# Unsupervised mmWave Beamforming via Autoencoders

Ture Peken, Ravi Tandon, and Tamal Bose
Department of Electrical and Computer Engineering
University of Arizona, Tucson, Arizona 85721
Email: {*turepeken*, *tandonr*, *tbose*}@email.arizona.edu

***Abstract*—We provide unsupervised machine learning (ML) schemes based on autoencoders for unconstrained beamforming (BF) and hybrid BF in millimeter-waves (mmWaves). An autoencoder is a powerful unsupervised ML model, and it is used to reconstruct the input with a minimal error by finding a low-dimensional representation of the input. In this paper, we present a linear autoencoder for finding the beamformers at the transmitter (Tx) and receiver (Rx), which maximize the achieved rates over the mmWave channel. Since the autoencoder has a close relationship with the singular value decomposition (SVD), we first study autoencoders for unconstrained BF based on SVD. In hybrid BF, beamformers are designed by using finite-precision phase shifters in the radio frequency (RF) domain along with power constraints. Therefore, we propose a hybrid BF algorithm based on autoencoders, which incorporates these constraints. We present our simulation results for both unconstrained BF as well as hybrid BF, and compare their performance with state-of-the-art. By using the stochastic and NYUSIM channel models, we achieve $30-40\%$ and $60-70\%$ gains in rates with the proposed autoencoder based approach compared to the supervised hybrid BF with the stochastic and NYUSIM channel models, respectively.**

## I. Introduction

Millimeter-wave (mmWave) bands (30-300 GHz), which offer high throughput and capacity by utilizing the vast amount of mmWave frequency spectrum, will play a critical role in upcoming 5G wireless systems and beyond [1]. Moreover, high BF gains can be achieved by employing a massive number of antennas (massive MIMO) at mmWaves. To keep the power consumption low and data rates high, hybrid BF, which divides BF operation into analog and digital domains, has been proposed [2]. Hybrid BF architectures can be categorized into fully-connected, sub-connected, and adaptively-connected. In the fully-connected architecture, each antenna is supported by all RF chains. Authors of [3] have proposed a hybrid BF algorithm to minimize the Euclidean distance between the unconstrained BF and hybrid BF matrices for the fully-connected structure. The authors in [4] have proposed a hybrid BF design for the sub-connected architecture, where each RF chain is connected to a set of antennas. The adaptively-connected architecture has been proposed to provide a flexible connection between RF chains and antennas [5]. The codebook-based hybrid BF has been studied in [6], where the beams are selected exhaustively based on maximum signal-to-noise-ratio (SNR). Moreover, sparsity of the mmWave channel has been exploited to reduce the computational complexity [7]. However, sparse hybrid BF methods bring overhead due to the training, which scales with the number of antennas.

There has been recent interest in applying ML algorithms to BF [8]–[10]. The authors of [8] have applied support vector machine (SVM) algorithm to select beamformers by formulating beam selection in hybrid BF as a multi-class classification problem. In [9], a supervised deep feedforward network (DFN) is used to predict the BF vectors from received signals using omni beam patterns. Authors of [10] present a supervised hybrid BF procedure for SVD and hybrid BF based on convolutional neural networks (CNNs). However, supervised approaches bring additional overhead since they require a large amount of labeled data during the training.

**Contributions of this paper:** Optimal unconstrained beamformers (which maximize channel capacity) can be found through the SVD of the channel, i.e., $k$ singular vectors corresponding to the most significant singular values of the channel matrix can be used to determine $k$ optimum beam directions [11]. Since the computational complexity of the conventional SVD algorithms [12] is $O(N^3)$ for $N$-by-$N$ matrix, our motivation is to reduce the complexity of computing the SVD as well as hybrid beamformers by leveraging the autoencoders. It has been shown that the weights of a linear autoencoder trained with a squared error loss function span the same subspace with the principal component loading vectors [13]. Moreover, a method for recovering the loading vectors from the weights of the linear autoencoder has been proposed in [14]. In this paper, motivated by the fact that optimal solution for principal component analysis (PCA), which is closely related to SVD, can be found through a linear autoencoder with squared error loss [15], we formulate unconstrained and hybrid BF design problems using linear autoencoders.

- We first provide an unsupervised scheme based on a linear autoencoder to find unconstrained beamformers.
- We then propose an unsupervised scheme for hybrid BF based on autoencoders by incorporating the RF and power constraints. We assume that the finite-precision phase shifters are used in the RF domain, which restricts the RF beamformers to have constant modulus and quantized phase values. To circumvent the non-differentiability of uniform quantization, we approximate uniform quantization using a combination of sigmoid functions with different parameters.
- By using the stochastic mmWave channel model, we compare the rates of the proposed autoencoder based algorithms

978-1-7281-5089-5/20/$31.00 ©2020 IEEE

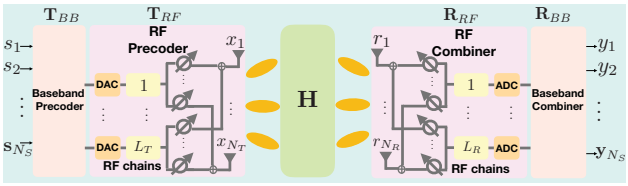



Fig. 1. Hybrid BF architecture.

for unconstrained BF and hybrid BF with the supervised hybrid BF algorithm based on CNNs [10], the supervised hybrid BF algorithm based on DFNs [9], and three conventional hybrid BF algorithms [3], [16], [17]. The results show that the proposed algorithms for unconstrained BF and hybrid BF achieve up to $65\%$ and $39\%$ gain in rates compared to the supervised hybrid BF based on DFNs [9], respectively. This is because the performance of the supervised BF method in [9] is bounded by the local optimal beamformers obtained by maximizing the rate using the received omni beam patterns while the performance of the proposed unsupervised BF approach is bounded by the global optimal beamformers found by performing the SVD of the channel matrix. Moreover, the autoencoder based hybrid BF achieves $64\%$ gain in rates compared to the DL-based algorithm [9] when the NYUSIM model is used.

## II. System Model

This section presents the system models for unconstrained and hybrid BF, and then explains the mmWave channel model.

### A. Unconstrained BF

Consider a communication system with $N_T$ transmit and $N_R$ receive antennas, which transmit $N_S$ number of data streams. We denote the channel matrix of this system by $\mathbf{H} \in \mathbb{C}^{N_R \times N_T}$, which can be written as $\mathbf{H} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^*$ through SVD. Here, $\mathbf{U} \in \mathbb{C}^{N_R \times N_R}$ and $\mathbf{V} \in \mathbb{C}^{N_T \times N_T}$ are unitary matrices, which correspond to left and right singular vectors of $\mathbf{H}$. $\mathbf{\Sigma} \in \mathbb{C}^{N_R \times N_T}$ is a diagonal matrix with non-negative singular values of $\mathbf{H}$ on its diagonal. We denote the precoder at the Tx as $\mathbf{T} \in \mathbb{C}^{N_T \times N_S}$ and the combiner at the Rx as $\mathbf{R} \in \mathbb{C}^{N_R \times N_S}$. We define the average total transmit power as $P$ and, the vector of transmitted symbols $\mathbf{s} \in \mathbb{C}^{N_S \times 1}$ satisfies $\mathbb{E}[\mathbf{s}\mathbf{s}^*] = \left(\frac{P}{N_S}\right)\mathbf{I}_{N_S}$. First, $\mathbf{s} \in \mathbb{C}^{N_S \times 1}$ is processed by $\mathbf{T}$, and then transmitted over the channel. The received signal $\mathbf{r} \in \mathbb{C}^{N_R \times 1}$ is given as $\mathbf{r} = \mathbf{H}\mathbf{T}\mathbf{s} + \mathbf{n}$. Here, $\mathbf{n} \in \mathbb{C}^{N_R \times 1} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$ is the Gaussian noise vector. Finally, the vector of received symbols $\mathbf{y} \in \mathbb{C}^{N_S \times 1}$ is obtained as $\mathbf{y} = \mathbf{R}^*\mathbf{H}\mathbf{T}\mathbf{s} + \mathbf{R}^*\mathbf{n}$. The unconstrained beamformers are found by maximizing the rate given as,

$$R = \log_2\left(\left|\mathbf{I} + \tfrac{P}{N_S}\mathbf{C}_n^{-1}\mathbf{R}_{opt}^*\mathbf{H}\mathbf{T}_{opt}\mathbf{T}_{opt}^*\mathbf{H}^*\mathbf{R}_{opt}\right|\right). \quad (1)$$

Here, $\mathbf{T}_{opt} = \mathbf{V}_{N_S}$ and $\mathbf{R}_{opt} = \mathbf{U}_{N_S}$ denote the optimal unconstrained precoder and combiner, respectively.

### B. Hybrid BF

We consider hybrid BF with fully-connected architecture, which is shown in Fig. 1. At the Tx, $N_S$ data streams, where $N_S \leq L_T \leq N_T$ and $N_S \leq L_R \leq N_R$, are processed by a baseband precoder $\mathbf{T}_{BB} \in \mathbb{C}^{L_T \times N_S}$ followed by an RF precoder $\mathbf{T}_{RF} \in \mathbb{C}^{N_T \times L_T}$. Then, the Rx processes $\mathbf{r} \in \mathbb{C}^{N_R \times 1}$ with an RF combiner $\mathbf{R}_{RF} \in \mathbb{C}^{N_R \times L_R}$ and a baseband combiner $\mathbf{R}_{BB} \in \mathbb{C}^{L_R \times N_S}$. The received signal $\mathbf{y} \in \mathbb{C}^{N_S \times 1}$ is given as,

$$\mathbf{y} = \mathbf{R}_{BB}{}^*\mathbf{R}_{RF}{}^*\mathbf{H}\mathbf{T}_{RF}\mathbf{T}_{BB}\mathbf{s} + \mathbf{R}_{BB}{}^*\mathbf{R}_{RF}{}^*\mathbf{n}. \quad (2)$$

**RF and power constraints:** Due to the usage of finite-precision phase shifters in the RF domain, $\mathbf{T}_{RF}$ and $\mathbf{R}_{RF}$ must have constant modulus, i.e., $|[\mathbf{T}_{RF}]_{i,j}|^2 = N_T^{-1}$ and $|[\mathbf{R}_{RF}]_{i,j}|^2 = N_R^{-1}$, where $|[\mathbf{T}_{RF}]_{i,j}|(|[\mathbf{R}_{RF}]_{i,j}|)$ corresponds to the magnitude of $(i,j)$th element of $\mathbf{T}_{RF}(\mathbf{R}_{RF})$. Moreover, the elements of each column in $\mathbf{T}_{RF}$ and $\mathbf{R}_{RF}$ are represented as quantized phase shifts, where each phase shifter is controlled by an $N_q$-bit input. The phase shifts of the $n(m)$th antenna of the $\mathbf{T}_{RF}(\mathbf{R}_{RF})$, can be given as $e^{\frac{j2\pi n k_q}{2^{N_q}}}\left(e^{\frac{j2\pi m k_q}{2^{N_q}}}\right)$ for $k_q = 0, 1, ..., 2^{N_q} - 1$. Due to the power constraint, $\|\mathbf{T}_{RF}\mathbf{T}_{BB}\|_F^2 = N_S$ and $\|\mathbf{R}_{RF}\mathbf{R}_{BB}\|_F^2 = N_S$. We aim to find $\mathbf{R}_{RF}$, $\mathbf{R}_{BB}$, $\mathbf{T}_{RF}$, and $\mathbf{T}_{BB}$, which maximize the achievable rate $R$ while satisfying the RF and power constraints.

### C. mmWave Channel Models

For the scope of this paper, we consider the stochastic and the measurement-based NYUSIM channel models.

**Stochastic mmWave Channel Model**: The stochastic channel model, which has been proposed in [18], is suitable to characterize the mathematical structure of mmWave channels. Therefore, we first consider this mmWave channel model, which is defined as,

$$\mathbf{H} = \sqrt{\tfrac{N_T N_R}{\rho}}\sum_{s=1}^{S} g_s \mathbf{a}_R(\theta_s)\mathbf{a}_T^*(\phi_s), \quad (3)$$

where $\rho$ is the average path-loss between the Tx and Rx, $S$ is the number of scatterers, and $g_s$ is the complex gain of the $s$th path with Rayleigh distribution, i.e., $g_s \sim \mathcal{N}(0, \overline{G})$ for $s = 1, 2, ..., S$. $\overline{G}$ is the average power gain. $\mathbf{a}_T(\phi_s)$ and $\mathbf{a}_R(\theta_s)$ denote the array response vectors at the Tx and Rx, respectively. $\phi_s \in [0, 2\pi]$ and $\theta_s \in [0, 2\pi]$ represent the $s$th path's azimuth Angle of Arrival (AoA) and Angle of Departure (AoD), respectively.

**Measurement-Based NYUSIM Model**: We consider the NYUSIM channel model, which was developed based on the measurements conducted at mmWave frequencies in New York City [19]. The NYUSIM channel model characterizes temporal and angular properties of multipath components (MPCs). This model is a clustering-based model in which a time cluster is defined by a group of rays coming from different angular directions. We refer the reader to [19] for further details.

## III. Autoencoders for Unconstrained BF

In this section, we propose an autoencoder based unsupervised scheme to design the unconstrained beamformers at the Tx and Rx. First, we show that a linear autoencoder with squared error loss can be used to compute the eigendecomposition of a matrix. We then formulate the unconstrained BF design problem with a linear autoencoder, and present our algorithm to compute the unconstrained beamformers.

### A. Autoencoders for Eigendecomposition

Fig. 2 represents a linear autoencoder, which consists of input and output layers containing $n$ nodes and a hidden layer containing $k$ nodes. Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n] \in \mathbb{C}^{n\times n}$, $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_k] \in \mathbb{C}^{n\times k}$, and $\hat{\mathbf{X}} = [\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, ..., \hat{\mathbf{x}}_n] \in \mathbb{C}^{n\times n}$ be the complex matrices formed by the input, hidden, and output nodes respectively. The output matrix $\hat{\mathbf{X}}$ is obtained from the input matrix $\mathbf{X}$ as,

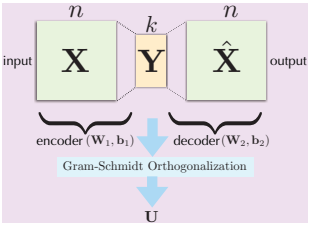



Fig. 2. Linear autoencoder.

$$\mathbf{Y} = \mathbf{X}\mathbf{W}_1 + \mathbf{b}_1 1_k{}^T, \qquad \hat{\mathbf{X}} = \mathbf{Y}\mathbf{W}_2 + \mathbf{b}_2 1_n{}^T, \tag{4}$$

where $\mathbf{W}_1 \in \mathbb{C}^{n\times k}$ and $\mathbf{b}_1 \in \mathbb{C}^{n\times 1}$ are the weight matrix and the vector of biases, which map the input to the output of the hidden layer. $\mathbf{W}_2 \in \mathbb{C}^{k\times n}$ and $\mathbf{b}_2 \in \mathbb{C}^{n\times 1}$ are the weight matrix and the bias vector of the second layer (hidden to output). The objective is to find the optimal weight matrices and bias vectors, which minimize the following cost function,

$$\min_{\mathbf{W}_1,\mathbf{b}_1,\mathbf{W}_2,\mathbf{b}_2} \left|\left|\mathbf{X} - \left(\left(\mathbf{A}_i\mathbf{W}_1 + \mathbf{b}_1 1_k{}^T\right)\mathbf{W}_2 + \mathbf{b}_2 1_n{}^T\right)\right|\right|^2. \tag{5}$$

For fixed $\mathbf{W}_1$, $\mathbf{W}_2$, and $\mathbf{b}_1$, the minimization of the cost function w.r.t. $\mathbf{b}_2$ yields,

$$\mathbf{b}_2 = \tfrac{1}{n}\left(\mathbf{X} - \left(\mathbf{X}\mathbf{W}_1 + \mathbf{b}_1 1_k{}^T\right)\mathbf{W}_2\right) 1_n. \tag{6}$$

By denoting $\mathbf{Y} = \mathbf{X}\mathbf{W}_1 + \mathbf{b}_1 1_k{}^T$, (5) can be rewritten as,

$$\min_{\mathbf{Y},\mathbf{W}_2} \left|\left|\left(\mathbf{X} - \mathbf{Y}\mathbf{W}_2\right) - \frac{1}{n}\left(\mathbf{X} - \mathbf{Y}\mathbf{W}_2\right) 1_n 1_n{}^T\right|\right|^2. \tag{7}$$

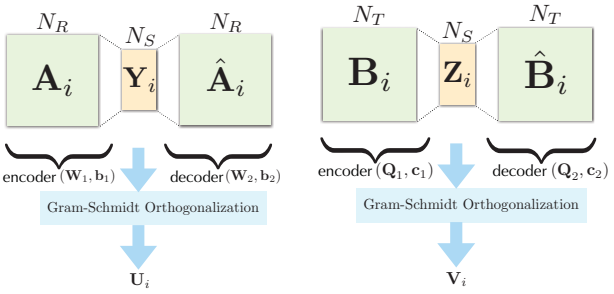
Let us assume $\mathbf{X}$ is a rank-$k$ matrix. In this case, (7) is minimized in terms of Euclidean norm if $\mathbf{Y}\mathbf{W}_2$ is the best rank-$k$ approximation of $\mathbf{X}$. We also assume that $\mathbf{X}$ is a diagonalizable matrix, then it can be written by using eigendecomposition as $\mathbf{X} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^*$, where $\mathbf{U} \in \mathbb{C}^{n\times n}$ is the unitary matrix whose columns are the eigenvectors of $\mathbf{X}$, and $\mathbf{\Lambda}$ is the diagonal matrix whose diagonal elements are the eigenvalues of $\mathbf{X}$. Then, the optimum weight matrix $\mathbf{W}_2$ and the optimum output of the encoder $\mathbf{Y}$ must satisfy $\mathbf{Y}\mathbf{W}_2 = [\mathbf{U}]_{.,\leq k}\,[\mathbf{\Lambda}]_{k,k}\,[\mathbf{U}^*]_{.,\leq k}$, where $[\mathbf{\Lambda}]_{k,k} = \mathsf{diag}(\lambda_1, ..., \lambda_k)$ and $[\mathbf{U}]_{.,\leq k}$ is formed by first $k$ columns of $\mathbf{U}$. Consequently,

$$\mathbf{W}_2 = \mathbf{T}\,[\mathbf{U}^*]_{.,\leq k}\,, \qquad \mathbf{Y} = [\mathbf{U}]_{.,\leq k}\,[\mathbf{\Lambda}]_{k,k}\,\mathbf{T}^{-1}, \tag{8}$$

where $\mathbf{T}$ is an arbitrary $k \times k$ nonsingular matrix. The eigenvectors matrix $[\mathbf{U}]_{.,\leq k}$ of $\mathbf{X}$ can be estimated by applying Gram-Schmidt orthogonalization to the output of the encoder $\mathbf{Y}$. Moreover, the columns of $[\mathbf{U}]_{.,\leq k}$ must form a set of orthonormal vectors, which implies that $||\mathbf{u}_i^*\mathbf{u}_i||_2 = 1$ and $||\mathbf{u}_i^*\mathbf{u}_j||_2 = 0\ \forall\ i,j$ s.t. $i \neq j$. Therefore, we define the cost function to estimate the eigenvectors of $\mathbf{X}$ as,

$$J = ||\mathbf{X} - \hat{\mathbf{X}}||^2 + \lambda \textstyle\sum_{i\neq j} ||\mathbf{u}_i^*\mathbf{u}_j||_2, \tag{9}$$

where $\lambda$ is the non-negative constant of the penalty term. The above discussion indicates that linear autoencoders with a squared error loss function essentially are equivalent to performing the eigendecomposition of a matrix.




(a) Autoencoder for the combiner. (b) Autoencoder for the precoder.

Fig. 3. Autoencoders for the unconstrained beamformers.

### B. Problem Formulation

We formulate the problem of unconstrained BF design by using a linear autoencoder based on eigendecomposition. We receive $m$ channel matrices $(\{\mathbf{H}_i\}_{i=1}^m)$, and train the linear autoencoder to estimate the unconstrained beamformers of the given channel matrix. Let us define $i$th matrix in the training set as $\mathbf{H}_i \in \mathbb{C}^{N_R\times N_T}$, which can be decomposed through SVD as $\mathbf{H}_i = \mathbf{U}_i\mathbf{\Sigma}_i\mathbf{V}_i^*$. We assume that $N_S \leq \mathsf{rank}(\mathbf{H}_i) = k$, where $N_S$ is the number of data streams. We denote $\mathbf{A}_i \in \mathbb{C}^{N_R\times N_R}$ and $\mathbf{B}_i \in \mathbb{C}^{N_T\times N_T}$ as $\mathbf{A}_i = \mathbf{H}_i\mathbf{H}_i^*$ and $\mathbf{B}_i = \mathbf{H}_i^*\mathbf{H}_i$, respectively. Through eigendecomposition, $\mathbf{A}_i = \mathbf{U}_i\mathbf{\Sigma}_i^2\mathbf{U}_i^*$ and $\mathbf{B}_i = \mathbf{V}_i\mathbf{\Sigma}_i^2\mathbf{V}_i^*$. Here, $\mathbf{\Sigma}_i^2$ is a diagonal matrix, which consists of the square of the singular values of $\mathbf{H}_i$. Therefore, we can use linear autoencoders to solve the eigendecomposition of $\mathbf{A}_i$ and $\mathbf{B}_i$, whose eigenvectors correspond to the unconstrained combiner $(\mathbf{U}_i)$ and the unconstrained precoder $(\mathbf{V}_i)$ of $\mathbf{H}_i$, respectively.

To estimate the unconstrained combiner based on the given channel matrix, we propose to use the linear autoencoder shown in Fig. 3-a. In the $i$th iteration of the training, the input matrix $\mathbf{A}_i$ passes through the hidden layer of the autoencoder, which generates $\mathbf{Y}_i \in \mathbb{C}^{N_R\times N_S}$ according to mapping $\mathbf{Y}_i = \mathbf{A}_i\mathbf{W}_1 + \mathbf{b}_1 1_{N_S}{}^T$. Then, the second layer maps $\mathbf{Y}_i$ to $\hat{\mathbf{A}}_i \in \mathbb{C}^{N_R\times N_R}$ such that $\hat{\mathbf{A}}_i = \mathbf{Y}_i\mathbf{W}_2 + \mathbf{b}_2 1_{N_R}{}^T$. Similarly, we propose to use the linear autoencoder given in Fig. 3-b to estimate the unconstrained precoder of the given matrix. In the $i$th iteration of the training, the input matrix $\mathbf{B}_i$ passes through the hidden layer, which generates $\mathbf{Z}_i \in \mathbb{C}^{N_T\times N_S}$ such that $\mathbf{Z}_i = \mathbf{B}_i\mathbf{Q}_1 + \mathbf{c}_1 1_{N_S}{}^T$. Then, the second layer maps $\mathbf{Z}_i$ to $\hat{\mathbf{B}}_i \in \mathbb{C}^{N_T\times N_T}$ according to $\hat{\mathbf{B}}_i = \mathbf{Z}_i\mathbf{Q}_2 + \mathbf{c}_2 1_{N_T}{}^T$.

### C. Loss Function of the Autoencoder for Unconstrained BF

The autoencoder given in Fig. 3-a is trained on the dataset $\{\mathbf{A}_i\}_{i=1}^m$ based on the cost function below,

$$\min_{\mathbf{W}_1,\mathbf{b}_1,\mathbf{W}_2,\mathbf{b}_2} \frac{1}{m}\sum_{i=1}^{m} ||\mathbf{A}_i - \hat{\mathbf{A}}_i||^2 + \lambda \sum_{i\neq j} ||\mathbf{u}_i^*\mathbf{u}_j||_2, \tag{10}$$

where $\hat{\mathbf{A}}_i = \left(\mathbf{A}_i\mathbf{W}_1 + \mathbf{b}_1 1_{N_S}{}^T\right)\mathbf{W}_2 + \mathbf{b}_2 1_{N_R}{}^T$ and $\lambda$ is the non-negative constant of the penalty term. By following steps similar to those as in (8), the optimum $\mathbf{W}_2$ and $\mathbf{Y}_i$ are found as,

$$\mathbf{W}_2 = \mathbf{T}_i\,[\mathbf{U}_i^*]_{.,\leq N_S}\,, \qquad \mathbf{Y}_i = [\mathbf{U}_i]_{.,\leq N_S}\,\left[\mathbf{\Sigma}_i^2\right]_{N_S,N_S}\,\mathbf{T}_i^{-1}, \tag{11}$$

where $\mathbf{T}_i$ is an arbitrary $N_S \times N_S$ nonsingular matrix. In the $i$th iteration during the training, we apply Gram-Schmidt

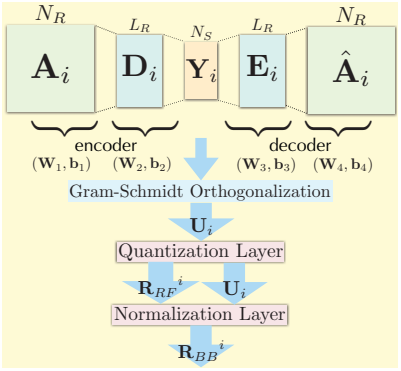



Fig. 4. Autoencoder for estimating hybrid beamformers.

orthogonalization to $\mathbf{Y}_i$ as shown in Fig. 3-a to estimate the unconstrained combiner $\mathbf{U}_i$. To satisfy the orthonormality of $\mathbf{U}_i$, the penalty term is added to (10). We use the autoencoder, which is shown in Fig. 3-b, to estimate the unconstrained precoder at the Tx. The cost function to train the autoencoder on the dataset $\{\mathbf{B}_i\}_{i=1}^m$ can be given similarly as,

$$\min_{\mathbf{Q}_1,\mathbf{c}_1,\mathbf{Q}_2,\mathbf{c}_2} \frac{1}{m}\sum_{i=1}^{m} ||\mathbf{B}_i - \hat{\mathbf{B}}_i||^2 + \lambda \sum_{i\neq j} ||\mathbf{v}_i^* \mathbf{v}_j||_2, \tag{12}$$

where $\hat{\mathbf{B}}_i = \left(\mathbf{B}_i\mathbf{Q}_1 + \mathbf{c}_1 1_{N_S}{}^T\right)\mathbf{Q}_2 + \mathbf{c}_2 1_{N_T}{}^T$ and $\lambda$ is the non-negative constant of the penalty term. Similar to (11), we calculate the optimum $\mathbf{Q}_2$ and $\mathbf{Z}_i$ as,

$$\mathbf{Q}_2 = \mathbf{S}_i \left[\mathbf{V}_i^*\right]_{.,\leq N_S}, \qquad \mathbf{Z}_i = \left[\mathbf{V}_i\right]_{.,\leq N_S} \left[\mathbf{\Sigma}_i^2\right]_{N_S,N_S} \mathbf{S}_i^{-1}, \tag{13}$$

where $\mathbf{S}_i$ is an arbitrary $N_S \times N_S$ nonsingular matrix. We then use Gram-Schmidt orthogonalization to estimate the unconstrained precoder $\mathbf{V}_i$. To satisfy the orthonormality of $\mathbf{V}_i$, the penalty term is added to (12).

### D. Algorithm for Unconstrained BF via Autoencoder

The proposed autoencoder based scheme for computing the unconstrained beamformers is shown in Algorithm 1. In the training phase, from step 2 to step 5, the autoencoder given in Fig. 3-a is trained using the cost function defined in (10). In the $i$th iteration, $\mathbf{Y}_i = \left[\mathbf{U}_i\right]_{.,\leq N_S} \left[\mathbf{\Sigma}_i^2\right]_{N_S,N_S} \mathbf{T}_i^{-1}$ is orthogonalized to estimate the unconstrained combiner $\mathbf{R}_{opt}^i = \left[\mathbf{U}_i\right]_{.,\leq N_S}$ for $i$th matrix in the training set. From step 7 to step 10, the autoencoder given in Fig. 3-b is trained using the cost function defined in (12). The output of the encoder $\mathbf{Z}_i = \left[\mathbf{V}_i\right]_{.,\leq N_S} \left[\mathbf{\Sigma}_i^2\right]_{N_S,N_S} \mathbf{S}_i^{-1}$ in the $i$th iteration is orthogonalized to estimate the unconstrained precoder $\mathbf{T}_{opt}^i = \left[\mathbf{V}_i\right]_{.,\leq N_S}$. In the test phase, $\mathbf{R}_{opt}^j = \left[\mathbf{U}_j\right]_{.,\leq N_S}$ and $\mathbf{T}_{opt}^j = \left[\mathbf{V}_j\right]_{.,\leq N_S}$ are generated using the trained autencoders for $j = 1, ..., n$.

## IV. Autoencoders for Hybrid BF

In this section, we introduce an unsupervised approach using autoencoders for computing the hybrid beamformers of the system given in Fig. 1. We seek to design hybrid combiners $(\mathbf{R}_{RF}, \mathbf{R}_{BB})$ and precoders $(\mathbf{T}_{RF}, \mathbf{T}_{BB})$, which maximize the rate $R$ given in (1), by satisfying the RF and power constraints defined in Section II-B.

**Algorithm 1** Unconstrained BF via Autoencoder

**Training Phase:**

**Input:** Training dataset $\{\mathbf{H}_i\}_{i=1}^m$
**Output:** Trained autoencoder models
1: Compute $\mathbf{A}_i = \mathbf{H}_i\mathbf{H}_i^*$
2: **for** $i \leftarrow 1$ to $m$ **do**
3: Train autoencoder given in Fig. 3-a with $\mathbf{A}_i$ based on (10)
4: Apply Gram-Schmidt orthogonalization to the output of the encoder $\mathbf{Y}_i$ to estimate $\mathbf{R}_{opt}^i = \left[\mathbf{U}_i\right]_{.,\leq N_S}$
5: **end for**
6: Compute $\mathbf{B}_i = \mathbf{H}_i^*\mathbf{H}_i$
7: **for** $i \leftarrow 1$ to $m$ **do**
8: Train autoencoder given in Fig. 3-b with $\mathbf{B}_i$ based on (12)
9: Apply Gram-Schmidt orthogonalization to the output of the encoder $\mathbf{Z}_i$ to estimate $\mathbf{T}_{opt}^i = \left[\mathbf{V}_i\right]_{.,\leq N_S}$
10: **end for**

**Test Phase:**

**Input:** Test dataset $\{\mathbf{H}_j\}_{j=1}^n$
**Output:** Unconstrained combiner $\mathbf{R}_{opt}^j$ and precoder $\mathbf{T}_{opt}^j$, $j = 1, ..., n$
1: Compute $\mathbf{A}_j = \mathbf{H}_j\mathbf{H}_j^*$ and $\mathbf{B}_j = \mathbf{H}_j^*\mathbf{H}_j$
2: Obtain $\mathbf{R}_{opt}^j$ and $\mathbf{T}_{opt}^j$ with the trained autoencoders

### A. Problem Formulation

We propose a linear autoencoder shown in Fig. 4 to predict the RF and baseband combiners at the Rx. In the training phase, we use $\mathbf{A}_i \in \mathbb{C}^{N_R \times N_R}$, $i = 1, ..., m$, as the input data. The first hidden layer generates $\mathbf{D}_i \in \mathbb{C}^{N_R \times L_R}$ according to $\mathbf{D}_i = \mathbf{A}_i\mathbf{W}_1 + \mathbf{b}_1 1_{L_R}^T$. Then, the output of the encoder $\mathbf{Y}_i \in \mathbb{C}^{N_R \times N_S}$ is generated as $\mathbf{Y}_i = \mathbf{D}_i\mathbf{W}_2 + \mathbf{b}_2 1_{N_S}^T$. The third layer maps $\mathbf{Y}_i$ to $\mathbf{E}_i \in \mathbb{C}^{N_R \times L_R}$ such that $\mathbf{E}_i = \mathbf{Y}_i\mathbf{W}_3 + \mathbf{b}_3 1_{L_R}^T$. Finally, the last layer maps $\mathbf{E}_i$ to $\hat{\mathbf{A}}_i \in \mathbb{C}^{N_R \times N_R}$ according to $\hat{\mathbf{A}}_i = \mathbf{E}_i\mathbf{W}_4 + \mathbf{b}_4 1_{N_R}^T$. A linear autoencoder, which has a similar structure with Fig. 4, has been proposed to predict the RF and baseband precoders at the Tx. We use $\mathbf{B}_i \in \mathbb{C}^{N_T \times N_T}$, $i = 1, ..., m$, as the input data during the training. The proposed autoencoder is trained to estimate $\hat{\mathbf{B}}_i$, where $\hat{\mathbf{B}}_i = ((((\mathbf{B}_i\mathbf{Q}_1 + \mathbf{c}_1 1_{L_T}^T)\mathbf{Q}_2) + \mathbf{c}_2 1_{N_S}^T)\mathbf{Q}_3 + \mathbf{c}_3 1_{L_T}^T)\mathbf{Q}_4 + \mathbf{c}_4 1_{N_T}^T$.

### B. Loss Function of the Autoencoder for Hybrid BF

The linear autoencoder shown in Fig. 4 is trained on the dataset $\{\mathbf{A}_i\}_{i=1}^m$, based on the cost function below,

$$\min_{\mathbf{W}_1,\mathbf{b}_1,\mathbf{W}_2,\mathbf{b}_2,\mathbf{W}_3,\mathbf{b}_3,\mathbf{W}_4,\mathbf{b}_4} \frac{1}{m}\sum_{i=1}^{m} ||\mathbf{A}_i - \hat{\mathbf{A}}_i||^2 + \lambda \sum_{i\neq j} ||\mathbf{u}_i^* \mathbf{u}_j||_2, \tag{14}$$

where $\hat{\mathbf{A}}_i = ((((\mathbf{A}_i\mathbf{W}_1 + \mathbf{b}_1 1_{L_R}^T)\mathbf{W}_2) + \mathbf{b}_2 1_{N_S}^T)\mathbf{W}_3 + \mathbf{b}_3 1_{L_R}^T)\mathbf{W}_4 + \mathbf{b}_4 1_{N_R}^T$ and $\lambda$ is the non-negative constant of the penalty term. Here, $\mathbf{u}_i$ denotes the $i$th column of $\mathbf{U}_i = \mathbf{R}_{RF}{}^i\mathbf{R}_{BB}{}^i$.

Using steps similar to those used in (8), the optimum values for $\mathbf{Y}_i$, $\mathbf{W}_3$, and $\mathbf{W}_4$ can be determined as,

$$\begin{aligned} \mathbf{Y}_i &= \left[\mathbf{U}_i\right]_{.,\leq N_S} \left[\mathbf{\Sigma}_i^2\right]_{N_S,N_S} \mathbf{L}_i^{-1}, \\ \mathbf{W}_3 &= \mathbf{L}_i \left(\mathbf{R}_{BB}{}^i\right)^* \mathbf{T}_i^{-1}, \quad \mathbf{W}_4 = \mathbf{T}_i \left(\mathbf{R}_{RF}{}^i\right)^*, \end{aligned} \tag{15}$$

where $\mathbf{L}_i$ and $\mathbf{T}_i$ are arbitrary $N_S \times N_S$ and $L_R \times L_R$ nonsingular matrices, respectively. Gram-Schmidt orthogonalization is applied to $\mathbf{Y}_i$ to estimate the unconstrained combiner $\mathbf{U}_i$. Then, $\mathbf{U}_i$ is sent through a quantization layer, which estimates $\mathbf{R}_{RF}{}^i$. The quantization layer begins with

a fully connected neural network, followed by a quantization operation. Finally, normalization layer estimates $\mathbf{R}_{BB}{}^{i}$ by using $\mathbf{R}_{RF}{}^{i}$ and $\mathbf{U}_i$. We then train a linear autoencoder, which is similar to Fig. 4, to estimate the RF and baseband precoders $(\mathbf{T}_{RF}, \mathbf{T}_{BB})$. We use the dataset $\{\mathbf{B}_i\}_{i=1}^{m}$ to train the autoencoder by using the cost function in (12), where $\hat{\mathbf{B}}_i = ((((\mathbf{B}_i\mathbf{Q}_1 + \mathbf{c}_1 1_{L_T}^T)\mathbf{Q}_2) + \mathbf{c}_2 1_{N_S}^T)\mathbf{Q}_3 + \mathbf{c}_3 1_{L_T}^T)\mathbf{Q}_4 + \mathbf{c}_4 1_{N_T}^T$. By following the similar procedures used to estimate $\mathbf{R}_{RF}{}^{i}$ and $\mathbf{R}_{BB}{}^{i}$, we can obtain estimated values of $\mathbf{T}_{RF}{}^{i}$ and $\mathbf{T}_{BB}{}^{i}$.

### C. Incorporation of RF and Power Constraints

As finite-precision phase shifters are used in the RF domain, the phase of each element in RF combiner and precoder matrices must be a quantized value. However, gradient-based optimization techniques generate zero gradients if uniform quantization is used, which would prevent weight updates of autoencoder during training. To overcome this, we replace each step function in the uniform quantization with a sigmoid function, which was used in [10]. Moreover, the power constraint of the hybrid BF system must be satisfied. Therefore, normalization layers are included to estimate $\mathbf{R}_{BB}$ and $\mathbf{T}_{BB}$.

### D. Algorithm for Hybrid BF via Autoencoder

Leveraging the above ideas, the autoencoder based unsupervised scheme for hybrid BF is summarized in Algorithm 2. In the training phase, from step 2 to step 8, the autoencoder given in Fig. 4 is trained by using the cost function in (14). In the $i$th iteration, $\mathbf{Y}_i$ is orthogonalized to estimate $\mathbf{U}_i$. Then, $\mathbf{R}_{RF}{}^{i}$ is estimated through the quantization layer. $\mathbf{R}_{BB}{}^{i}$ is estimated by using $\mathbf{R}_{RF}{}^{i}$ and $\mathbf{U}_i$, and normalized to satisfy the power constraint. Similar calculations are applied during the training phase of the linear autoencoder for the hybrid precoders to estimate $\mathbf{T}_{RF}{}^{i}$ and $\mathbf{T}_{BB}{}^{i}$. In the test phase, trained autoencoders predict $\mathbf{R}_{RF}{}^{j}$, $\mathbf{R}_{BB}{}^{j}$, $\mathbf{T}_{RF}{}^{j}$, and $\mathbf{T}_{BB}{}^{j}$ for $j = 1, ..., n$.

**Algorithm 2** Hybrid BF via Autoencoder

**Training Phase:**

**Input:** Training dataset $\{\mathbf{H}_i\}_{i=1}^{m}$
**Output:** Trained autoencoder models
1: Compute $\mathbf{A}_i = \mathbf{H}_i\mathbf{H}_i{}^{*}$
2: **for** $i \leftarrow 1$ to $m$ **do**
3: With $\mathbf{A}_i$ train the autoencoder in Fig. 4-a based on (14)
4: Apply Gram-Schmidt orthogonalization to $\mathbf{Y}_i$ to estimate $\mathbf{U}_i$
5: Process $\mathbf{U}_i$ through quantization layer to estimate $\mathbf{R}_{RF}{}^{i}$
6: Compute $\mathbf{R}_{BB}{}^{i} = ((\mathbf{R}_{RF}{}^{i})^{*}\mathbf{R}_{RF}{}^{i})^{-1}(\mathbf{R}_{RF}{}^{i})^{*}\mathbf{U}_i$
7: Normalize $\mathbf{R}_{BB}{}^{i}$ as $\mathbf{R}_{BB}{}^{i} = \sqrt{N_S}\frac{\mathbf{R}_{BB}{}^{i}}{\|\mathbf{R}_{RF}{}^{i}\mathbf{R}_{BB}{}^{i}\|_F}$
8: **end for**
9: Computations similar to steps $1-8$ are applied during the training of the autoencoder for estimating $\mathbf{T}_{RF}{}^{i}$ and $\mathbf{T}_{BB}{}^{i}$

**Test Phase:**

**Input:** Test dataset $\{\mathbf{H}_j\}_{j=1}^{n}$
**Output:** $\mathbf{R}_{RF}{}^{j}$, $\mathbf{R}_{BB}{}^{j}$, $\mathbf{T}_{RF}{}^{j}$, and $\mathbf{T}_{BB}{}^{j}$, $j = 1, ..., n$
1: Compute $\mathbf{A}_j = \mathbf{H}_j\mathbf{H}_j{}^{*}$ and $\mathbf{B}_j = \mathbf{H}_j{}^{*}\mathbf{H}_j$
2: Obtain $\mathbf{R}_{RF}{}^{j}$, $\mathbf{R}_{BB}{}^{j}$, $\mathbf{T}_{RF}{}^{j}$, and $\mathbf{T}_{BB}{}^{j}$ with the trained autoencoders

## V. Simulation Results

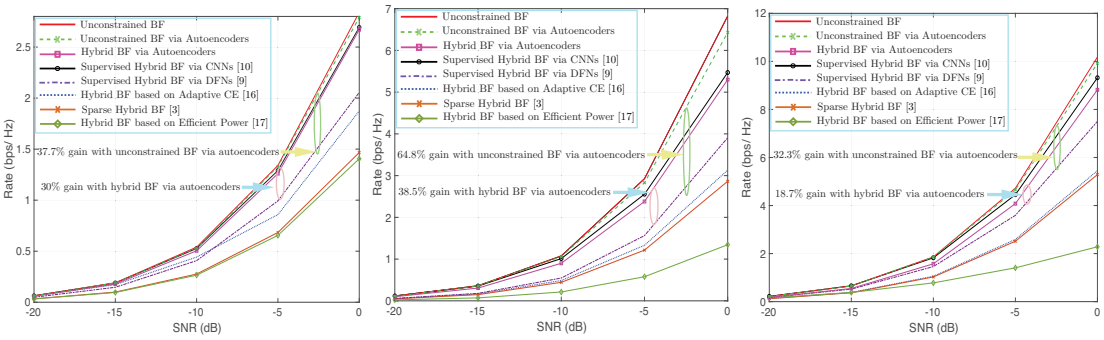
This section compares the rates of the proposed approaches with state-of-the-art by using Tensorflow based simulations.

**Autoencoder Model:** The loss functions given in (10) and (12) with Adam optimization are used for training autoencoders shown in Fig. 3-a and 3-b, respectively. We train the autoencoders for estimating the hybrid beamformers using the cost functions defined in Section IV-B with Adam optimization. The mini-batch size and $\lambda$ of the penalty term are set to 32 and $0.05$, respectively. The number of training iterations is $10000$. We set the learning rate to $0.0001$, $0.0002$, and $0.002$ for 4-by-4, 8-by-8, and 16-by-16 mmWave systems, respectively. For the stochastic model, we use $50000$ channel matrices, which are divided into a training set with $40000$ and a test set with $10000$ samples. For the NYUSIM model, we use a training set with $8000$ and a test set with $2000$ matrices.

**Channel Model and Data Generation:** In the stochastic model, we use uniform linear arrays (ULAs). The AoDs/AoAs are uniformly distributed in $[0, 2\pi]$. We set the path loss exponent (PLE) to 3 and the bandwidth to $100$ MHz. For NYUSIM model, we consider uniform rectangular arrays (URAs), which consist of $4$ rows and $4$ columns resulting in a total of $16$ antenna elements. The PLE is $2$ and the bandwidth is $850$ MHz. In both models, the distance between the Tx and Rx is $50$ m, the carrier frequency is $28$ GHz, and the transmit power is 7 dB. We use 1-bit, 2-bit, and 3-bit phase shifters in 4-by-4, 8-by-8, and 16-by-16 mmWave systems, respectively.

**Autoencoder based BF versus State-of-the-Art:** We first conduct simulations using the stochastic mmWave channel model. It is shown in Fig. 5-a that we obtain $37.7\%$ gain with the unconstrained BF via autoencoder compared to the supervised BF [9] for 4-by-4 mmWave system. We observe that $30\%$ gain in rates is achieved with the hybrid BF via autoencoder compared to the algorithm based on DFNs [9]. Fig. 5-b shows that we achieve $64.8\%$ gain in rates with the unconstrained BF via autoencoder compared to the algorithm based on DFNs for 8-by-8 system. Furthermore, $17.7\%$ improvement in rates is obtained with the unconstrained BF via autoencoder compared to the supervised BF based on CNNs. Fig. 5-b shows that $38.5\%$ improvement is achieved with our approach compared to [9]. We observe in Fig. 5-c that $32.3\%$ gain is obtained with the unconstrained BF via autoencoder compared to the algorithm based on DFNs for 16-by-16 system. Finally, an improvement of $18.7\%$ is achieved with the autoencoder based hybrid BF compared to the algorithm based on DFNs. The proposed autoencoder based approaches and the other DL-based BF algorithms [9], [10] work better than the conventional BF approaches. The DL-based algorithms find the global optimum as long as they are trained with lots of data with appropriate hyperparameters.

**Adaptability to Other Scenarios:** We extend the simulations from $-20$ to 20 dB SNR using the NYUSIM model. Fig. 6 shows the achieved rates of the hybrid BF via autoencoder and the state-of-the-art algorithms when the NYUSIM model is used. We show that the gain in rates achieved with the autoencoder based algorithm compared to the supervised algorithm and conventional hybrid BF algorithms increases from $18.7\%$ to $64\%$ in a more realistic environment, which shows the adaptability of our approach.




(a) 4-by-4 mmWave system. (b) 8-by-8 mmWave system. (c) 16-by-16 mmWave system.

Fig. 5. *Stochastic mmW model results:* Rates of unconstrained BF via SVD, unconstrained BF via autoencoder, hybrid BF via autoencoder, supervised hybrid BF [9], [10], conventional hybrid BF algorithms [3], [16], [17].

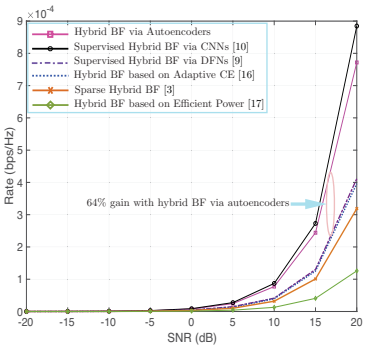



Fig. 6. *NYUSIM model results:* Rates of 16-by-16 mmWave system using hybrid BF via autoencoder and state-of-the-art.

## VI. Conclusion

In this paper, we first presented a linear autoencoder for estimating the unconstrained beamformers. We then introduced an autoencoder to design the hybrid beamformers under the constraints of the finite-precision phase shifters and the power restriction. With the stochastic and NYUSIM mmWave models, we compared the rates of the proposed approaches with the unconstrained BF, supervised approaches for hybrid BF, and conventional hybrid BF algorithms. Simulation results show that the proposed approach for unconstrained BF obtains up to $65\%$ gain in rates compared to the supervised algorithm [9]. Moreover, the unsupervised approach for hybrid BF achieves $30-40\%$ and $60-70\%$ gains compared to the algorithm based on DFNs when the stochastic and NYUSIM models are used, respectively. In the future, we will study different types of autoencoders and generative adversarial networks (GANs).

## Acknowledgement

This work was partially supported by the Broadband Wireless Access and Applications Center (BWAC); NSF Award No. 1822071; the work of R. Tandon was supported in part by NSF grants CAREER 1651492, CNS 1715947 and the 2018 Keysight Early Career Professor Award.

## References

[1] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What Will 5G Be?" *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, June 2014.

[2] A. F. Molisch, V. V. Ratnam, S. Han, Z. Li, S. L. H. Nguyen, L. Li, and K. Haneda, "Hybrid Beamforming for Massive MIMO: A Survey," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 134–141, Sep. 2017.

[3] O. E. Ayach, S. Rajagopal, S. Abu-Surra, Z. Pi, and R. W. Heath, "Spatially Sparse Precoding in Millimeter Wave MIMO Systems," *IEEE Trans. on Wireless Communications*, vol. 13, no. 3, pp. 1499–1513, March 2014.

[4] N. Li, Z. Wei, H. Yang, X. Zhang, and D. Yang, "Hybrid Precoding for mmWave Massive MIMO Systems With Partially Connected Structure," *IEEE Access*, vol. 5, pp. 15 142–15 151, 2017.

[5] S. Park, A. Alkhateeb, and R. W. Heath, "Dynamic Subarrays for Hybrid Precoding in Wideband mmWave MIMO Systems," *IEEE Transactions on Wireless Communications*, vol. 16, no. 5, pp. 2907–2920, May 2017.

[6] Y. Ren, Y. Wang, C. Qi, and Y. Liu, "Multiple-Beam Selection With Limited Feedback for Hybrid Beamforming in Massive MIMO Systems," *IEEE Access*, vol. 5, pp. 13 327–13 335, 2017.

[7] A. Alkhateeb, O. El Ayach, G. Leus, and R. W. Heath, "Hybrid precoding for millimeter wave cellular systems with partial channel knowledge," in *2013 Information Theory and Applications Workshop (ITA)*, Feb 2013, pp. 1–5.

[8] Y. Long, Z. Chen, J. Fang, and C. Tellambura, "Data-Driven-Based Analog Beam Selection for Hybrid Beamforming Under mm-Wave Channels," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 2, pp. 340–352, May 2018.

[9] A. Alkhateeb, S. Alex, P. Varkey, Y. Li, Q. Qu, and D. Tujkovic, "Deep Learning Coordinated Beamforming for Highly-Mobile Millimeter Wave Systems," *CoRR*, vol. abs/1804.10334, 2018.

[10] T. Peken, S. Adiga, R. Tandon, and T. Bose, "Deep Learning for SVD and Hybrid Beamforming." *IEEE Transactions on Wireless Communications*, submitted for publication in July 2019.

[11] A. Goldsmith, S. A. Jafar, N. Jindal, and S. Vishwanath, "Capacity limits of MIMO channels," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 5, pp. 684–702, June 2003.

[12] G. H. Golub and C. Reinsch, "Singular Value Decomposition and Least Squares Solutions," *Numer. Math.*, vol. 14, pp. 403–420, Apr. 1970.

[13] P. Baldi and K. Hornik, "Neural networks and principal component analysis: Learning from examples without local minima," *Neural Networks*, vol. 2, pp. 53–58, 1989.

[14] E. Plaut, "From principal subspaces to principal components with linear autoencoders," 2018.

[15] P. Baldi and K. Hornik, "Neural networks and principal component analysis: Learning from examples without local minima," *Neural Networks*, vol. 2, pp. 53–58, 1989.

[16] A. Alkhateeb, O. E. Ayach, G. Leus, and R. W. H. Jr., "Channel Estimation and Hybrid Precoding for Millimeter Wave Cellular Systems," *CoRR*, vol. abs/1401.7426, 2014.

[17] J. Singh and S. Ramakrishna, "On the Feasibility of Codebook-Based Beamforming in Millimeter Wave Systems With Multiple Antenna Arrays." *IEEE Trans. Wireless Communications*, vol. 14, no. 5, pp. 2670–2683, 2015.

[18] A. A. M. Saleh and R. Valenzuela, "A Statistical Model for Indoor Multipath Propagation," *IEEE Journal on Selected Areas in Communications*, vol. 5, no. 2, pp. 128–137, February 1987.

[19] M. K. Samimi and T. S. Rappaport, "3-D Millimeter-Wave Statistical Channel Model for 5G Wireless System Design," *IEEE Transactions on Microwave Theory and Techniques*, vol. 64, no. 7, pp. 2207–2225, July 2016.