N_r receive antennas as well as N_r^{RF} receive RF chains ($N_r^{\text{RF}} \leq N_r$). The number of simultaneously communicated data streams is denoted by N_s . We leverage a hybrid analog-digital beamspace

²Although previous optimization methods have high complexity to this end, they are attractive from two perspectives. First, these methods provide a benchmarking case for further comparison. Second, they offer training samples widely used for supervised learning methods (such as the model of this paper).

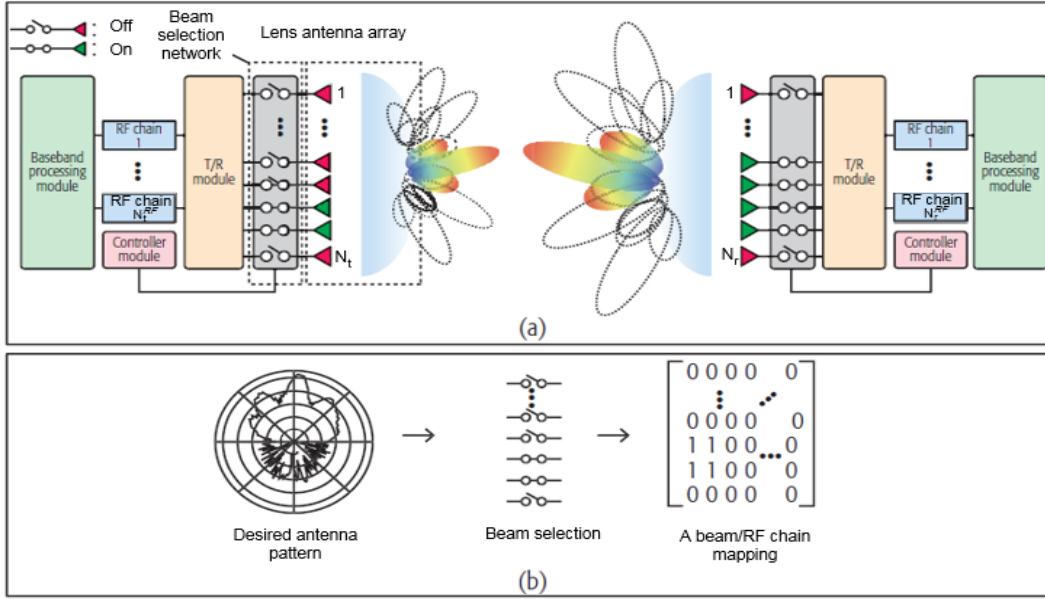


Fig. 1: Schematic of beamspace technology: (a) architectural components; (b) analog beam selection.

architecture that preserves system flexibility as well as efficiency in hardware cost and energy consumption [16]. Under this architecture, transceivers employ a baseband module responsible for digital precoding/combining, which controls the amplitude/phase of the data streams, an analog beam selection network for effective mapping among the RF chains and the predominant beams, as well as an electromagnetic lens for focusing the transmit/receive energy. In Fig. 1, the transmitter and receiver architectures are demonstrated, where the transmitted data symbols are precoded to mitigate inter-symbol interference. To precode the transmit symbols $\mathbf{s} \in \mathbb{C}^{N_s \times 1}$, the transmitter uses a baseband digital matrix $\mathbf{F}_{\text{BB}} \in \mathbb{C}^{N_t^{\text{RF}} \times N_s}$. The transmit symbols are assumed to be power-normalized, i.e., $\mathbb{E}[\mathbf{s}\mathbf{s}^H] = \mathbf{I}_{N_s}$. At the transmitter, N_t^{RF} transmit RF chains are mapped onto a subset of N_t transmit antennas/beams through an analog beam selection network denoted by $\mathbf{S}_t \in \mathbb{R}^{N_t \times N_t^{\text{RF}}}$ in the matrix form. Hence, the complex discrete-time transmit baseband signal is given by

$$\mathbf{x} = \sqrt{\rho/N_s} \mathbf{S}_t \mathbf{F}_{\text{BB}} \mathbf{s}, \quad (1)$$

with ρ denoting the transmit power. At the transmitter side, a lens antenna array is deployed, which includes an electromagnetic lens capable of energy-focusing, and its focal surface is equipped with a large-scale antenna array.

At the receiver side, once the lens antenna array receives the signals, a mapping is performed between the predominant receive antennas/beams and the receive RF chains through the receive analog beam selection network $\mathbf{S}_r \in \mathbb{R}^{N_r \times N_r^{\text{RF}}}$. Similar to the transmitter case, a baseband digital combining matrix $\mathbf{W}_{\text{BB}} \in \mathbb{C}^{N_r^{\text{RF}} \times N_s}$ is employed at the receiver to obtain the transmit symbols. Hence, the discrete-time received baseband complex signal is given by

$$\mathbf{y} = \mathbf{W}_{\text{BB}}^H \mathbf{S}_r^H \mathbf{H}_b \mathbf{x} + \mathbf{W}_{\text{BB}}^H \mathbf{S}_r^H \mathbf{n}, \quad (2)$$

wherein $\mathbf{n} \in \mathcal{CN}(0, \sigma^2 \mathbf{I}_{N_r})$ is the additive white Gaussian noise (AWGN) with noise power σ^2 and \mathbf{H}_b denotes the

beamspace channel obtained through the DFT operations in the lens antenna array as explained in what follows. Therefore, the received signal can be rewritten as

$$\mathbf{y} = \sqrt{\rho/N_s} \mathbf{W}_{\text{BB}}^H \mathbf{S}_r^H \mathbf{H}_b \mathbf{S}_t \mathbf{F}_{\text{BB}} \mathbf{s} + \mathbf{W}_{\text{BB}}^H \mathbf{S}_r^H \mathbf{n}. \quad (3)$$

According to the well-known Saleh-Valenzuela geometric model [17], a ray-based clustered THz channel is assumed with N_{cl} cluster of scatterers. A limited angle-of-departure/arrival (AoD/AoA) spread is further considered for the typical cluster l denoted by ψ_t^l and ψ_r^l , respectively. We further assume that there exist N_{ray} propagation rays, and for a typical cluster/ray l/u , a complex-valued gain is denoted by $\alpha^{l,u}$, while the physical AoD/AoA is given by $\theta_t^{l,u} \in \psi_t^l$ and $\theta_r^{l,u} \in \psi_r^l$, respectively. Let us denote the antenna element spacing by d , the speed of light by c , the wavelength by $\lambda = c/f_c$, and the carrier frequency by f_c . Subsequently, the spatial AoD/AoA can be represented as

$$\phi_t^{l,u} = (d/\lambda) \sin \theta_t^{l,u}, \quad (4)$$

and

$$\phi_r^{l,u} = (d/\lambda) \sin \theta_r^{l,u}, \quad (5)$$

respectively. Accordingly, the narrowband discrete-time spatial domain THz channel $\mathbf{H} \in \mathbb{C}^{N_r \times N_t}$ is expressed as

$$\mathbf{H} = \gamma \sum_{l=1}^{N_{\text{cl}}} \sum_{u=1}^{N_{\text{ray}}} \alpha_{l,u} \mathbf{a}_r(\phi_r^{l,u}) \mathbf{a}_t^H(\phi_t^{l,u}), \quad (6)$$

with the normalization factor of

$$\gamma = \sqrt{N_r N_t / N_{\text{cl}} N_{\text{ray}}}. \quad (7)$$

For the uniform linear array (ULA), the antenna array responses at the transmitter/receiver denoted by $\mathbf{a}_t(\phi_t^{l,u}) \in \mathbb{C}^{N_t \times 1}$ and $\mathbf{a}_r(\phi_r^{l,u}) \in \mathbb{C}^{N_r \times 1}$, respectively, can be represented by

$$\mathbf{a}_t(\phi_t^{l,u}) = \frac{1}{\sqrt{N_t}} [1, e^{j2\pi\phi_t^{l,u}}, \dots, e^{j2\pi(N_t-1)\phi_t^{l,u}}]^H, \quad (8)$$

and

$$\mathbf{a}_r(\phi_r^{l,u}) = \frac{1}{\sqrt{N_r}} \left[1, e^{j2\pi\phi_r^{l,u}}, \dots, e^{j2\pi(N_r-1)\phi_r^{l,u}} \right]^H. \quad (9)$$

With the lens antenna array, the spatial domain THz channel \mathbf{H} is effectively transformed into the equivalent beamspace domain THz channel, which is denoted by $\mathbf{H}_b = \mathbf{U}_r^H \mathbf{H} \mathbf{U}_t$, where

$$\mathbf{U}_t = [\mathbf{a}_t(\bar{\phi}_t^1), \mathbf{a}_t(\bar{\phi}_t^2), \dots, \mathbf{a}_t(\bar{\phi}_t^{N_t})], \quad (10)$$

denotes the transmitter domain transformation, whereas the receiver domain transformation is similarly given by

$$\mathbf{U}_r = [\mathbf{a}_r(\bar{\phi}_r^1), \mathbf{a}_r(\bar{\phi}_r^2), \dots, \mathbf{a}_r(\bar{\phi}_r^{N_r})], \quad (11)$$

where

$$\bar{\phi}_t^n = \frac{1}{N_t} \left(n - \frac{N_t + 1}{2} \right), \forall n \in \{1, 2, \dots, N_t\}, \quad (12)$$

and

$$\bar{\phi}_r^n = \frac{1}{N_r} \left(n - \frac{N_r + 1}{2} \right), \forall n \in \{1, 2, \dots, N_r\}, \quad (13)$$

respectively. Hence, the beamspace channel \mathbf{H}_b can be rewritten as

$$\mathbf{H}_b = \gamma \sum_{l=1}^{N_{cl}} \sum_{u=1}^{N_{ray}} \alpha_{l,u} \bar{\mathbf{a}}_r(\phi_r^{l,u}) \bar{\mathbf{a}}_t^H(\phi_t^{l,u}), \quad (14)$$

with the transmitter/receiver antenna array responses expressed as

$$\begin{aligned} \bar{\mathbf{a}}_t^H(\phi_t^{l,u}) &= \mathbf{U}_t^H \mathbf{a}_t(\phi_t^{l,u}) \\ &= [\Xi_{N_t}(\phi_t^{l,u} - \bar{\phi}_t^1), (\phi_t^{l,u} - \bar{\phi}_t^2), \dots, (\phi_t^{l,u} - \bar{\phi}_t^{N_t})]^T, \end{aligned} \quad (15)$$

and

$$\begin{aligned} \bar{\mathbf{a}}_r(\phi_r^{l,u}) &= \mathbf{U}_r^H \mathbf{a}_r(\phi_r^{l,u}) \\ &= [\Xi_{N_r}(\phi_r^{l,u} - \bar{\phi}_r^1), (\phi_r^{l,u} - \bar{\phi}_r^2), \dots, (\phi_r^{l,u} - \bar{\phi}_r^{N_r})], \end{aligned} \quad (16)$$

with

$$\Xi_{N_{tr}}(x) = \sum_{n=0}^{N_{tr}-1} \frac{e^{j2\pi nx}}{N_{tr}} = \frac{\sin N_{tr}\pi x}{N_{tr} \sin \pi x} e^{j\pi x(N_{tr}-1)}. \quad (17)$$

It is worth noting that $|\Xi_{N_{tr}}(x)| \approx 0$, when $|x| \gg 1$. Hence, $\bar{\mathbf{a}}_t^H$ and $\bar{\mathbf{a}}_r$ are assumed to be sparse vectors and the beamspace channel \mathbf{H}_b with a restricted number of clusters and small AoD/AoA spreads is a sparse matrix with a large number of zero values within it [16].

In a conventional THz massive MIMO system, the number of required RF chains at the transmitter is $N_t^{\text{RF}} = N_t$, which is usually large. From the energy consumption perspective, e.g., about 250mW is consumed by each RF chain, while for a massive MIMO system with $N_t = 256$, about 64W is required for implementation [18]. By exploiting the lens antenna array, a hybrid analog-digital beamspace MIMO system can be developed based on baseband digital beamforming as well as on analog beam selection. In this paper, we contribute to the beamspace MIMO system by addressing its two major drawbacks: beamspace channel estimation and analog beam selection. We, therefore, propose efficient solutions from the

machine learning perspective. By applying analog beam selection to a similar THz massive MIMO system equipped with the beamspace technology, the number of required RF chains is limited to the number of dominant beams, which is far fewer in the THz band, thus leading to reasonable energy consumption levels without significant performance loss.

In the considered THz beamspace system, analog beam selection is crucial for practical implementation. However, it is generally dependent on the communication channel structure, which usually varies over time. Hence, for effective analog beam selection, it is essential to track and predict the communication channel. Accordingly, in the following two sections, we specifically investigate the beamspace channel tracking and the analog beam selection problems.

III. THZ BEAMSPACE CHANNEL TRACKING

Regarding the sparse THz beamspace channel, and for the given \mathbf{F}_{BB} , \mathbf{W}_{BB} , \mathbf{S}_t , and \mathbf{S}_r as the transceiver design parameters³, one can characterize the THz beamspace channel by addressing the following signal recovery optimization problem [8]

$$\begin{aligned} \min_{\mathbf{H}_b} \quad & \|\mathbf{H}_b\|_0 \\ \text{s.t.} \quad & \|\mathbf{y} - \sqrt{\psi/N_s} \mathbf{W}_{\text{BB}}^H \mathbf{S}_r^H \mathbf{H}_b \mathbf{S}_t \mathbf{F}_{\text{BB}} \mathbf{s}\|_2 \leq \varepsilon, \end{aligned} \quad (18)$$

where $\|\cdot\|_0$ denotes the non-zero element numbers and an error tolerance is ε . In other words, we seek for $\|\mathbf{H}_b(t) - \widehat{\mathbf{H}}_b(t)\|_2$ to be minimized, with the predicted THz beamspace channel in time step t denoted by $\widehat{\mathbf{H}}_b(t)$. Since the THz beamspace channel is of high sparsity order, classical compressed sensing schemes with random measurements, such as the ones in [10] and [19], fail to provide an accurate beamspace channel estimation. Contrarily, prior information of the beamspace channel can be efficiently utilized for higher precision [20].

Motivated by this, we propose a modified version of the NARX for more precise tracking of the physical AoD/AoA directions in prior time steps and future prediction. Once the physical AoD/AoA directions are predicted at the next time step, one can outline the support set of the THz beamspace channel and derive its non-zero elements through pilot transmission [20] by using \mathbf{s} and \mathbf{y} in (18).

A. Training Sample Set Acquisition

Let us define $\boldsymbol{\chi} \stackrel{\bar{n}}{=} [\boldsymbol{\theta}_r, \boldsymbol{\theta}_t]$ as a motion feature state vector for the variation of AoDs/AoAs with $\boldsymbol{\theta}_r = \{\theta_r^1, \theta_r^2, \dots, \theta_r^{N_r^{\text{eff}}}\}$ and $\boldsymbol{\theta}_t = \{\theta_t^1, \theta_t^2, \dots, \theta_t^{N_t^{\text{eff}}}\}$, where

$$|N_t^{\text{eff}}| = |N_t| \times |N_{\text{cl}}| \times |N_{\text{ray}}|, \quad (19)$$

and

$$|N_r^{\text{eff}}| = |N_r| \times |N_{\text{cl}}| \times |N_{\text{ray}}|, \quad (20)$$

respectively. Following [11], it is generally assumed that the AoDs/AoAs have a time-varying nature and follow their

³We initialize the analog beam selection and the digital baseband precoding/combining matrices based on the transceiver design scheme in [30] before proceeding with the beamspace channel tracking.

historical time steps $\chi(1), \chi(2), \dots, \chi(t-1)$. According to the temporal variation law of physical directions [20], AoD/AoA values can be tracked efficiently during the time without beamspace channel estimation or pilot signaling being required. In fact, the AoD/AoA values are time-varying features of mobile transceivers⁴. Based on this, we train a modified version of the NARX model to accurately forecast $\chi(t)$.

B. Time-Series Forecasting

Artificial neural networks (ANNs) with temporal dynamic behavior and time-varying structure are capable of processing time-series information (i.e., sequences of information across time steps). The time-series ANN can track the historical knowledge and forecast the desired sequences of data in the forthcoming time steps. Owing to the dynamic characteristics of the transceiver feature states [11], they can be modeled in a time-series form. Hence, the time-series ANNs can efficiently track their historical time steps and forecast them in the future.

To this aim, we use NARX as a capable time-series ANN benefiting from precise tracking capability that is due to considering the variations of environmental variables [22]. Additionally, we employ the Levenberg-Marquardt policy to intelligently train the NARX parameters (i.e., its weights and biases). This policy interpolates between the Gauss-Newton and the gradient descent methods, which leads to remarkably lower prediction variance. For implementing the NARX, parallel and series-parallel models have been proposed thus far. In the rest of this section, we elaborate on the training and forecasting phases of NARX based on the beamspace channel features (i.e., AoDs/AoAs), and explain how to train the NARX using the Levenberg-Marquardt policy. Further, we properly initialize the NARX hyperparameters by means of a Bayesian optimizer. An analysis of the approximation and generalization errors is presented as well.

1) *NARX training phase*: Under the assumption of known historical AoA ground truth values $\{\theta_r(0), \theta_r(1), \dots, \theta_r(t-1)\}$ as well as the historical AoDs $\{\theta_t(0), \theta_t(1), \dots, \theta_t(t-1)\}$, we aim at training the NARX to predict the AoAs in time step t as denoted by $\hat{\theta}_r(t)$. The main input of NARX according to Fig. 2(a), is the historical AoA ground truth values, while the historical AoDs are also injected as the exogenous input. The output of NARX can be given by [22]

$$\hat{\theta}_r(t) = f^{\text{NARX}}\left(\theta_r(t-1), \theta_r(t-2), \dots, \theta_r(t-n_{\theta_r}), \theta_t(t-1), \theta_t(t-2), \dots, \theta_t(t-n_{\theta_t})\right), \quad (21)$$

where f^{NARX} denotes the NARX mapping function. For the hidden layers of NARX, we adopt ReLU as a simple activation function. Further, n_{θ_t} and n_{θ_r} are the numbers of input/output delays indicating the number of historical AoD/AoA time steps. As demonstrated in Fig. 2(a), we use the series-parallel architecture with an open-loop structure for the training phase of NARX, which comprises an MLP network. The training phase is, therefore, data-driven (i.e., supervised), which is

⁴Due to highly dynamic features of the THz channel including blockage, LoS and NLoS links may change abruptly. Therefore, it is necessary to take all of the AoD/AoA values into consideration while tracking, not only those corresponding to the dominant beams.

more precise than in unsupervised models due to the availability of historical ground truth AoAs as the training samples.

2) *NARX forecasting phase*: In this phase, it is assumed that the ground truth AoAs are unavailable. More precisely, for the n_{θ_r} latest time steps, we only have the NARX forecasted AoAs denoted by $\hat{\theta}_r(t-1), \hat{\theta}_r(t-2), \dots, \hat{\theta}_r(t-n_{\theta_r})$. Similar to the previous phase, we seek to forecast $\theta_r(t)$. In this case, we use the parallel architecture of NARX with a close-loop structure as in Fig. 2(b), where the output can be given as

$$\hat{\theta}_r(t) = f^{\text{NARX}}\left(\begin{matrix} \hat{\theta}_r(t-1), \hat{\theta}_r(t-2), \dots, \hat{\theta}_r(t-n_{\theta_r}), \\ \theta_t(t-1), \theta_t(t-2), \dots, \theta_t(t-n_{\theta_t}) \end{matrix}\right). \quad (22)$$

The series-parallel NARX architecture with recursive feedback employed by this phase can realize multi-step ahead forecasting⁵ [23].

3) *Levenberg-Marquardt policy*: To train the NARX, we employ the Levenberg-Marquardt policy based on the least square principle. As an interpolation between the Gauss-Newton and the gradient descent method, this policy aims at minimizing the least square errors in problems with non-linear structure as well as at approaching a generic curve-fitting. Within a trust region, this policy can achieve the local minima under appropriate initial values of its parameters. The objective here is to seek for the NARX training parameters, i.e., weight and bias vectors denoted by $\tau = \{W_{\text{NARX}}, b_{\text{NARX}}\}$, to model any curve $g(\hat{\theta}_r(t), \tau)$, such that the total deviation from the ground truth $\theta_r(t)$ is minimized. Hence, the optimal training parameters can be acquired as:

$$\tau^* = \arg \min_{\tau} Dev(\tau) = \arg \min_{\tau} \sum_{i=1}^{\|N_{\text{tr}}^{\text{eff}}\|} \left[\theta_r^i(t) - g\left(\hat{\theta}_r^i(t), \tau\right) \right]^2. \quad (23)$$

For curve-fitting, an approximation of the curve is given by

$$g\left(\hat{\theta}_r(t), \tau + \xi\right) = g\left(\hat{\theta}_r(t), \tau\right) + \xi Gr(\tau), \quad (24)$$

where ξ is a shifting value and

$$Gr(\tau) = \frac{\partial g\left(\hat{\theta}_r(t), \tau\right)}{\partial \tau}. \quad (25)$$

Hence, with respect to the aforementioned approximation, the square of deviations in τ^* can be restated as

$$Dev(\tau) = \sum_{i=1}^{\|N_{\text{tr}}^{\text{eff}}\|} \left[\theta_r^i(t) - g\left(\hat{\theta}_r^i(t), \tau\right) - \xi Gr(\tau) \right]^2. \quad (26)$$

Facilitated by the Gauss-Newton method, the best value of shifting ξ can be achieved by $\xi^* = \frac{\partial Dev(\tau)}{\partial \xi} = 0$, which leads to

$$Gr^T(\tau) Gr(\tau) \xi^* = Gr^T(\tau) \sum_{i=1}^{\|N_{\text{tr}}^{\text{eff}}\|} \left[\theta_r^i(t) - g\left(\hat{\theta}_r^i(t), \tau\right) \right]^2. \quad (27)$$

⁵Owing to separated architectures for training and prediction as well as to the exogenous input gate, the NARX model exhibits a higher degree of precision in predictivity, as compared to its counterparts, e.g., LSTM model.

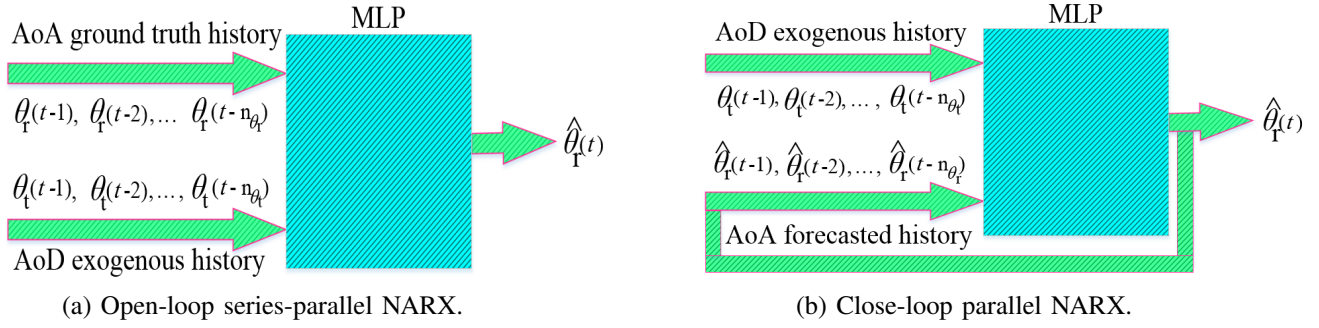


Fig. 2: NARX architectures for training and forecasting phases.

The Levenberg-Marquardt policy [24] introduces a damped version as

$$(Gr^T(\boldsymbol{\tau})Gr(\boldsymbol{\tau}) + \beta\mathbf{I})\boldsymbol{\xi}^* = Gr^T(\boldsymbol{\tau}) \sum_{i=1}^{||N_{\theta_r}^{\text{eff}}||} \left[\boldsymbol{\theta}_r^i(t) - g\left(\hat{\boldsymbol{\theta}}_r^i(t), \boldsymbol{\tau}\right) \right]^2, \quad (28)$$

with the damping factor β updated in an iterative manner. In each iteration toward minimizing the deviation, the Levenberg-Marquardt policy tends to the Gauss-Newton method for smaller β , whereas for larger β , the gradient descent method is approached. The process terminates when the squared deviation falls below a predetermined error threshold.

4) *Bayesian Optimizer*: To improve the convergence behavior of NARX, the hyperparameters such as the learning rate or the number of hidden layers in its MLP need to be initialized efficiently before training [25]. To this end, we consider a degree of error ε_j^i for the j th neuron and the i th training sample in the final layer of the MLP. Henceforth, we seek to minimize the total degree of error for each training sample i represented by

$$\varepsilon^i = \frac{1}{2} \sum_{j \in \mathcal{J}_{N_{\text{Lay}}}} \left[\boldsymbol{\theta}_r^i(t) - \hat{\boldsymbol{\theta}}_r^i(t) \right]_j, \quad (29)$$

where $\mathcal{J}_{N_{\text{Lay}}}$ indicates the set of neuron indices in the final layer of the MLP. The training parameters $\boldsymbol{\tau}$ corresponding to the i th training sample and the j th neuron are updated as

$$\Delta \boldsymbol{\tau}_j^i = -\eta \frac{\partial \varepsilon_j^i}{\partial \zeta_j^i} \left[\boldsymbol{\theta}_r^i(t) \right]_j, \quad (30)$$

where ζ_j^i is the induced local field and η denotes the learning rate.

For faster convergence with a limited number of training epochs, the hyperparameter η should be well-initialized. To this aim, a promising method is the so-called Bayesian optimizer, wherein the prior states of the hyperparameter are determinant in updating its current state. Compared to the grid search method, the number of iterations in the Bayesian approach to acquire the optimal initial values is much lower due to pruning the non-optimistic choices observed in prior states. In more detail, a probability distribution is considered here for the learning rate hyperparameter η in the MLP. An evaluation function is then responsible for the assessment of any candidate within the probability distribution. Since the least square error minimization in the training MLP is a regression problem, we define f^{loss} , which is a loss-based

evaluation function⁶ that returns $\bar{\eta}$, for temporal benchmarking to evaluate the next candidates of η , as follows $\bar{\eta} = f^{\text{loss}}(\eta)$. The optimal initial value of the hyperparameter η can be approached when the loss function is minimized as

$$\eta^* = \arg \min_{\eta} f^{\text{loss}}(\eta). \quad (31)$$

The Bayesian optimizer then traces the candidates evaluated thus far to select the best one in terms of the loss function minimization. However, evaluating the loss function f^{loss} for a vast search domain of the hyperparameter η imposes a heavy computational burden. To alleviate the computation load, a surrogate function approximates f^{loss} with respect to the posterior observations of η . On the basis of Bayes theorem [25], the surrogate function under given η as a probability distribution can be expressed as

$$\Pr(\bar{\eta}|\eta) = \frac{\Pr(\eta|\bar{\eta})\Pr(\bar{\eta})}{\Pr(\eta)}, \quad (32)$$

which can be implemented by the well-known tree Parzen estimator (TPE) [26] that acts as a probabilistic model on the loss function f^{loss} for mapping the hyperparameter η onto a loss distribution $\bar{\eta}$. In the last step, the expected improvement (EI) criterion is employed for estimating the TPE-implemented surrogate function according to the so-called “exploration and exploitation” trade-off principle [27].

Therefore, the hyperparameter optimization based on the Bayesian theorem can be categorized into several steps. First, the hyperparameter domain (of a grid or a probability distribution) is to be searched via an objective function (which in our regression problem is stated as a loss function). A loss value, i.e., $\bar{\eta}$ is returned by the loss function indicating the evaluation score. A surrogate function is also employed for approximating the loss function and initiating a mapping between the hyperparameter values η and the corresponding loss value $\bar{\eta}$ according to the Bayes theorem. The best choice for the hyperparameter is adopted by the well-known criterion EI according to the posterior observations. Building upon a history of (hyperparameter, score) pairs, updating process is performed on the surrogate function up until a stop condition (such as the maximum number of iterations) is reached.

⁶It is worth noting that in regression problems, evaluations are performed to minimize the regression loss, while in classification problems, they are aimed at maximizing the classification accuracy.

C. Generalization Error Analysis

The proposed ReLU-driven NARX estimator can be treated as a minimum mean square error (MMSE) estimator when two main requirements are met. First, the NARX structural elements, i.e., the set of weights, biases, and learning rate denoted by $\Gamma = \{\mathbf{W}_{\text{NARX}}, \mathbf{b}_{\text{NARX}}, \eta\}$ need to be properly configured. Second, the size of the training and validation sample set is large enough. The NARX mapping function in (21) and (22) with the specified structural elements Γ can be represented by

$$f_{\Gamma}^{\text{NARX}}(\theta_r) = \bar{\Gamma}_l \circ f^{\text{ReLU}} \bar{\Gamma}_{l-1} \circ f^{\text{ReLU}} \circ \dots \circ \bar{\Gamma}_0(\theta_r), \quad (33)$$

where $\bar{\Gamma}_l$ indicates an affine transformation for Γ in the l th layer, \circ is the function composition, and f^{ReLU} represents the ReLU activation function. In more detail, for the sample set $\chi = \left\{ \left(\theta_r(t), \hat{\theta}_r(t) \right) \right\}$, we define the expected loss and the empirical loss as

$$\mathcal{L}(f_{\Gamma}^{\text{NARX}}) = \mathbb{E} \left\{ \left\| f_{\Gamma}^{\text{NARX}} \left(\theta_r(t) - \hat{\theta}_r(t) \right) \right\|_2^2 \right\}, \quad (34)$$

and

$$\mathcal{L}_{\chi}(f_{\Gamma}^{\text{NARX}}) = \frac{1}{\|\chi\|} \sum_{(\theta_r(t), \hat{\theta}_r(t))} \left\| f_{\Gamma}^{\text{NARX}} \left(\theta_r(t) - \hat{\theta}_r(t) \right) \right\|_2^2, \quad (35)$$

respectively. We further define

$$\Gamma^* = \arg \min_{\Gamma} \mathcal{L}(f_{\Gamma}^{\text{NARX}}), \quad (36)$$

and

$$\mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) = \min_{\Gamma} \mathcal{L}(f_{\Gamma}^{\text{NARX}}), \quad (37)$$

as the optimal NARX configuration, as well as the minimum mean square error (MSE) achieved among all the configurations for the NARX, respectively. Accounting for the sampling set χ , we can similarly define

$$\Gamma^* = \arg \min_{\Gamma} \mathcal{L}_{\chi}(f_{\Gamma}^{\text{NARX}}), \quad (38)$$

and

$$\mathcal{L}_{\chi}(f_{\Gamma^*}^{\text{NARX}}) = \min_{\Gamma} \mathcal{L}_{\chi}(f_{\Gamma}^{\text{NARX}}), \quad (39)$$

where

$$\mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) \geq 0. \quad (40)$$

Hence, we have

$$\mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) = \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) + \left[\mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) \right], \quad (41)$$

where $\left[\mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) \right]$ indicates the generalization error reliant on the sampling set for a fixed configuration of NARX, while $\mathcal{L}(f_{\Gamma^*}^{\text{NARX}})$ is captured by the optimized configuration of NARX irrespective of the training samples and is given by [28]

$$\mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) = \mathbb{E} \left\{ \left\| f_{\Gamma^*}^{\text{NARX}} - f^* \right\|_2^2 \right\} + \mathcal{L}(f^*), \quad (42)$$

where $\mathbb{E} \left\{ \left\| f_{\Gamma^*}^{\text{NARX}} - f^* \right\|_2^2 \right\}$ represents the approximation error in a finite-length ReLU-driven NARX. The performance of the proposed deep learning structure can be assessed by investigating the aforementioned approximation and generalization errors. Here, the approximation error for any given precision can be bounded as per the following lemma.

Lemma 1. *For a given precision level $\varepsilon > 0$, there exists a ReLU-driven NARX estimator f^* that employs an optimized configuration and incorporates up to $\lceil \log_2(L_{\text{max}}^{\text{NARX}} + 1) \rceil$ hidden layers with $L_{\text{max}}^{\text{NARX}} = N_r \times N_t$, provided that*

$$\mathbb{E} \{ \|f_{\Gamma^*}^{\text{NARX}} - f^*\|_2^2 \} < \varepsilon. \quad (43)$$

Proof. See Appendix A for the proof. \square

To complement, the generalization error becomes negligible when the number of training samples is large enough as set in the following lemma.

Lemma 2. *For finite-length $\|\Gamma^*\|_2$ and $\|\Gamma_{\chi}\|_2$, as well as for the probability convergence $\xrightarrow{\text{Pr}}$, we can see that*

$$\mathcal{L}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) \xrightarrow{\text{Pr}} 0. \quad (44)$$

Proof. See Appendix B for the proof. \square

Relying on Lemma 2, the following corollary reflects the efficiency of the proposed ReLU-driven NARX structure.

Corollary 1. *For finite-length $\|\Gamma^*\|_2$ and $\|\Gamma_{\chi}\|_2$, there exists a ReLU-driven NARX estimator that incorporates up to $\lceil \log_2(L_{\text{max}}^{\text{NARX}} + 1) \rceil$ hidden layers with $L_{\text{max}}^{\text{NARX}} = N_r \times N_t$, provided that for a given precision $\varepsilon > 0$*

$$\lim_{\|\chi\| \rightarrow +\infty} \text{Pr}(\left| \mathcal{L}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}(f^*) \right| > \varepsilon) = 0. \quad (45)$$

Proof. See Appendix C for the proof. \square

IV. ANALOG BEAM SELECTION

In the considered hybrid analog-digital beamspace massive MIMO system, we focus on the digital baseband precoding/combining and analog beam selection problems for the transmitter and the receiver under the assumption of a given beamspace channel. This problem can be formally stated⁷ as in [29]

$$\begin{aligned} & \min_{\mathbf{W}_{\text{BB}}, \mathbf{F}_{\text{BB}}, \mathbf{S}_t, \mathbf{S}_r} \|\mathbf{H}_b - \mathbf{S}_r \mathbf{W}_{\text{BB}} \mathbf{F}_{\text{BB}}^H \mathbf{S}_t^H\|^2 \\ & \text{s.t. } \mathbf{S}_r \in \mathcal{S}_r, \mathbf{S}_t \in \mathcal{S}_t, \mathbf{W}_{\text{BB}} \in \mathcal{W}_{\text{BB}} \text{ and } \mathbf{F}_{\text{BB}} \in \mathcal{F}_{\text{BB}}, \end{aligned} \quad (46)$$

where \mathcal{S}_t (\mathcal{F}_{BB}) and \mathcal{S}_r (\mathcal{W}_{BB}) are the analog beam selection (digital baseband) candidate sets at the transmitter and the receiver, respectively. The optimal baseband precoding/combining problem for a given beamspace channel \mathbf{H}_b can be considered as $\mathbf{F}_{\text{BB}}^* = \left(\frac{\mathbf{V}}{\mathbf{S}_t^H \mathbf{S}_r} \right)$ and $\mathbf{W}_{\text{BB}}^* = \bar{\mathbf{U}}$, where $\bar{\mathbf{V}}$

⁷Note that since beamspace channel estimation errors are inevitable, the baseband precoding/combining matrices, in turn, cannot be calculated exactly at the transceivers. Hence, the subject optimization problem is aimed at minimizing the difference between the estimated beamspace channel and the corresponding beamspace channel, as calculated by relying on the transceiver design parameters $\mathbf{S}_r \in \mathcal{S}_r, \mathbf{S}_t \in \mathcal{S}_t, \mathbf{W}_{\text{BB}} \in \mathcal{W}_{\text{BB}}$, and $\mathbf{F}_{\text{BB}} \in \mathcal{F}_{\text{BB}}$.

and $\bar{\mathbf{U}}$ correspondingly denote the right and the left singular vector matrices of \mathbf{H}_b [30]. The optimal solution for the analog beam selection problem is via an exhaustive search method that is computationally expensive. For a typical THz massive MIMO system with $N_t = 256$ and $N_r^{\text{RF}} = 32$, the total number of searches in \mathcal{S}_t is as large as 6×10^{40} , which is impractical from the computational perspective. In what follows, we detail our solution approach to this problem.

A. Training Sample Set Acquisition

We consider a multi-snapshot scenario to obtain the training sample set, where the network parameters such as path gain, transmit power, AoD, and AoA change depending on the location of the transceivers in each snapshot [31]. For each training sample, we consider $4N_{\text{cl}} \times N_{\text{ray}} + 2$ random real-valued features including one feature for the transmit power of the transmitter, one feature for the normalization factor, $2N_{\text{cl}} \times N_{\text{ray}}$ features for the AoDs/AoAs of the transmitter/receiver, and hence $2N_{\text{cl}} \times N_{\text{ray}}$ features for the real and imaginary parts of the complex-valued gain to form a sample. In what follows, we conduct a normalization process, a Gaussian mixture model (GMM) fitting, and a labeling operation over the training samples for analog beam selection.

1) *Normalization*: Due to diversity in the training sample ranges (e.g., transmit power is based on dB, while AoDs are within $(0, 2\pi]$), a normalization pre-processing needs to be conducted for every feature in each training sample as

$$\bar{a}_f^m = \frac{a_f^m - \text{Mean}(a_f^m)}{a_f^{\max} - a_f^{\min}}, \quad (47)$$

where a_f^m indicates the value of the f th feature in the m th training sample and $\text{Mean}(a_f^m)$ is the mean of all a_f^m . Further, a_f^{\max} and a_f^{\min} denote the maximum and minimum values of the f th feature among all the training samples, respectively. Hence, the m th training sample as a feature row vector can be characterized as $\mathbf{z}_m \in \mathbb{C}^{1 \times (4N_{\text{cl}} \times N_{\text{ray}} + 2)}$ with $4N_{\text{cl}} \times N_{\text{ray}} + 2$ normalized features.

2) *GMM Fitting*: Since the beamspace channel features ϕ_t , ϕ_r , and α follow a Gaussian distribution [8], we adopt a GMM for appropriately fitting the beamspace channel. Therefore, we have:

$$\tilde{\mathbf{H}}_b = A \times \left(\sum_{k=1}^K w_k \exp \left(- \frac{(\phi_r - \mu_{\phi_{rk}})^2}{2\sigma_{\phi_{rk}}^2} - \frac{(\phi_t - \mu_{\phi_{tk}})^2}{2\sigma_{\phi_{tk}}^2} - \frac{(\phi_r - \mu_{\alpha_k})^2}{2\sigma_{\alpha_k}^2} \right) \right), \quad (48)$$

with the GMM-fitted beamspace channel $\tilde{\mathbf{H}}_b$, GMM amplitude A , and K Gaussian components, where $w_k \in [0, 1]$ is the weight of the Gaussian component k and $\sum_{k=1}^K w_k = 1$. Note that in (48), the central coordinates are $(\mu_{\phi_{rk}}, \mu_{\phi_{tk}}, \mu_{\alpha_k})$, whereas $\sigma_{\phi_{rk}}$, $\sigma_{\phi_{tk}}$, and σ_{α_k} indicate their corresponding standard deviations. In a vector representation, the Gaussian component k can be expressed as

$$q_k = [w_k, \mu_{\phi_{rk}}, \mu_{\phi_{tk}}, \mu_{\alpha_k}, \sigma_{\phi_{rk}}, \sigma_{\phi_{tk}}, \sigma_{\alpha_k}]. \quad (49)$$

Equivalently, the spatial features of the training samples based on all of the Gaussian components can be given by

$$\mathbf{q} = [A; q_1; q_2; \dots; q_K]^T = [A, \mu_{\phi_{r1}}, \mu_{\phi_{t1}}, \mu_{\alpha_1}, \sigma_{\phi_{r1}}, \sigma_{\phi_{t1}}, \sigma_{\alpha_1}, \mu_{\phi_{r2}}, \mu_{\phi_{t2}}, \mu_{\alpha_2}, \sigma_{\phi_{r2}}, \sigma_{\phi_{t2}}, \sigma_{\alpha_2}, \dots, \mu_{\phi_{rK}}, \mu_{\phi_{tK}}, \mu_{\alpha_K}, \sigma_{\phi_{rK}}, \sigma_{\phi_{tK}}, \sigma_{\alpha_K}]^T. \quad (50)$$

The optimal vector \mathbf{q} , which is used to model the beamspace channel distribution, can be determined according to [32].

3) *Labeling*: Each training sample derived from the beamspace channel domain is thus unlabeled and needs an RF chain to be assigned. Due to the importance of accurate labeling, we adopt the near-optimal Gram-Schmidt [33] method for evaluating the analog beam selection decisions. Accordingly, the pairs (beam, RF) comprising a training sample obtained from the beamspace channel (i.e., a typical energy-focused analog beam), as well as a typical RF chain, are the candidates that are evaluated in terms of the cost function. From the classification viewpoint, each RF chain is a class label, to which analog beams are assigned. Therefore, transmit (receive) analog beam selection is a multi-class mapping problem, wherein the number of classes is N_t^{RF} at the transmitter (or N_r^{RF} at the receiver).

4) *Loss Function*: The loss function is $\sum_{i=1}^{\|N_{\text{tr}}^{\text{RF}}\|} \left[RF^i - \widehat{RF}^i \right]^2$, with RF^i and \widehat{RF}^i indicating the ground truth label and predicted label for the i th sample, respectively.

B. GoogleNet Architecture

GoogleNet has been developed by researchers at Google corporation [34] based on the convolutional neural network (ConvNet) and scored the best classification results in ImageNet large-scale visual recognition challenge 2014 (ILSVRC14). As an off-the-shelf pre-trained network, GoogleNet has been trained by well-known datasets (e.g., ImageNet) beforehand, while its weights, biases, and other training parameters have been already preset. According to Fig. 3, the network has 22 layers with an input layer of size $224 \times 224 \times 3$ for receiving a two-dimensional (2D) image of width and length 224 and 3 channels of RGB (i.e., red, green, and blue).

The main parts of the GoogleNet architecture are its inception modules that incorporate multiple convolutions, kernels, and max-pooling layers simultaneously within a single layer. The inception modules can also effectively reduce the dimensionality of the input and ensure that the network trains with optimal weights. The main activation function in GoogleNet is ReLU, which is computationally inexpensive and embedded upon a filter concatenation layer within the inception module (see Fig. 3) for improved training performance. Going deeper into the GoogleNet architecture as observed in Fig. 3, the linear layer of size 1000 is followed by a dropout layer with 40% ratio of dropped outputs and connected to a Softmax activation function with 1000 classes.

C. Swish-driven GoogleNet

Despite its notable classification capability, the performance of GoogleNet can still be improved by further architectural

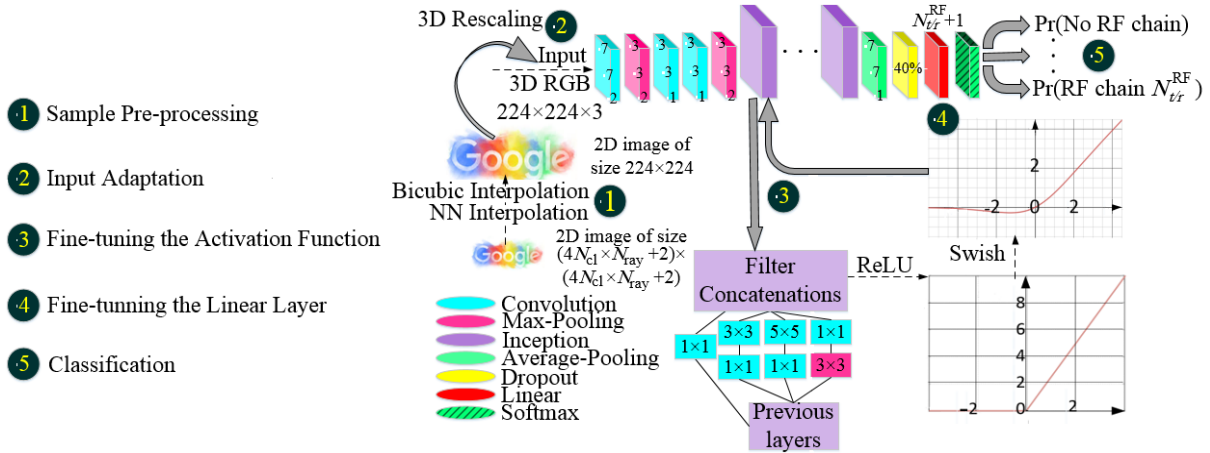


Fig. 3: Architecture of GoogleNet, modifications performed on training samples to fit into the input layer, replacing ReLU with Swish, and setting the number of linear layer classes from 1000 to $N_t^{\text{RF}} + 1$ (or $N_r^{\text{RF}} + 1$).

modifications. For instance, the authors in [35] proposed to substitute the ReLU activation function of GoogleNet with Leaky-ReLU (an extension of the conventional ReLU) for faster convergence. In [36], large convolutional filters in GoogleNet were factorized into smaller ones, and this modification benefited the middle layers of GoogleNet. In this paper, we modify the ReLU activation function in the filter concatenation layer of the inception modules (see Fig. 3) within the GoogleNet architecture by Swish [37]. The latter is a self-gated, smooth, and non-monotonic activation function recently proposed by the Google Brain Team. By definition, the Swish activation function for any input x can be given by $f^{\text{Swish}}(x) = x \cdot f^{\text{Sigmoid}}(x) = \frac{x}{1 + e^{-x}}$. The numerical results in [37] indicate that Swish is more precise than ReLU (and its possible extensions, such as Leaky-ReLU) while having a similar level of computational complexity, especially in deeper architectures.

D. Transfer Learning

To fit the size of the training samples into the input layer of the fine-tuned Swish-driven GoogleNet, certain modifications are required in accordance with Fig. 3. First, a 2D image of size $(4N_{cl} \times N_{ray} + 2) \times (4N_{cl} \times N_{ray} + 2)$ is derived from the samples by using the nearest neighbor (NN) interpolation, while its outcome is resized via a bicubic interpolation into the size of 224×224 . These transformations preserve the quality of primary samples [38] by extracting the most determinant features, which correspondingly relate to the LoS links with power concentration (see [8] and [16] for more details). The 224×224 resized 2D image is then extended into a three-dimensional (3D) image, where the RGB color triplet for each element is set separately [39], thus producing a 3D RGB image of size $224 \times 224 \times 3$ to be fed into the input layer of the Swish-driven GoogleNet.

We further adjust the final linear layer of the fine-tuned GoogleNet by setting $N_t^{\text{RF}} + 1$ classes for the transmitter (or $N_r^{\text{RF}} + 1$ for the receiver), which trains the GoogleNet to map any sample (beam) onto the correct class (RF chain). During the training process, the beamspace channel feature space is processed through the layers of GoogleNet, while its main

features (energy-focused features of the beam) are extracted. The Softmax classifier eventually learns a multi-class mapping based on the labeled training samples obtained from [33]. The probability of the i th RF chain being selected by the Softmax function is

$$\delta(N_t^{\text{RF}})_i = [e^{(N_t^{\text{RF}})_i}] \times \left[\sum_{i=1}^{N_t^{\text{RF}}} e^{(N_t^{\text{RF}})_i} \right]^{-1}. \quad (51)$$

Therefore, as observed in Fig. 3, our modified GoogleNet version is trained by fine-tuning its linear layer and activation functions. This approach is known as transfer learning, whereby the main layers of a pre-trained network are directly imported into a new application, while other layers remain unchanged. Accordingly, the fine-tuned GoogleNet learns analog beam selection at the transceivers based on the beamspace channel feature space, while its internal weights, biases, and other parameters remain largely unchanged. Hence, off-the-shelf pre-trained networks that are fine-tuned based on the transfer learning principles (e.g., GoogleNet) are less vulnerable to accuracy loss and do not need to be trained from scratch. For this reason, the classification accuracy related to fine-tuning the GoogleNet architecture (i.e., retraining its certain layers) is higher than when training a conventional ConvNet from scratch [40].

It is also important to note that deep networks are highly susceptible to overfitting⁸ due to incorporating a massive number of weights and biases. The fine-tuned GoogleNet is, however, largely immune to this effect due to its architectural and practical advantages. First, dropout regularization and batch normalization are envisioned and efficiently embedded beforehand into the GoogleNet architecture [34]. Second, owing to the transfer learning technique, weights and biases mostly remain unchanged during the training process. Therefore, once limited weights and biases of the GoogleNet are retrained and adequately adapted to the training samples, no further modifications are required for mapping the new observations onto the appropriate output classes.

⁸Commonly occurs in an unstable trained network when small variations in inputs of the network lead to a significant unpredicted output.

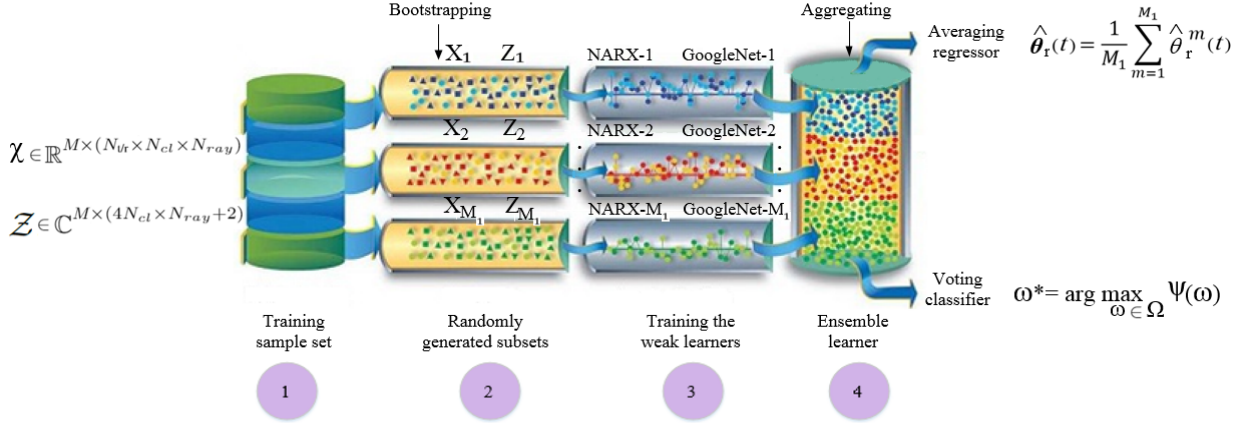


Fig. 4: Ensemble learning schematic.

V. IMPROVING ACCURACY VIA ENSEMBLE LEARNING

Ensemble learning is an efficient technique, by which a strong ensemble learner combines the predictions for a multitude of weak learners to acquire a more accurate prediction. Bootstrap aggregation, as one of the simplest algorithms for ensemble learning, is known to be capable of enhancing the prediction accuracy versus any of its weak learners⁹ [41]. This method, as displayed in Fig. 4, relies on a diversity of weak learners trained over the bootstrapped replicas (i.e., subsets) of the training sample data.

By drawing random subsets, the weak learners can be trained simultaneously in an independent fashion, wherein each learner is trained on a different subset. A combination phase is eventually performed to aggregate the predictions of the weak learners, so as to train the strong ensemble learner. For a classification problem, the strong ensemble learner prediction class is the one adopted by most of the weak learners (classifiers). Similarly, for a regression problem, the final prediction of the strong ensemble learner is an average over the predictions made by its weak learners (i.e., regressors). The bootstrap aggregation algorithm (also known as bagging) for analog beam selection and beamspace channel tracking, as a classification/regression problem, can be detailed as follows.

- **Classification:** We adopt M random subsets $Z_m (m \in M)$ of the entire training sample set \mathcal{Z} . We also consider M Swish-driven GoogleNet classifiers as weak learners for ensembling. The learners are trained over the different training subsets independently in a parallel fashion. For any unseen sample $z_m \in Z_m$ of size $\mathbb{C}^{1 \times (4N_{cl} \times N_{ray} + 2)}$, the classifiers perform a classification task and assign it a specific class from $\omega_m \in \Omega = \{0, \dots, N_{ur}^{RF}\}$. The ensemble learner then collects the predicted classes, with its prediction being the class adopted by a majority voting mechanism among the classifiers. To this aim, a voting counter $\Psi \in \mathbb{N}^{1 \times \Omega}$ indicates the total number of classifiers adopted by each RF chain class. The class with

the highest value of the voting counter is thus adopted as a prediction of the ensemble learner.

- **Regression:** Similar to the classification approach, M random subsets $X_m (m \in M)$ of the entire training sample set χ are considered for training M NARX regressors as weak learners. Once the weak learners (i.e., NARX regressors) are trained simultaneously, a typical unseen sample $x_m \in X_m$ can be predicted by each regressor. The ensemble learner then collects the predicted values, with its prediction given by averaging over the predictions made by the regressors. Note that for the case where the adopted regressors are of various types, a weighted averaging mechanism is to be more accurate, wherein the involved regressors with higher prediction accuracy have more impact on the final prediction for the ensemble learner.

It is also worth noting that it may be preferable for weak learners to be trained in multiple iterations over different subsets to avoid an accuracy loss.

VI. COMPLEXITY ANALYSIS

In this section, the order of complexity for the proposed methods of beamspace channel tracking and analog beam selection is investigated. The computational complexity of training a deep learning network mainly depends on the function that it is trained with. The computational complexity of training the proposed fine-tuned GoogleNet via the conventional stochastic gradient descent method (SGDM) is given by

$$\mathcal{O}_{GN} \left(I_{conv} M (4N_{cl} \times N_{ray} + 2) \right), \quad (52)$$

where I_{conv} is the number of iterations for the fine-tuned GoogleNet to converge. The computational complexity of training the proposed fine-tuned NARX via the Levenberg-Marquardt policy is expressed as

$$\mathcal{O}_{NA} \left(\mathcal{C} \left(1 + \log_{\Upsilon} \left[\frac{\kappa}{v\epsilon^2} \right] \right) \right) \epsilon^{-2}, \quad (53)$$

where v and Υ are the updating rule and the updating step, respectively. Further, $0 < \epsilon < 1$ is the error tolerance with $\|\nabla f^{NARX}\| < \epsilon$, while κ and \mathcal{C} correspondingly denote

⁹Note that a “weak learner” is generally referred to here as a classifier (e.g., GoogleNet beam selector) or a regressor (e.g., NARX beamspace channel predictor).

TABLE I:
Deep learning configurations

GoogleNet Parameter	Value
TrainingSize	70%
ValidationSize	30%
MiniBatchSize	128
MaxEpochs	6
Shuffle	every epoch
InitialLearnRate	1e-3
ValidationFrequency	3
NARX Parameter	Value
TrainingSize	70%
ValidationSize	30%
Number of batches	50
No. input delay n_{θ_i}	10
No. output delay n_{θ_r}	10
No. observations	50
Observation interval	1(s)

the determinant factor and the constant of the Levenberg-Marquardt policy as defined in [42]. Finally, the computational complexities of the ensemble GoogleNet network and the ensemble NARX network can respectively be expressed as

$$\begin{aligned} \mathcal{O}_{\text{Ens}^{\text{GN}}} &= \mathcal{O}_{\text{GN}_1} + \mathcal{O}_{\text{GN}_2} + \dots + \mathcal{O}_{\text{GN}_{M_1}} \\ &= \max_j \{\mathcal{O}_{\text{GN}_j}\}, \quad \forall j \in \{1, 2, \dots, M_1\}, \end{aligned} \quad (54)$$

and

$$\begin{aligned} \mathcal{O}_{\text{Ens}^{\text{NA}}} &= \mathcal{O}_{\text{NA}_1} + \mathcal{O}_{\text{NA}_2} + \dots + \mathcal{O}_{\text{NA}_{M_1}} \\ &= \max_j \{\mathcal{O}_{\text{NA}_j}\}, \quad \forall j \in \{1, 2, \dots, M_1\}. \end{aligned} \quad (55)$$

VII. SIMULATION RESULTS

In this section, we consider a clustered THz channel with $N_{\text{cl}} = 10$ clusters and $N_{\text{ray}} = 3$ propagation rays in each cluster. The signal wavelength is $\lambda = 1.36$, the AoDs and AoAs are i.i.d. and follow a uniform distribution over $[-\frac{1}{2}, \frac{1}{2}]$, while the complex-valued gain follows $\mathcal{CN}(0,1)$. Simulations are conducted for a lens-aided MIMO system equipped with $N_r = 64, N_t = 256$. For the NARX parameters according to Table I, we set a 10-layer MLP with the number of input and output delays $n_{\theta_i} = n_{\theta_r} = 10$ over 50 observation time slots, where the samples are injected within 1-second interval. We consider 10000 samples in 50 batches, 70% of which are used for training, while the rest are used for validation and testing.

For the simulations related to the GoogleNet, we follow the configurations presented in Table I, where 70% of the sampling data are used for training and the rest are used for validation, all in a randomized manner. Further, the ‘‘Mini-BatchSize’’ corresponds to the number of images employed at each iteration of training/validation. The maximum number of training epochs is indicated by ‘‘MaxEpochs’’ and the ‘‘Shuffle’’ field is for every epoch that randomly initiates a new datastore with the same training/validation data. The initial learning rate ‘‘InitialLearnRate’’ slows down the learning in the transferred layers owing to its adopted small value and the ‘‘ValidationFrequency’’ field specifies that validation is performed every three iterations during training.

A. Performance Measures

To evaluate the performance of our proposed method, we assess the well-known error metrics, such as RMSE, as well as the normalized mean square error (NMSE) over M training samples, which correspond to

$$\text{RMSE} = \sqrt{\sum_{i=1}^M \|\mathbf{H}_b^i(t) - \widehat{\mathbf{H}}_b^i(t)\|_2^2}, \quad (56)$$

and

$$\text{NMSE} = \mathbb{E} \left\{ \frac{\sum_{i=1}^M \|\mathbf{H}_b^i(t) - \widehat{\mathbf{H}}_b^i(t)\|_2^2}{\sum_{i=1}^M \|\widehat{\mathbf{H}}_b^i(t)\|_2^2} \right\}, \quad (57)$$

where $\mathbf{H}_b^i(t)$ and $\widehat{\mathbf{H}}_b^i(t)$ are the ground truth and the predicted values of the beamspace channel for the i th sample at time step t , respectively. We also consider the standard deviation and the mean absolute deviation metrics for evaluating the channel tracking variance [43], which correspond to

$$\tau_1 = \sqrt{\frac{\sum_{i=1}^M \|\widehat{\mathbf{H}}_b^i(t) - \mu\|_2^2}{Q}}, \quad (58)$$

and

$$\tau_2 = \frac{\sum_{i=1}^M \|\widehat{\mathbf{H}}_b^i(t) - \mu\|_2}{Q}, \quad (59)$$

with the prediction average μ . The achievable SE of a hybrid analog-digital system can be expressed as (60), where $R_n = (\mathbf{W}_{\text{BB}})^H (\mathbf{S}_r)^H \mathbf{S}_r \mathbf{W}_{\text{BB}}$ is the noise covariance matrix after combining.

B. Performance Evaluation for Analog Beam Selection

The baseline analog beam selection strategies MLP, k -NN, and SVM with the same internal configurations as in [13], the conventional ReLU-driven GoogleNet, the Swish-driven GoogleNet, and the ensemble learning schemes are compared here in terms of the achievable SE¹⁰. Additionally, the fully digital zero-forcing (ZF) strategy that employs entire beams at the transceivers is the optimal benchmark baseline. Simulation results in this subsection report that the proposed analog beam selection outperforms other baselines and remains the closest to the fully digital ZF baseline.

The performance gains mainly come from the high precision of the GoogleNet and its ensemble model for analog beam selection. First, we assess the accuracy and the loss ratios for the training/validation process of the proposed Swish-driven GoogleNet scheme in Figs. 5 and 6, respectively. Clearly, the training/validation process is inaccurate in the first iterations. That is because the weights and biases of the input layer and the linear layer are not well-adjusted with

¹⁰Since our analog beam selection method comprises of three stages, the accuracy of each stage for analog beam selection should be analyzed individually. Further, possible performance gains of the overall beamspace system composed by each stage need to be assessed independently and compared to similar schemes via simulations. To this aim, the intermediate stages of our analog beam selection method, including the GoogleNet-ReLU, the GoogleNet-Swish, and the ensemble model, are evaluated numerically with separate baseline schemes.

$$SE = \log_2 \left| \mathbf{I}_{N_s} + \frac{\rho}{\sigma^2 N_s} R_n^{-1} (\mathbf{W}_{BB})^H (\mathbf{S}_r)^H \mathbf{H}_b \mathbf{S}_t \mathbf{F}_{BB} (\mathbf{F}_{BB})^H (\mathbf{S}_t)^H (\mathbf{H}_b)^H \mathbf{S}_r \mathbf{W}_{BB} \right|. \quad (60)$$

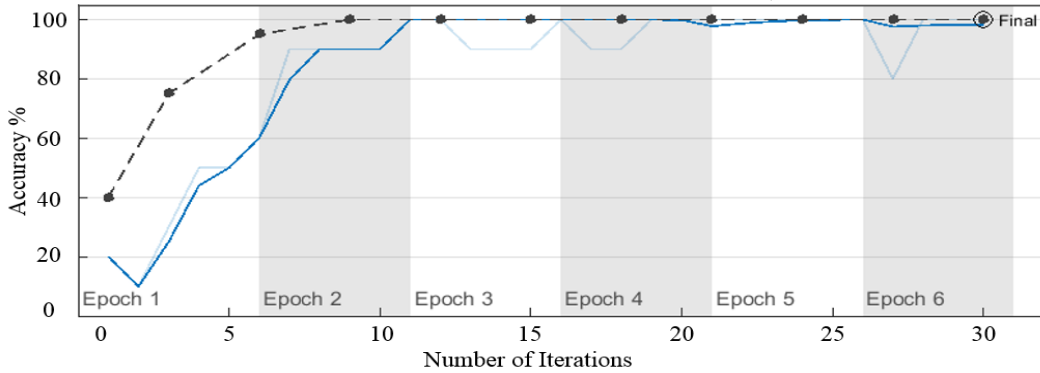


Fig. 5: Accuracy of fine-tuned GoogleNet for training and validation.

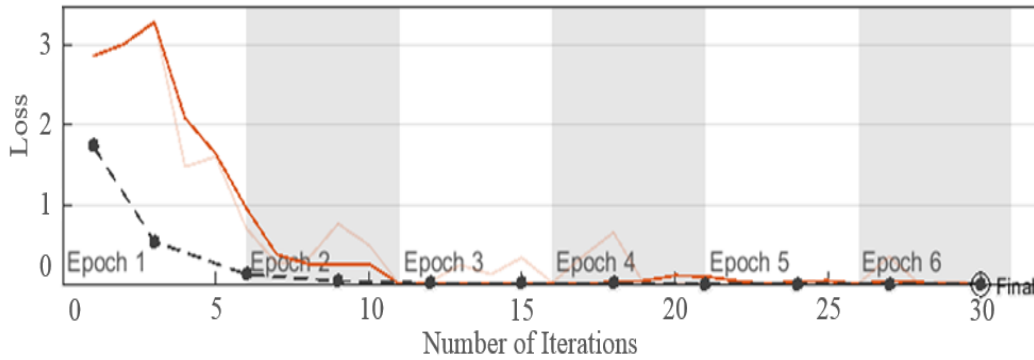


Fig. 6: Loss rate of fine-tuned GoogleNet for training and validation.

the sampling data. Gradually, as the iterations progress, the training/validation accuracy improves (tends to 100%), while the training/validation loss drops (tends to 0).

Further, we analyze the performance of our considered schemes in a comparative fashion. The benchmark fully-digital ZF strategy that uses $N_t^{\text{RF}} = 256$ and $N_r^{\text{RF}} = 16$ RF chains, as expected, has the largest achievable SE in Fig. 7(a) and Fig. 7(b) at the expense of high system complexity, energy consumption, and hardware cost. Fig. 7(a) with varying SNR in 0dB~30dB and $N_t^{\text{RF}} = N_r^{\text{RF}} = N_s$, where $N_s = 4$, indicates that under increased SNR the achievable SE improves for all the baselines as per (60). In Fig. 7(b), with varying N_s in 4~10, where $N_t^{\text{RF}} = N_r^{\text{RF}} = N_s$ and SNR = 10dB, the achievable SE increases for a higher number of simultaneous data streams. Our proposed ensemble learning scheme is superior amongst others and remains the closest to the benchmark due to its better accuracy. According to Fig. 7(a), this scheme improves the achievable SE of the MLP strategy [13] by up to 34% at SNR = 30dB. Also at SNR = 30dB, the proposed Swish-driven GoogleNet and the conventional ReLU-driven GoogleNet schemes achieve better performance with respect to other strategies (MLP, SVM, and k -NN) by exhibiting 21% and 17% achievable SE gains, respectively, as compared to the MLP option [13].

TABLE II:
GoogleNet-based analog beam selection accuracy comparison.

Architecture/Function	RMSPROP	ADAM	SGDM
GoogleNet-ReLU	83.4%	81.37%	82.22%
GoogleNet-Swish	86.21%	85.27%	86.93%

In Fig. 7(c), under the same configuration as in Fig. 7(b) with $N_s = 4$, we assess the accuracy of analog beam selection strategies. The ensemble learning strategy with 94% accuracy is the best, while the Swish-driven GoogleNet and the conventional ReLU-driven GoogleNet schemes with 86% and 83%, on average, are the second and the third best options for analog beam selection. The reason is that retraining/modifying pre-trained networks such as GoogleNet based on the transfer learning method for classification tasks (e.g., analog beam selection) is more accurate than training a deep network such as MLP [13] from scratch. Following the transfer learning principles, the parameters of a pre-trained deep structure are for the most part kept unchanged, whereas a few certain parameters are adjusted based on samples.

Considering a multi-user scenario in Fig. 7(d), we compare the performance of our proposed analog beam selection against the baseline scheme “deep reinforcement learning” (DRL) in

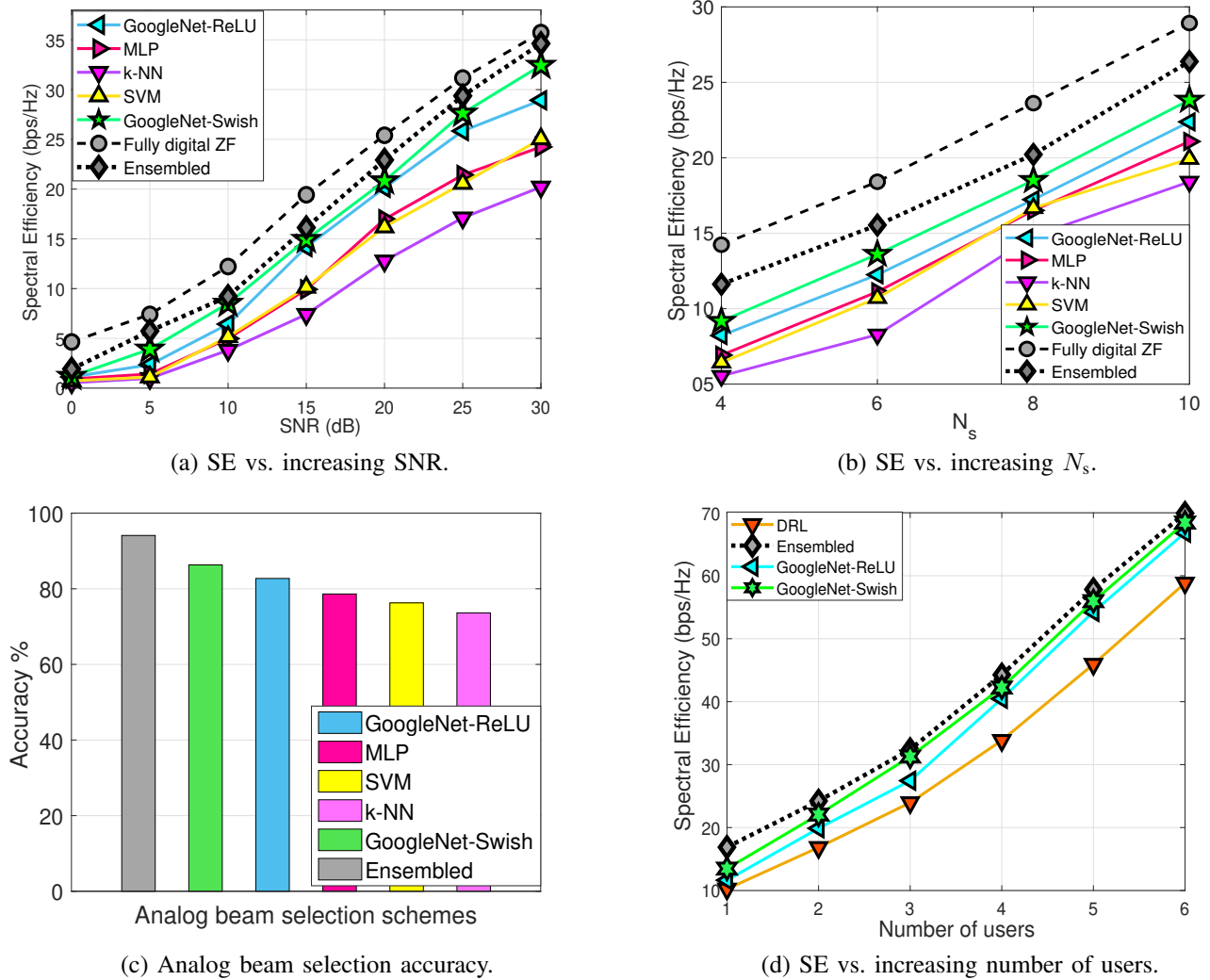


Fig. 7: Performance evaluation of beam selection strategies for increased SNR, number of selected transmit beams, and number of users.

[45]. Accordingly, for a larger number of users, a higher achievable SE for the beamspace system is observed. Further, our proposed baseline schemes (ensemble, GoogleNet-Swish, and GoogleNet-ReLU) outperform the DRL baseline option by up to 19%, 17%, and 14%, respectively. We also examine the accuracy of the conventional ReLU-driven GoogleNet as well as of the fine-tuned Swish-enabled GoogleNet schemes by applying different training functions, e.g., root mean square propagation (RMSPROP), adaptive moment estimation (ADAM), and SGDM, as demonstrated in Table II. One can observe that the Swish-driven GoogleNet scheme trained by the SGDM can achieve the highest analog beam selection accuracy.

C. Performance Evaluation for Beamspace Channel Tracking

In this subsection, we evaluate the performance of the Gaussian mixture learning-aided AMP (GM-LAMP) [8], learning-aided AMP (LAMP) [9], LSTM [11], and orthogonal matching pursuit (OMP) [19] baseline schemes for beamspace channel tracking in comparison with the proposed NARX scheme as well as the ensemble learning strategy. Here, the benchmark

scheme named full-CSI is having the ground truth information. Simulation results indicate that our proposed methods are the closest ones to the full-CSI baseline as compared to the counterparts. The performance benefits come from a more accurate tracking capability of the fine-tuned NARX and its ensemble model on the beamspace channel.

First, we assess the performance of our proposed schemes in a comparative manner. To this end, Fig. 8(a) and Fig. 8(b) contrast the achievable SE for the baseline beamspace channel tracking schemes, while increasing the SNR and the number of data streams, respectively. In Fig. 8(a), with varying SNR in 0dB~30dB and $N_t^{\text{RF}} = N_r^{\text{RF}} = N_s$, where $N_s = 4$, the ensemble learning scheme shows the best performance against other weak learner schemes, which improves the LSTM performance [11] by up to 23% for SNR = 15dB. The proposed NARX scheme closely approaches the upper bound in both figures owing to its much lower error and thus offers up to 15% gain over [11] for SNR = 15dB in Fig. 8(a). Further, as SNR grows in Fig. 8(a), all of the schemes become closer to the perfect CSI option due to their accurate predictions. In summary, the proposed schemes can tightly track the perfect

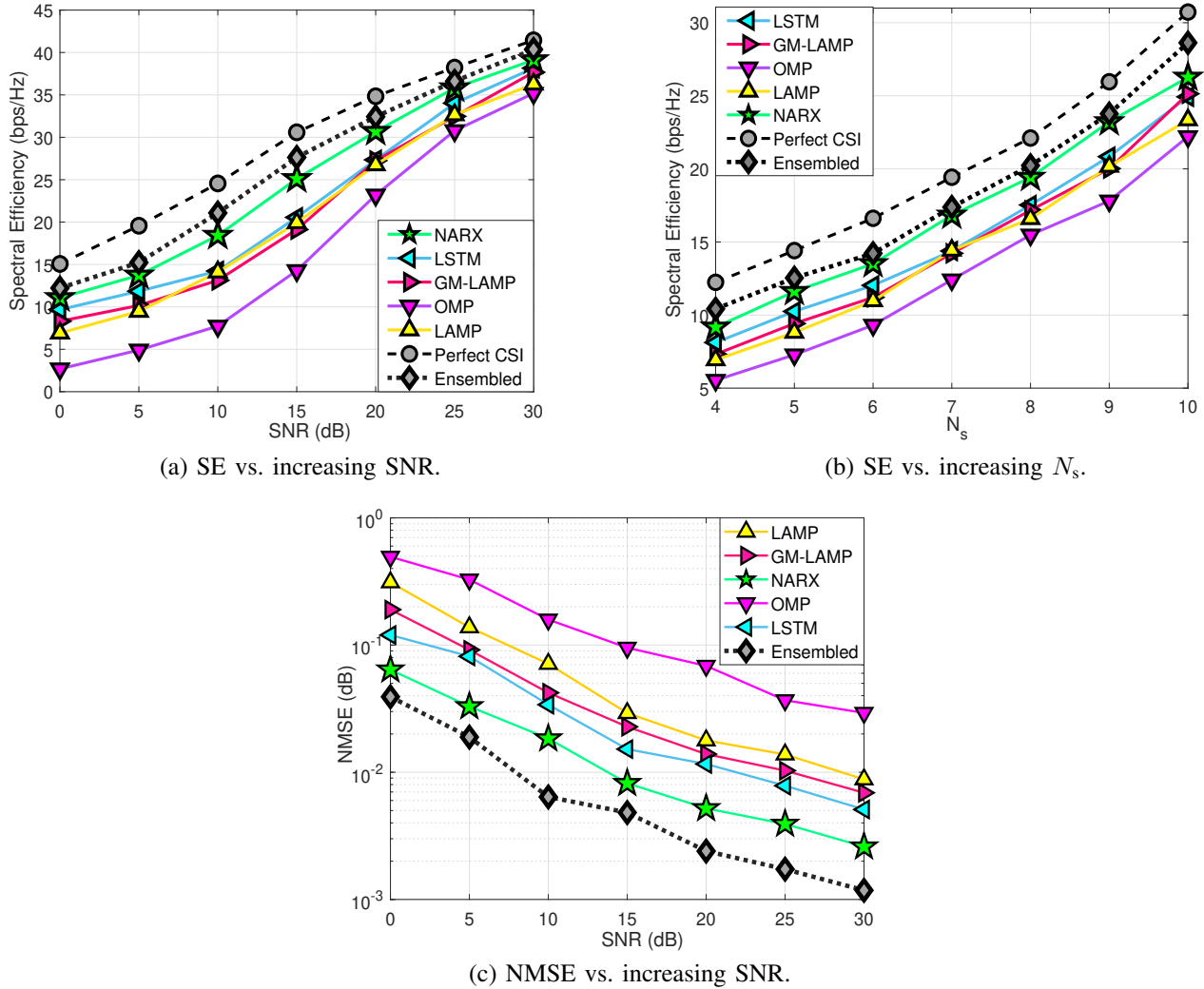


Fig. 8: Performance evaluation of beamspace channel tracking strategies for increased SNR and number of data streams.

TABLE III:
Average generalization error analysis

$\mathcal{L}(f_{\Gamma_x}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}})$	$L_{\max}^{\text{NARX}} = 2$	$L_{\max}^{\text{NARX}} = 4$	$L_{\max}^{\text{NARX}} = 6$	$L_{\max}^{\text{NARX}} = 8$	$L_{\max}^{\text{NARX}} = 10$
$\ N_{t_r}^{\text{eff}}\ = 2000$	3.828×10^{-2}	1.034×10^{-3}	9.386×10^{-3}	7.836×10^{-2}	0.037×10^{-1}
$\ N_{t_r}^{\text{eff}}\ = 4000$	8.631×10^{-3}	5.535×10^{-4}	2.957×10^{-3}	2.835×10^{-2}	4.859×10^{-2}
$\ N_{t_r}^{\text{eff}}\ = 6000$	2.094×10^{-3}	1.289×10^{-4}	7.847×10^{-4}	4.166×10^{-3}	9.946×10^{-3}
$\ N_{t_r}^{\text{eff}}\ = 8000$	5.226×10^{-4}	4.295×10^{-5}	2.384×10^{-4}	1.693×10^{-3}	3.857×10^{-3}
$\ N_{t_r}^{\text{eff}}\ = 10000$	1.639×10^{-4}	1.184×10^{-5}	9.263×10^{-5}	3.048×10^{-4}	8.583×10^{-4}

CSI at higher SNR values.

In Fig. 8(b), under varying N_s in 4~10, where $N_t^{\text{RF}} = N_r^{\text{RF}} = N_s$ and SNR = 10dB, the achievable SE increases with a higher number of simultaneous data streams, while the superiority of our proposed schemes is evident for a varying number of N_s . In Fig. 8(c), we analyze the NMSE values at varying SNR in 0dB~30dB, where $N_t^{\text{RF}} = N_r^{\text{RF}} = N_s$ and $N_s = 4$. As observed, for increased SNR, all of the schemes perform more accurately, while the ensemble learning approach shows better performance as it exploits multiple predicting modules simultaneously. Also, our proposed NARX scheme remarkably outperforms the LSTM option, especially

at lower SNR values, due to its higher time-series prediction accuracy.

Further, Fig. 9(a) confirms that training the NARX with Levenberg-Marquardt policy yields lower variance deviations in predictions, as compared to training with random weights, by up to 35% and 42% for τ_1 and τ_2 , respectively. Additionally, the ensemble learning technique reduces the prediction variance, in contrast to a single trained NARX module, by up to 47% and 52% for τ_1 and τ_2 , respectively. In Fig. 9(b), the convergence behavior of NARX during training, validation, and testing is reported. First, it is apparent that due to the lower prediction variance in Fig. 9(a), a negligible RMSE for

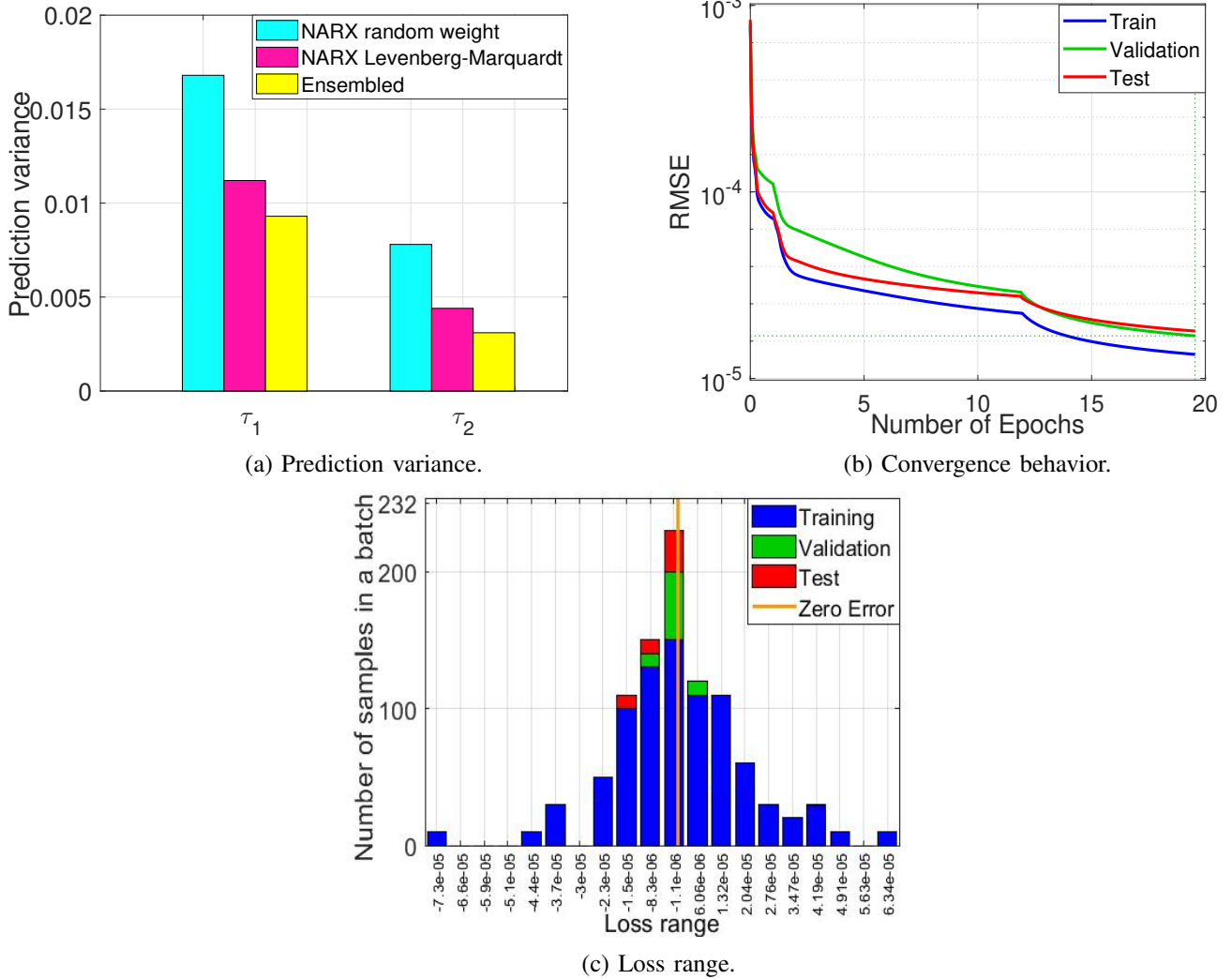


Fig. 9: Training, validation, testing, and prediction accuracy of the proposed scheme for beamspace channel tracking.

training, validation, and testing at the convergence point is observed. Importantly, appropriate initialization of the learning rate in MLP of NARX through the Bayesian optimizer offers faster convergence. Unlike the typical ANNs such as spiking neural network (SNN), which take a long time to converge (one can refer to Fig. 7 in [15]), NARX training converges within a limited number of epochs subject to adequate initialization of its hyperparameter.

In Fig. 9(c), we analyze the loss range during training, validation, and testing against the number of samples in a batch. Clearly, the higher number of samples there is in a batch, the lower errors are observed. Also, the error is much smaller during the training phase of NARX (indicated by blue lines), which is due to the availability of the ground truth samples in the open-loop structure. In the validation and testing phases, errors mostly occur close to the zero error line, which is negligible. In Table III, the generalization error of the proposed ensemble model for beamspace channel tracking is evaluated for a varying number of training samples as well as the maximum layers of the incorporated NARX models in the ensemble model. As observed, with the higher number of training samples provided, the respective generalization error

is lower while predicting the AoDs/AoAs. As an example, for 10000 training samples, the lowest errors are observed in this table. Concerning the number of NARX layers, we report the lowest error for $L_{\max}^{\text{NARX}} = 4$ as compared to other baselines, while these demonstrate higher generalization errors. Note that the number of layers as a hyperparameter in the NARX model can be optimized by the Bayesian hyperparameter optimization error.

VIII. CONCLUSIONS

This paper addresses beamspace channel tracking and analog beam selection in THz beamspace MIMO systems. First, a time-series-based deep learning approach is detailed to track the historical beamspace channel features and predict them for the upcoming time steps. Simulation results suggest that the proposed scheme, due to employing various learning-based enhancements, offers more accurate performance for the beamspace channel tracking as compared to its existing counterparts. Second, we propose to fine-tune GoogleNet being a pre-trained image classifier to learn analog beam selection as a classification task. Numerical results demonstrate that this approach adapts well to the analog beam selection

problem and achieves remarkable improvements in accuracy by contrast to prior research. This is owing to the superior classification performance of the pre-trained networks such as GoogleNet over the conventional deep learning techniques such as ConvNet.

APPENDIX A PROOF OF LEMMA 1

Note that

$$f_{\Gamma}(\theta_r(t)) = \mathbf{W}_{\text{NARX}}\theta_r(t) + \mathbf{b}_{\text{NARX}} \quad (61)$$

is an affine and, therefore, a piecewise linear function. Hence, there exists a ReLU-driven NARX estimator incorporating at most $\lceil \log_2(L_{\max}^{\text{NARX}} + 1) \rceil$ hidden layers (with L_{\max}^{NARX} indicating the size of the input layer) to represent such a function. Owing to this, f^* can be well-approximated and convergent to MMSE [28]. Therefore, for the proposed ReLU-driven NARX estimator equipped with at most $\lceil \log_2(L_{\max}^{\text{NARX}} + 1) \rceil$ hidden layers ($L_{\max}^{\text{NARX}} = N_r \times N_t$ in our case), the configuration Γ_{ε} ,

and a given precision $\varepsilon > 0$, we have $\mathbb{E}\left\{\left\|\left\|f_{\Gamma_{\varepsilon}} - f^*\right\|_2\right\|^2\right\} \leq \varepsilon$.

As defined in Section III.C, $f_{\Gamma^*}^{\text{NARX}}$ indicates the minimum MSE achieved across all the NARX configurations, e.g., Γ_{ε} . Hence, it can be observed that

$$\mathbb{E}\left\{\left\|\left\|f_{\Gamma^*}^{\text{NARX}} - f^*\right\|_2\right\|^2\right\} \leq \mathbb{E}\left\{\left\|\left\|f_{\Gamma_{\varepsilon}} - f^*\right\|_2\right\|^2\right\} \leq \varepsilon, \quad (62)$$

where the definition of the expectation $\mathbb{E}(\cdot)$ entails that

$$\mathbb{E}\left\{\left\|\left\|f_{\Gamma^*}^{\text{NARX}} - f^*\right\|_2\right\|^2\right\} \leq \mathbb{E}\{\varepsilon\} = \varepsilon. \quad (63)$$

The proof is completed.

APPENDIX B PROOF OF LEMMA 2

For finite-length $\|\Gamma^*\|_2$ and $\|\Gamma_{\chi}\|_2$, the large numbers principle [44] induces

$$\mathcal{L}_{\chi}(f_{\Gamma^*}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) \longrightarrow 0, \quad (64)$$

and

$$\mathcal{L}_{\chi}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma_{\chi}}^{\text{NARX}}) \longrightarrow 0. \quad (65)$$

Concerning the approximation error that results in $\mathcal{L}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) \geq 0$, we have

$$\begin{aligned} & \left[\mathcal{L}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) \right] - \left[\mathcal{L}_{\chi}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}_{\chi}(f_{\Gamma^*}^{\text{NARX}}) \right] \\ & \quad + \left[\mathcal{L}_{\chi}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}_{\chi}(f_{\Gamma^*}^{\text{NARX}}) \right] \geq 0. \end{aligned} \quad (66)$$

In addition, since $\mathcal{L}_{\chi}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}_{\chi}(f_{\Gamma^*}^{\text{NARX}}) \geq 0$, it can be observed that

$$\begin{aligned} & \left[\mathcal{L}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) \right] - \left[\mathcal{L}_{\chi}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}_{\chi}(f_{\Gamma^*}^{\text{NARX}}) \right] \\ & \quad + \left[\mathcal{L}_{\chi}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}_{\chi}(f_{\Gamma^*}^{\text{NARX}}) \right] \\ & \leq \left[\mathcal{L}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) \right] \\ & \quad - \left[\mathcal{L}_{\chi}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}_{\chi}(f_{\Gamma^*}^{\text{NARX}}) \right]. \end{aligned} \quad (67)$$

Accordingly, since [28]

$$\left[\mathcal{L}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) \right] - \left[\mathcal{L}_{\chi}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}_{\chi}(f_{\Gamma^*}^{\text{NARX}}) \right] \xrightarrow{\text{Pr}} 0, \quad (68)$$

the proof is completed as $\mathcal{L}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) \xrightarrow{\text{Pr}} 0$.

APPENDIX C PROOF OF COROLLARY 1

Relying on (41) and (42), one can observe that

$$\begin{aligned} \mathcal{L}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}(f^*) &= \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) + \left[\mathcal{L}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) \right] \\ & \quad + \mathbb{E}\left\{\left\|\left\|f_{\Gamma^*}^{\text{NARX}} - f^*\right\|_2\right\|^2\right\} - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) \\ &= \mathbb{E}\left\{\left\|\left\|f_{\Gamma^*}^{\text{NARX}} - f^*\right\|_2\right\|^2\right\} \\ & \quad + \mathcal{L}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}). \end{aligned} \quad (69)$$

On the one hand, the approximation error is bounded by a predetermined threshold according to Lemma 1, i.e.,

$$\mathbb{E}\left\{\left\|\left\|f_{\Gamma^*}^{\text{NARX}} - f^*\right\|_2\right\|^2\right\} \leq \varepsilon, \quad (70)$$

whereas the generalization error, on the other hand, is convergent to zero for the sufficiently large number of samples, i.e.,

$$\mathcal{L}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}(f_{\Gamma^*}^{\text{NARX}}) \xrightarrow{\text{Pr}} 0, \quad (71)$$

from Lemma 2. Therefore, it can be written that

$$\lim_{\|\chi\| \rightarrow +\infty} \Pr\left(\left[\mathcal{L}(f_{\Gamma_{\chi}}^{\text{NARX}}) - \mathcal{L}(f^*) \right] > \varepsilon\right) = 0, \quad (72)$$

which completes the proof.

ACKNOWLEDGMENTS

This paper is supported in part by the (1) Academy of Finland via: (a) Profi6 336449, (b) FIREMAN consortium n.326270 as part of CHIST-ERA-17-BDSI-003, (c) EnergyNet Fellowship n.321265/n.328869/n.352654, (d) X-SDEN project n.349965, (e) projects RADIANT, IDEA-MILL, and SOLID; and (2) Jane and Aatos Erkkö Foundation via STREAM project.

REFERENCES

- [1] H. Zarini, M. Robat Mili, M. Rasti, S. Andreev, P. H. J. Nardelli, "Swish-driven GoogleNet for analog beam selection in terahertz beamspace MIMO", *IEEE 95th Vehc. Technl. Conf. (VTC2022-Spring)*, Helsinki, Finland, 2022, pp. 1-6.
- [2] Z. Chen, et al., "A survey on terahertz communications," *China Commun.*, vol. 16, no. 2, pp. 1-35, Feb. 2019.
- [3] Y. J. Cho, G. Suk, B. Kim, D. K. Kim and C. Chae, "RF lens-embedded antenna array for THz MIMO: design and performance," *IEEE Commun. Mag.*, vol. 56, no. 7, pp. 42-48, Jul. 2018.
- [4] J. Brady, N. Behdad, and A. M. Sayeed, "Beamspace MIMO for millimeter-wave communications: system architecture, modeling, analysis and measurements," *IEEE Trans. Antennas Prop.*, vol. 61, no. 7, pp. 3814-3827, Jul. 2013.
- [5] Y. Zeng and R. Zhang, "Millimeter wave MIMO with lens antenna array: A new path division multiplexing paradigm," *IEEE Trans. Commun.*, vol. 64, no. 4, pp. 1557-1571, Apr. 2016.
- [6] X. Gao, L. Dai and A. M. Sayeed, "Low RF-complexity technologies to enable millimeter-wave MIMO with large antenna array for 5G wireless communications," *IEEE Commun. Mag.*, vol. 56, no. 4, pp. 211-217, Apr. 2018.
- [7] R. Jia, X. Chen, C. Zhong, D. W. K. Ng, H. Lin and Z. Zhang, "Design of non-orthogonal beamspace multiple access for cellular Internet-of-Things," *IEEE J. Sel. Top. Signal. Process.*, vol. 13, no. 3, pp. 538-552, Jun. 2019.
- [8] X. Wei, C. Hu, L. Dai, "Knowledge-aided deep learning for beamspace channel estimation in millimeter-wave massive MIMO systems", *arXiv preprint, arXiv:1910.12455*, Jan., 2020.
- [9] H. He, C. K. Wen, S. Jin, and G. Y. Li, "Deep learning-based channel estimation for beamspace millimeter-wave massive MIMO systems," *IEEE Commun. Lett.*, vol. 7, no. 5, pp. 852-855, Oct., 2018.
- [10] M. Borgerding, P. Schniter, and S. Rangan, "AMP-inspired deep networks for sparse linear inverse problems," *IEEE Trans. Signal Process.*, vol. 65, no. 16, pp. 4293-4308, Aug., 2017.
- [11] T. Peng, R. Zhang, X. Cheng and L. Yang, "LSTM-based channel prediction for secure massive MIMO communications under imperfect CSI," *IEEE Int. Conf. Commun. (ICC)*, Dublin, Ireland, pp. 1-6, 2020.
- [12] A. Klautau, P. Batista, N. Gonzalez-Prelcic, Y. Wang and R. W. Heath, "5G MIMO data for machine learning: application to beam selection using deep learning," *Proc. ITA*, pp. 1-9, 2018.
- [13] C. Anton-Haro and X. Mestre, "Learning and data-driven beam selection for millimeter-wave communications: an angle of arrival-based approach," *IEEE Access*, vol. 7, pp. 20404-20415, 2019.
- [14] X. Ma, Z. Chen, Z. Li, W. Chen and K. Liu, "Low complexity beam selection scheme for terahertz systems: a machine learning approach," *IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Shanghai, China, pp. 1-6, 2019.
- [15] K. Hamedani, L. Liu, S. Hu, J. Ashdown, J. Wu and Y. Yi, "Detecting dynamic attacks in smart grids using reservoir computing: A spiking delayed feedback reservoir based approach," *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 4, no. 3, pp. 253-264, June 2020.
- [16] W. Shen, X. Bu, X. Gao, C. Xing and L. Hanzo, "Beamspace precoding and beam selection for wideband millimeter-wave MIMO relying on lens antenna arrays," *IEEE Trans. Signal Process.*, vol. 67, no. 24, pp. 6301-6313, Dec., 2019.
- [17] A. A. M. Saleh and R. Valenzuela, "A statistical model for indoor multipath propagation," *IEEE J. Sel. Areas Commun.*, vol. 5, no. 2, pp. 128-137, Feb., 1987.
- [18] X. Gao, L. Dai, S. Han, C. L. I, and R. W. Heath, Jr., "Energy-efficient hybrid analog and digital precoding for mmWave MIMO systems with large antenna arrays," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 4, pp. 998-1009, Apr. 2016.
- [19] A. Alkhateeb, O. El Ayach, G. Leus, and R. W. Heath, "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE J. Sel. Top. Signal. Process.*, vol. 8, no. 5, pp. 831-846, Oct., 2014.
- [20] X. Gao, L. Dai, Y. Zhang, T. Xie, X. Dai and Z. Wang, "Fast Channel Tracking for Terahertz Beamspace Massive MIMO Systems," *IEEE Trans. Veh. Technl.*, vol. 66, no. 7, pp. 5689-5696, July 2017.
- [21] X. Gao, L. Dai, S. Han, C.-L. I, and X. Wang, "Reliable beamspace channel estimation for millimeter-wave massive MIMO systems with lens antenna array," *IEEE Trans. Wireless Commun.*, vol. 16, no. 9, pp. 6010-6021, Sep. 2017.
- [22] S.A. Billings, S. Chen, M.J. Korenberg, "Identification of MIMO nonlinear systems using a forward-regression orthogonal estimator," *Int. J. Control*, pp. 2157-2189, 1989.
- [23] S. Chen, S.A. Billings, W. Luo "Orthogonal least squares methods and their applications to nonlinear system identification", *Int. J. Control*, pp. 1873-1896, 1989.
- [24] D. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters", *J. App. Math.*, pp. 431-441, 1963.
- [25] H. Cho, Y. Kim, E. Lee, D. Choi, Y. Lee and W. Rhee, "Basic enhancement strategies when using Bayesian optimization for hyperparameter tuning of deep neural networks," *IEEE Access*, vol. 8, pp. 52588-52608, 2020.
- [26] J. Li and D. He, "A Bayesian optimization AdaBN-DCNN method with self-optimized structure and hyperparameters for domain adaptation remaining useful life prediction," *IEEE Access*, vol. 8, pp. 41482-41501, 2020.
- [27] J. G. Stoddard, J. S. Welsh and H. Hjalmarsson, "EM-based hyperparameter optimization for regularized Volterra kernel estimation," *IEEE Cont. Sys. Lett.*, vol. 1, no. 2, pp. 388-393, 2017.
- [28] H. Qiang, G. Feifei, Z. Hao, J. Shi and L. G. Ye, "Deep learning for MIMO channel estimation: interpretation, performance, and comparison", to appear in *IEEE Trans. Wireless Commun.*, available at <https://arxiv.org/abs/1911.01918>.
- [29] M. Wang, F. Gao, S. Jin and H. Lin, "An overview of enhanced massive MIMO with array signal processing techniques," *IEEE J. Sel. Top. Signal. Process.*, vol. 13, no. 5, pp. 886-901, Sept., 2019.
- [30] X. Li and A. Alkhateeb, "Deep learning for direct hybrid precoding in millimeter wave massive MIMO systems," *53rd Asilomar Conference on Signals, Systems, and Computers*, pp. 800-805, 2019.
- [31] Y. Yang, Z. Gao, Y. Ma, B. Cao and D. He, "Machine learning-enabling analog beam selection for concurrent transmissions in millimeter-wave V2V communications," *IEEE Trans. Veh. Technl.*, vol. 69, no. 8, pp. 9185-9189, 2020.
- [32] G. Celeux, S. Chretien, and F. Forbes, "A component-wise EM algorithm for mixtures," *Journal of Computational and Graphical Statistics*. no.4 pp. 697-712, Jan., 2012.
- [33] A. Alkhateeb and R. W. Heath, "Frequency selective hybrid precoding for limited feedback millimeter wave systems," *IEEE Trans. Commun.*, vol. 64, no. 5, pp. 1801-1818, May 2016.
- [34] C. Szegedy et al., "Going deeper with convolutions," *IEEE Conf. Comp. Vis. Patt. Recogn.*, Boston, pp. 1-9, 2015.
- [35] L. Balagourouchetty, J. K. Pragatheeswaran, B. Pottakkat and G. Ramkumar, "GoogLeNet-based ensemble FCNet classifier for focal liver lesion diagnosis," *IEEE J. Bio. Hlth. Inf.*, vol. 24, no. 6, pp. 1686-1694, Jun., 2020.
- [36] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, "Rethinking the inception architecture for computer vision," *IEEE Conf. Comput. Vis. Patt. Rec. (CVPR)*, Las Vegas, NV, pp. 2818-2826, 2016.
- [37] P. Ramachandran, B. Zoph, and Q. V. Lee, "Swish: A self-gated activation function." [Online]. Available: <https://arxiv.org/abs/1710.05941>.
- [38] "imresize: Resize image", mathworks.com [Online]. Available: <https://mathworks.com/help/matlab/ref/imresize.html>. [Accessed: 14-Feb.-2023].
- [39] "ind2rgb: Convert indexed image to RGB image", mathworks.com [Online]. Available: <https://mathworks.com/help/matlab/ref/ind2rgb.html>. [Accessed: 14-Feb.-2023].
- [40] Pratt, L. Y. and T. Sebastian, "Machine learning – special issue on inductive transfer", Jul., 1997.
- [41] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123-140, 1996.
- [42] E. H. Bergou, Y. Diouane and V. Kungurtsev, "Convergence and complexity analysis of a Levenberg–Marquardt algorithm for inverse problems," *Journal of Optimization Theory and Applications*, Springer Verlag, pp.927-944. April, 2020.
- [43] V. Sridhar, T. Gabillard and A. Manikas, "Spatiotemporal-MIMO channel estimator and beamformer for 5G," *IEEE Trans. Wireless Commun.*, vol. 15, no. 12, pp. 8025-8038, Dec., 2016.
- [44] W. Mendenhall, R. J. Beaver, and B. M. Beaver, *Introduction to Probability and Statistics*. Cengage Learning, 2012.
- [45] I. Ahmed, M. K. Shahid and T. Faisal, "Deep Reinforcement Learning Based Beam Selection for Hybrid Beamforming and User Grouping in Massive MIMO-NOMA System," *IEEE Access*, vol. 10, pp. 89519-89533, 2022.