



Wei Yu , Fellow, IEEE | University of Toronto, Toronto, ON, M5S 3G4, Canada | E-mail: weiyu@comm.utoronto.ca  
 Foad Sohrabi, Member, IEEE | Nokia Bell Labs, Murray Hill, NJ 07974 USA | E-mail: foad.sohrabi@nokia-bell-labs.com  
 Tao Jiang , Graduate Student Member, IEEE | University of Toronto, Toronto, ON, M5S 3G4, Canada | E-mail: taoca.jiang@mail.utoronto.ca

# Role of Deep Learning in Wireless Communications

**Abstract**—Traditional communication system design has always been based on the paradigm of first establishing a mathematical model of the communication channel, then designing and optimizing the system according to the model. The advent of modern machine learning techniques, specifically deep neural networks, has opened up opportunities for data-driven system design and optimization. This article draws examples from the optimization of reconfigurable intelligent surface, distributed channel estimation and feedback for multiuser beamforming, and active sensing for millimeter wave initial alignment to illustrate that a data-driven design that bypasses explicit channel modeling can often discover excellent solutions to communication system design and optimization problems that are otherwise computationally difficult to solve. We show that by performing an end-to-end training of a deep neural network using a large number of channel samples, a machine learning-based approach can potentially provide significantly system-level improvements as compared to the traditional model-based approach for solving optimization problems. The key to the successful applications of machine learning techniques are in choosing the appropriate neural network architecture to match the underlying problem structure.

## Introduction

Modern machine learning techniques, specifically deep neural networks (DNNs), have enabled tremendous progress for diverse applications, ranging from speech recognition, natural language processing, and image classification, to data analytics and self-driving cars, and many more. In this article, we

ask the following question: Is there a role for machine learning in physical-layer wireless communications system design? If so, where do opportunities lie, and where would the potential benefits come from?

Fundamental to the phenomenal success of the machine learning techniques across a wide range of applications is its apparent universal ability to approximate any functional mapping from an input space to an output space, given sufficiently complex neural network structure and enough training data [1]. In fact, common characteristics of application domains where machine learning has made the most impact, are that the inputs to these tasks are high-dimensional complex data, whose structure needs to be explored, while the outputs of these tasks can either be categorical (e.g., classification, segmentation, and sentiment analysis) or have complex structures themselves (e.g., machine translation, image labeling). The field of machine learning has developed myriad techniques to enable automatic feature extraction and to explore the structure of the problem in order to efficiently train a DNN to map the input to the desired output. The machine learning paradigm essentially solves optimization problems by pattern matching. This is a vastly different philosophy as compared to the traditional model-based information theoretical approach to communication system design.

This article aims to illustrate that machine learning has an important role to play even in the physical-layer wireless communications, which has traditionally been dominated by model-based design and optimization approaches. This is so for several reasons:

- (i) Traditional wireless communication design methodologies typically rely on the channel model, but models are inherently only an approximation to the reality. In applications where the models are complex and the channels are difficult to estimate, a data-driven methodology that allows the system design to bypass explicit channel estimation can potentially be a better approach.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see <https://creativecommons.org/licenses/by-nc-nd/4.0/>  
 Digital Object Identifier 10.1109/MBITS.2022.3212978  
 Date of publication 10 October 2022; date of current version 14 March 2023.

- (ii) Modern wireless communication applications often involve optimization problems that are highly dimensional, nonconvex, and difficult to solve efficiently. By exploiting the availability of training data, a machine learning approach may be able to learn the solutions of the optimization problems directly. This can lead to a more efficient way to explore the nonconvex optimization landscape than the traditional model-based optimization approaches.
- (iii) Traditional communication system designs are based on the principle of source-channel separation and the optimal design of compression and channel codes. But when the encoder and the decoder are block-length and/or complexity constrained, or when the overall communication scenario involves multiple transmitters and multiple receivers, the optimal design of a practical encoder and decoder is highly challenging. In this realm, there is the potential for discovering better source and channel encoders and decoders using machine learning, as many of these code design problems boil down to solving optimization problems over the codebook structure for which data-driven methods may be able to identify better solutions more efficiently.

The field of machine learning for communication system design has exploded in recent years [2], [3], [4] [5]. We mention some of the references here, e.g., in source and channel coding [6], [7], [8], waveform design [9], signal detection [10], [11], [12], resource allocation [13], [14], [15], [16], [17] [18], and channel estimation [19], [20], etc. This article does not attempt to do justice in surveying the entire literature and the recent progress on this topic. Instead, we focus on the questions of why and how machine learning can benefit wireless communication system design by presenting the following three specific examples.

First, we consider communication scenarios in which a naive parameterization of the channel would involve a large number of parameters, thus making the channel estimation is a challenging task. Specifically, we show that in a wireless communication system involving a reconfigurable intelligent surface (RIS), comprising of a larger number of reflective elements, a machine-learning approach that directly optimizes the reflection coefficients without first estimating the channel can significantly improve the overall performance [21].

Second, we consider a distributed source coding problem in the context of channel estimation and feedback for a massive multiple-input multiple-output (MIMO) system, and show that short block-length code design for distributed data compression with system-level objectives is feasible and can result in significant performance improvements over the single-user data compression codebook design [22].

Third, we use an active sensing problem for the millimeter wave (mmWave) initial alignment to illustrate the role of machine learning in exploring the optimization landscape in a complex sequential learning problem [23]. We show that selecting the right neural network architecture to match the problem structure is crucial for its success.

## Information Theoretical Approach to Communication System Design

Information theory has been the guiding principle in the development of communication system design in the past seventy years. The driving philosophy in information theory has always been reductionist—putting it in words of a famous quote: *everything should be as simple as possible, but no simpler*. A celebrated example of this philosophy is the additive white Gaussian noise (AWGN) channel model, in which the choice of the Gaussian noise distribution is justified both by a central limit theorem argument based on the assumption that the overall noise is comprised of many independent small components and by the fact that the Gaussian distribution is the worst-case noise distribution for the additive channel. The AWGN model is cherished in the research community and has played a central role in many historical developments in communication theory (e.g., from time-domain equalization, to orthogonal frequency-division multiplex, to multiuser detection), in coding theory (e.g., from maximum likelihood decoding, to Viterbi algorithm, to Turbo, low-density parity-check, and polar codes), and in multiuser information theory (e.g., from multiple-access, to broadcast, and to interference channel models).

The wireless channels are, however, much more complicated than the AWGN channel model. The wireless channel can be frequency selective; it is inherently time-varying; it often involves multiple users and multiple antennas. Historically, communication engineers have invested heavily in developing models for various types of wireless channels. These models are often based on the physics of electromagnetic wave propagation; many of these models are statistical in nature; these channel models have played an important role in the design, analysis, performance evaluation, and standardization of generations of wireless systems [24].

Channel modeling is important in wireless communication engineering because most modern wireless systems operate under the framework of first estimating the channel, then feeding back the estimated channel to the transmitter, and finally optimizing transmission and reception strategies to maximize the mutual information between the input and the output. In this article, we argue, however, that this model-then-optimize approach is not necessarily always the best approach.

## From Model-Based Optimization to Learning-Based Design

In traditional communication system design, maximizing the capacity of a wireless link typically requires channel estimation; the process of channel estimation always depends on the channel model. Choosing which model to use is, however, an art rather than a science. This is because wireless channels often have inherent structures that make certain models more appropriate than others. For example, a MIMO channel with  $M$  transmit antennas and  $N$  receive antennas can simply be modeled as a  $M \times N$  matrix. But a mmWave massive MIMO channel often has a sparsity structure, corresponding to the finite number of propagation paths from the transmitter to the receiver, so that a sparse path-based model in the spatial domain is a more efficient representation of the channel. Likewise, a frequency-selective channel can be modeled by its channel response across frequencies. But, the frequency selectivity is usually a consequence of the different delays across the multiple paths, so the channel variations across the frequencies are correlated. Instead of estimating the channel in the frequency domain, a multipath time-domain model may be more appropriate.

Moreover, the channel estimation process requires specifying a loss function. The squared-error metric is often adopted for tractability reasons, but minimizing the mean-squared-error (MSE) of the estimates of the channel parameters does not necessarily correspond to maximizing the overall system objective. For example, some parts of the channel may be more important to describe than others. Clearly, the specific parameterization of the channel and the choice of the estimation error metric have a significant impact on the ultimate system performance.

Traditionally, wireless researchers rely on experience and engineering judgment in choosing the best channel model and the best optimization formulation. The design decisions need to balance the inherent tradeoffs between:

- i) how complex the model is, e.g., the number of parameters in the model;
- ii) how well the model approximates the reality;
- iii) how easy it is to estimate the model parameters;
- iv) how easily the model can be used for subsequent transmitter and receiver optimization.

We emphasize that in a wireless fading channel with limited coherence time/frequency, model estimation comes at a significant cost in terms of the coherence slots occupied by pilot transmissions. For example, a highly complex model may better approximate the reality, but may require too many pilots for parameter estimation, hence may not be worth the effort. The point is that there is no universal theory about

how to choose the best channel model and how to best perform channel estimation. To characterize and to take advantage of the underlying channel structure in the design of the channel estimation process require engineering intuition and highly nontrivial tasks.

In contrast, this article shows that a machine-learning approach can be used to allow automatic discovery of the appropriate representation of the channel based on training data. Furthermore, it allows the optimization of the system metric that actually matters (e.g., the achievable rate as opposed to the MSE of the channel reconstruction) without having to first explicitly estimate the channel. This can have a significant advantage as illustrated in the example of optimizing the RIS coefficients directly based on the received pilots to maximize the capacity of RIS system and the application of neural networks for channel feedback for the massive MIMO system.

### Model-Based Optimization

In many communication system design problems, even if the model parameters are perfectly estimated, the resulting transmitter and receiver optimization problem may still be not so easy to solve. The formulation of the optimization problem is also an art rather than a science. In fact, wireless engineers often adopt optimization formulations, *because* the resulting mathematical programming problem is amendable to either analytic or computationally efficient numerical solution. We remark that a mathematical optimization problem can often be parameterized in many different ways. The “holy grail” of mathematical optimization is often thought of as to transform a problem into a convex form so that computationally efficient numerical procedures can be developed to find the global optimal solution of the resulting mathematical programming problem. But there is no universal theory about how best to transform the optimization landscape.

In contrast, this article shows that a machine-learning approach can be used for the automatic discovery of the mapping from the problem representation to the optimal solution based on training data. This is illustrated in the examples of optimizing RIS coefficients based on received pilots to maximize the capacity of RIS system, and optimizing beamformers based on channel feedback for FDD massive MIMO system, finally optimizing a sequence of active sensing strategies for mmWave channel initial alignment in the subsequent sections.

### Data-Driven Communication System Design

The article advocates the viewpoint that a data-driven approach can circumvent many of the modeling and optimization difficulties for wireless system design as

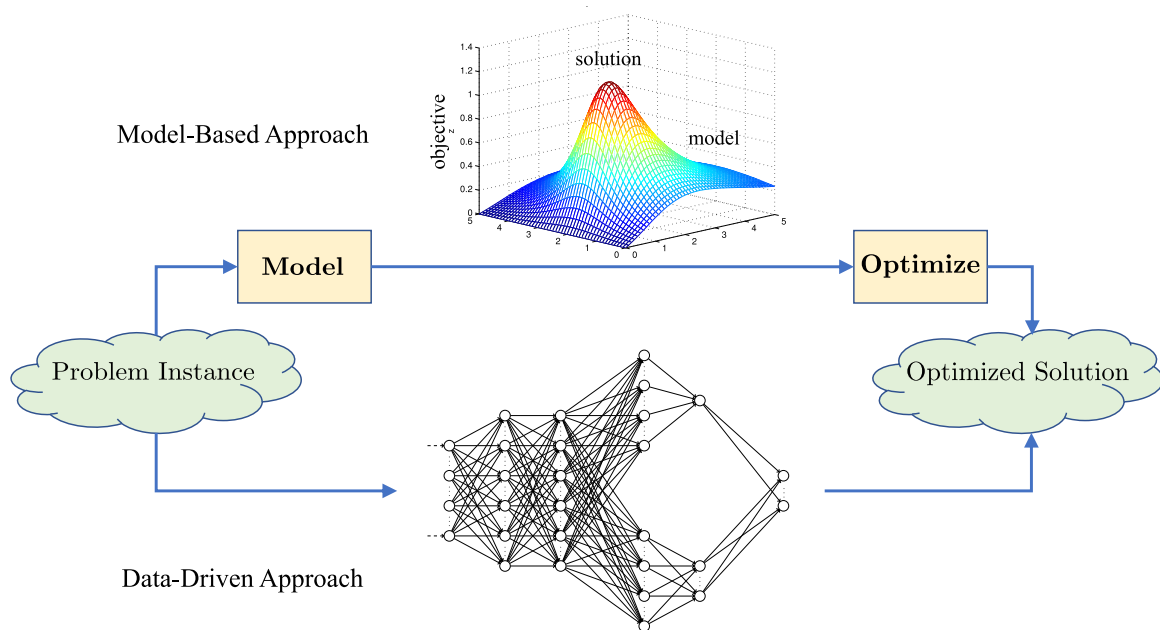


Figure 1

Traditional wireless system design follows the paradigm of model-then-optimize (top branch). The design problem is modeled mathematically; the model parameters are then estimated, which allows the associated optimization landscape to be characterized; finally, the optimal solution is obtained by mathematical programming. The machine-learning approach (bottom branch) aims to directly learn the optimal solution from a representation of the problem instance. The neural network is trained over many problem instances, by adjusting its weights according to the overall system objective as a function of the representation of the problem instances.

mentioned in the previous section. The main idea is shown in Figure 1. Instead of the traditional model-then-optimize approach, which involves choosing an appropriate parameter space, then characterizing the associated optimization landscape, and finally performing the resulting mathematical optimization, we adopt a data-driven approach to directly map the problem instances to the corresponding optimized solutions. By training such a neural network over many problem instances, the task of optimization is essentially turned into *pattern matching*. When a new optimization task comes along, the trained neural network can then simply output the corresponding solution. This is akin to a human learner who is trained to use past experience to perform future optimization tasks.

The advantages of the proposed data-driven paradigm are the following:

- (i) It allows direct system-level optimization without the intermediary step of channel estimation. The modeling uncertainty and the channel estimation error are implicitly taken into account in the overall optimization process.

- (ii) It allows an end-to-end design with a realistic system-level objective function, instead of relying on some arbitrary metric in the model parameter estimation process.
- (iii) It allows the problem instances to be represented in an arbitrary fashion. Additional side information which is often not easy to incorporate into a model can now be accounted for in the optimization process.
- (iv) By using a large number of problem instances as training data, it allows the optimization process to efficiently explore the high-dimensional optimization landscape in the training stage.
- (v) Once trained, the neural network can efficiently output the optimized solution for new problem instances. In effect, the computational complexity is moved from the optimization stage to the neural network training process.

Thus, instead of using a mathematical optimization approach that requires highly structured models over well-defined problems and relies on the specific (e.g., convex) structure of the optimization landscape, a machine-learning approach is capable of solving relatively poorly defined problems and exploring high-dimensional optimization space without first identifying the problem structure. This is made possible because of the ability of the neural network to find patterns in

the vast amount of training data, thanks to the nowadays prevalent highly parallel computer architectures for both neural network training and implementation processes [25], [26].

Machine learning is about approximating functions—its broad impact comes from the fact that it is particularly effective in processing high-dimensional data. The phenomenal success of deep learning in domains such as image and speech processing is due to the fact that the specific task at hand is often governed by some low-dimensional characteristics (e.g., labels) embedded in high-dimensional observations (e.g., images). As we shall see in the examples in the sequel, the wireless communications scenarios in which the data-driven optimization can be shown to substantially outperform the traditional model-driven design are also precisely the situations in which the problem instances have some low-dimensional structure and are observable only through a limited number of high-dimensional outputs. In the communications setting, the observations are typically the received pilots; the low-dimension problem structure is typically due to the sparsity of the underlying wireless channel. The benefit of machine learning comes from bypassing the explicit modeling of the channel structure and instead using a DNN to directly process the high-dimensional received pilots to arrive at the desired communication action. The remaining of this article uses three examples to illustrate the success of machine learning in wireless applications.

## Capacity Maximization Using Reconfigurable Intelligent Surface

Wireless channels are often highly dimensional. This is the case for massive MIMO systems in which the transmitters and the receivers are equipped with large antenna arrays, and is also true of emerging devices such as a class of metasurfaces known as RIS, which consists of a large number of reflecting elements and can be dynamically reconfigured to focus electromagnetic waves to the intended receivers for the purpose of maximizing the overall channel capacity [27].

The physical electromagnetic propagation environment of a wireless channel is also often sparse, especially as compared to the number of elements in the antenna array or the reflective surface. This is because the propagation characteristics typically only depend on a small number of scatters, and the number of propagation paths in the environment can be significantly less than the number of transmitting and receiving antennas or the number of reflecting elements. On the other hand, due to the limited number of radio-frequency (RF) chains and the finite pilot overhead, the available observations of the channel are typically limited.

How can we estimate a sparse high-dimensional channel through a limited number of observations? The traditional approach is to take advantage of the channel sparsity and to

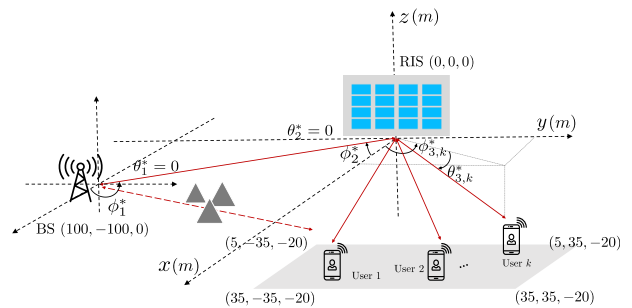


Figure 2  
Geographic layout of an RIS-assisted downlink system [21].

build a channel model with a small number of parameters, then proceed with estimating the parameters of the channel based on the received pilots, followed by optimizing the system according to the estimated channel. How well such an approach works would depend on how well the model approximates the actual channel. In this section, we advocate an alternative data-driven approach that bypasses the explicit modeling stage and directly optimizes the system using a neural network with the received pilots as inputs. We use the RIS as an example in which explicit channel estimation is especially challenging, but the proposed approach can be adopted equally well in many other scenarios, including the conventional massive MIMO system.

A commonly used model for RIS is to regard it as a device consisting of a large number of tunable elements that can reflect incoming signals with arbitrary phase shifts (see Figure 2). The goal is to dynamically reconfigure the phase shifts at the RIS according to the channel realizations of the users in order to maximize a system-level metric, e.g., the system downlink throughput. The problem is that channel estimation is highly nontrivial for the RIS system. Assuming a time-division duplex (TDD) system with channel reciprocity, channel estimation can be done using uplink pilots. However, the large number of RIS elements gives rise to a high-dimensional channel, which would require many pilots to estimate. Furthermore, even if the channel can be accurately estimated, the optimization of the RIS coefficients is a highly complex and nonconvex optimization problem, which is difficult to solve.

We show that the approach of using machine learning to directly map the received pilots to the optimized RIS reflective coefficients can yield a significant performance improvement as compared to the traditional channel estimation based approach [21]. The performance gain comes from the fact that channel models are only an approximation of the reality and that traditional channel estimation always needs to assume an estimation error metric (such as the MSE), but such metric does not perfectly match the system-level objective. This problem can be alleviated by bypassing the modeling stage,



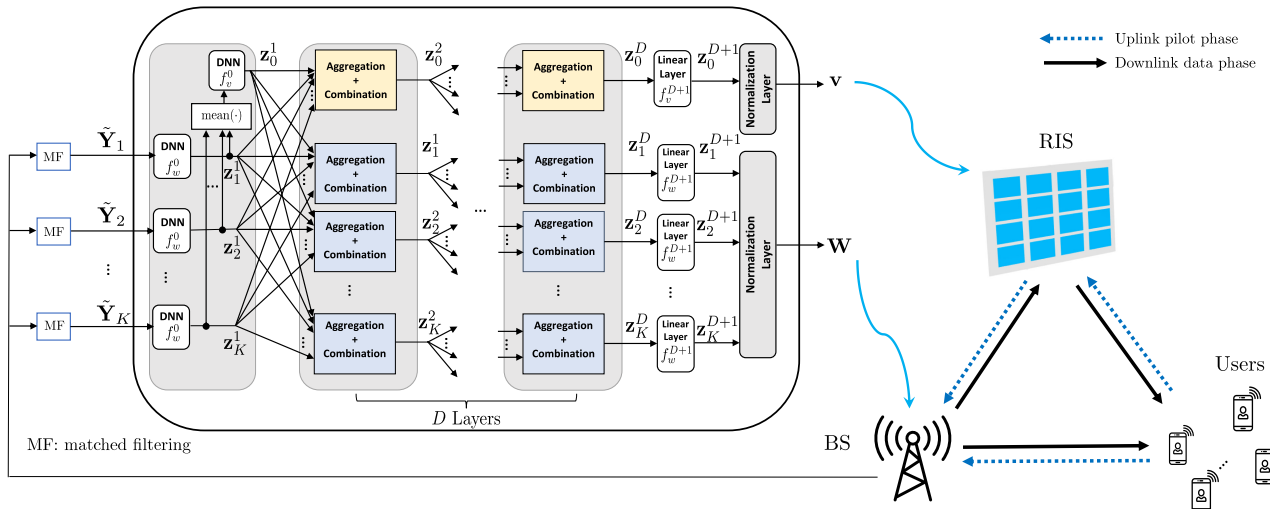


Figure 3

Deep learning framework for directly designing the multiuser beamformers and reflection coefficients based on the received uplink pilots for a downlink RIS-assisted multiuser system.

by using the true system objective as the loss function, and by training a neural network to directly output the optimized reflective coefficients based on the received pilots. Essentially, the wireless channel is now represented by the received pilots. The complexity of high-dimensional optimization is shifted to the training stage, where a large number of channel instances and the corresponding reflecting coefficients are processed by the neural network so that it can produce the desired solution when a new channel realization is observed.

Choosing the right architecture for the neural network turns out to be important. For this application, we experimentally find that the best system-level performance is obtained by adopting a graph neural network (GNN) [15], [16], [28] that captures the spatial relationship between the base-station (BS), the RIS, and the users. The proposed approach and the interpretations of the solutions are presented in the following.

### System Model and Problem Formulation

Consider an RIS-assisted MIMO system with a BS equipped with  $M$  antennas serving  $K$  single-antenna users as shown in Figure 2. An RIS consisting of  $N$  elements is deployed between the BS and the users to enable a reflection link. Let  $\mathbf{h}_k^d \in \mathbb{C}^M$  denote the direct channel from the BS to user  $k$ , and  $\mathbf{h}_k^r \in \mathbb{C}^N$  denote the channel from the RIS to user  $k$ , and  $\mathbf{G} \in \mathbb{C}^{M \times N}$  denote the channel from the RIS to the BS. We assume a block-fading channel model. In the downlink, the BS sends the data symbol  $s_k \in \mathbb{C}$  with  $\mathbb{E}[|s_k|^2] = 1$  to user  $k$  using a beamforming vector  $\mathbf{w}_k \in \mathbb{C}^M$ , which satisfies a total power constraint  $\sum_{k=1}^K \|\mathbf{w}_k\|_2^2 \leq P_d$ . The RIS reflection coefficients are denoted by  $\mathbf{v} = [e^{j\omega_1}, e^{j\omega_2}, \dots, e^{j\omega_N}]^T$ , where  $\omega_i \in (-\pi, \pi]$  is the phase shift of the  $i$ th element. Then, the

received signal at user  $k$  is represented as

$$\begin{aligned} r_k &= \sum_{j=1}^K (\mathbf{h}_k^d + \mathbf{G} \text{diag}(\mathbf{v}) \mathbf{h}_k^r)^\top \mathbf{w}_j s_j + n_k \\ &= \sum_{j=1}^K (\mathbf{h}_k^d + \mathbf{A}_k \mathbf{v})^\top \mathbf{w}_j s_j + n_k \end{aligned} \quad (1)$$

where  $\mathbf{A}_k = \mathbf{G} \text{diag}(\mathbf{h}_k^r) \in \mathbb{C}^{M \times N}$  denotes the cascaded channel from the BS to user  $k$  through reflection at the RIS, and  $n_k \sim \mathcal{CN}(0, \sigma_0^2)$  is the AWGN. The  $k$ th user's achievable rate  $R_k$  is computed as

$$R_k = \log_2 \left( 1 + \frac{|\mathbf{h}_k^d + \mathbf{A}_k \mathbf{v}|^\top \mathbf{w}_k|^2}{\sum_{i \neq k} |\mathbf{h}_k^d + \mathbf{A}_k \mathbf{v}|^\top \mathbf{w}_i|^2 + \sigma_0^2} \right). \quad (2)$$

The overall problem is to maximize some network utility function  $\mathcal{U}(R_1, \dots, R_K)$  by optimizing the beamforming vectors at the BS  $\{\mathbf{w}_k\}_{k=1}^K$  and the RIS reflection coefficients  $\mathbf{v}$ . Now, since the channel coefficients are not known, we need to use a pilot training phase to learn the channel. Assuming a TDD system with channel reciprocity, we let each user  $k$  send an uplink pilot sequence  $x_k(\ell)$ ,  $\ell = 1, \dots, L$ , with  $|x_k(\ell)|^2 \leq P_u$ , to the BS. Then, the received pilots at the BS can be denoted as

$$\mathbf{y}(\ell) = \sum_{k=1}^K (\mathbf{h}_k^d + \mathbf{A}_k \tilde{\mathbf{v}}(\ell)) x_k(\ell) + \mathbf{n}(\ell) \quad (3)$$

where  $\tilde{\mathbf{v}}(\ell)$  is the vector of RIS reflection coefficients at the uplink transmission slot  $\ell$  and can be thought of as part of the pilot, and  $\mathbf{n}(\ell) \sim \mathcal{CN}(\mathbf{0}, \sigma_1^2 \mathbf{I})$  is the additive Gaussian noise. Denoting  $\mathbf{Y} = [\mathbf{y}(1), \mathbf{y}(2), \dots, \mathbf{y}(L)] \in \mathbb{C}^{M \times L}$  and

$\mathbf{W} = [\mathbf{w}_1, \dots, \mathbf{w}_K]$ , our goal is to design the downlink beamformers  $\mathbf{W}$  and the reflection coefficients  $\mathbf{v}$ , based on the received uplink pilots  $\mathbf{Y}$ , which contains information about the channel. This overall process can be thought of as solving the following optimization problem over the mappings from  $\mathbf{Y}$  to  $(\mathbf{W}, \mathbf{v})$

$$\underset{(\mathbf{W}, \mathbf{v}) = \mathcal{G}(\mathbf{Y})}{\text{maximize}} \quad \mathbb{E}[\mathcal{U}(R_1(\mathbf{W}, \mathbf{v}), \dots, R_K(\mathbf{W}, \mathbf{v}))] \quad (4a)$$

$$\text{subject to} \quad \sum_k \|\mathbf{w}_k\|_2^2 \leq P_d \quad (4b)$$

$$|v_i| = 1, \quad i = 1, 2, \dots, N \quad (4c)$$

where the function  $\mathcal{G}(\cdot) : \mathbb{C}^{M \times L} \rightarrow \mathbb{C}^{M \times K} \times \mathbb{C}^N$  is the mathematical representation of the mapping to be optimized over, and the expectation is taken over the random channel realizations and the uplink noise.

Directly solving problem (4) is challenging, because it involves optimizing over the high-dimensional mapping  $\mathcal{G}(\cdot)$ . The conventional approach is to first estimate the channels from the received pilots  $\mathbf{Y}$ , then to solve the subsequent network utility maximization problem based on the estimated channel. Instead, we propose a machine-learning approach to directly learn such a mapping using a GNN.

### Learning to Beamform and to Reflect

The overall learning framework is shown in Figure 3, where the received pilots after matched filtering  $\{\mathbf{Y}_k\}_{k=1}^K$  is the input to a neural network that learns the optimized reflection coefficients  $\mathbf{v}$  and the beamforming matrix  $\mathbf{W}$  without the intermediary channel estimation step.

The remaining key question is how to choose the neural network architecture. In theory, a fully connected neural network can already learn the mapping from the received pilots to the optimization variables. However, a more efficient architecture is one that captures the structure of the network utility maximization problem (4). Specifically, observe that in (4), if the indices of users permute, the optimal RIS coefficients  $\mathbf{v}$  should remain the same, while the optimal beamforming vectors  $\{\mathbf{w}_k\}_{k=1}^K$  should permute in the same manner. These properties are known as *permutation invariance* and *permutation equivariance*.

It is possible to design a neural network to automatically enforce these properties. This can be done using a GNN based on a graph representation of the RIS and the users. The details of the GNN structure are described in [21]. The idea is to associate a representation vector  $\mathbf{z}_k^d$  with each user and also with the RIS. The representation vectors are updated layer-by-layer, but the connections between the layers are based on aggregation and combination operations that are invariant with

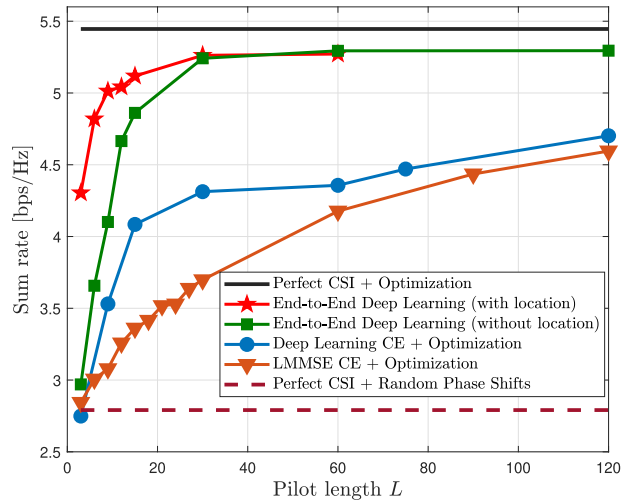


Figure 4

Sum rate versus pilot length for an RIS-assisted multiuser downlink system with an 8-antenna BS, a 100-element RIS, and three single-antenna users, comparing end-to-end deep learning approach to the conventional approach of channel estimation (CE) followed by RIS coefficients and BS beamforming optimization [21].

respect to input permutation, e.g., the  $\text{mean}()$  or  $\text{max}()$  functions. After multiple layer iterations, the node representation vectors are mapped to the beamforming matrix  $\mathbf{W}$  and the RIS coefficients  $\mathbf{v}$ . To make the architecture generalizable with respect to the number of users, the neural network weights across the users are tied together. The overall neural network can be trained to maximize the network utility function.

### Numerical Results

To illustrate the performance of the machine-learning approach for optimizing the beamformers and the reflective coefficients, we report the simulation results<sup>1</sup> in [21] on a scenario with  $M = 8$  antennas at the BS,  $N = 100$  elements at the RIS, and three users. The direct-link channel  $\mathbf{h}_k^d$  is assumed to be Rayleigh fading, and the BS-RIS and RIS-users channels are assumed to be Rician fading with Rician factor set as 10. The geographic locations of the BS, RIS, and users are shown in Figure 2. The uplink pilot transmits power and the downlink data transmit power are respectively set to be 15 and 20 dBm. The uplink and downlink noise power are  $-100$  and  $-85$  dBm, respectively.

Figure 4 plots the average sum rate versus pilot length for different approaches. As can be seen from Figure 4, the

<sup>1</sup>The code for this simulation is available at <https://github.com/taojiang-github/GNN-IRS-Beamforming-Reflection>.

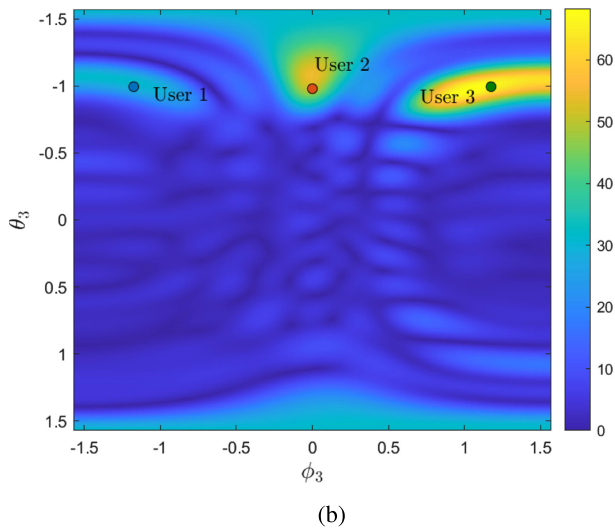
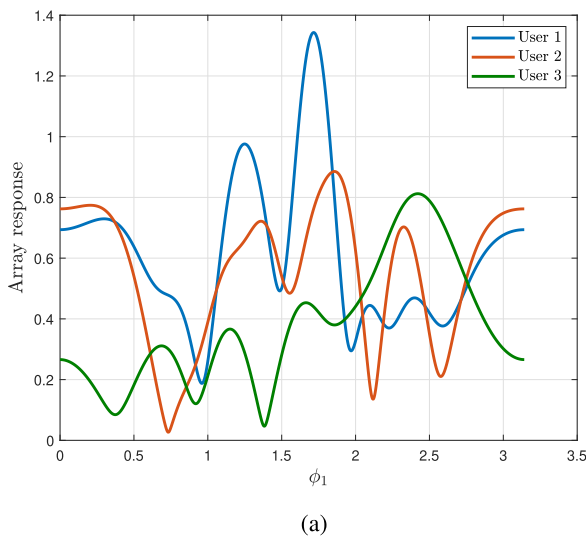


Figure 5

Array response of the BS and the RIS obtained from the GNN for  $N = 100$  and  $M = 8$  for a three-user system for maximizing the minimum rate. The true  $(\phi_3^*, \theta_3^*)$  are  $(-1.176, -0.994)$ ,  $(0, -0.980)$ , and  $(1.176, -0.994)$ , respectively, for the three users [21]. (a) Array response of the BS. (b) Array response of the RIS.

performance of the linear minimum mean-squared-error (LMMSE) channel estimation based method is able to approach the perfect channel state information (CSI) baseline as the pilot length increases. However, the end-to-end deep learning method approaches the perfect CSI baseline much faster, showing that the GNN can utilize the pilots in a more efficient way.

We also provide the simulation results of the model-then-optimize approach in which a GNN is used for explicit channel estimation, and the beamforming matrix and RIS coefficients are optimized based on the estimated channel. While this method shows better performance as compared to the LMMSE-based approach, its performance is still much worse than the GNN approach, which directly learns the solution from the pilots. This shows the benefit of bypassing explicit channel estimation. Moreover, additional information, such as the locations of the users, can be easily incorporated into the end-to-end deep learning framework, which can further improve the performance, as shown in Figure 4.

The GNN produces interpretable solutions. Figure 5 shows the array responses learned by the GNN for a problem of maximizing the minimum rate for three users at different locations. We observe from Figure 5(b) that the learned RIS coefficients indeed focus the beams to the corresponding user locations, but the three users get different focusing strengths. Interestingly, because the BS beamformers and the RIS reflective coefficients are designed jointly, the user corresponding to the weakest RIS focusing is compensated by a stronger BS beamforming gain as seen in Figure 5(a). Thus, the combined channel strengths are equalized across the three users. Overall, these results show that the GNN

indeed is able to learn interpretable solutions, based on much fewer pilots than the conventional strategies.

## Distributed Source Coding for Channel Estimation and Feedback in FDD Massive MIMO

The channel estimation problem is more challenging in the frequency-division duplex (FDD) system, which cannot rely on channel reciprocity. In this case, as shown in Figure 6, the pilots are sent by the BS in the downlink and are observed by the users. The users need to estimate their channels, then send quantized versions of the channels through rate-limited feedback links to the BS, so that the BS can design a precoding strategy to serve all the users. The conventional approach to this problem relies on model-based channel estimation followed by independent codebook-based quantization and feedback [29], [30], [31]. This is far from optimal. We show here that machine-learning techniques can be used to train a set of optimized distributed source encoders together with a centralized decoder in an end-to-end fashion in order to maximize a system-level objective. Such an approach can significantly reduce the length of pilots needed to achieve the maximum throughput in an FDD massive MIMO system.

The channel estimation and feedback design for a multiuser FDD massive MIMO system can be thought of as a distributed source coding problem. Distributed source coding is a long-standing information theoretical problem in which distributed encoders compress their observations for centralized reconstruction at the decoder. Here, the users are the distributed source encoders



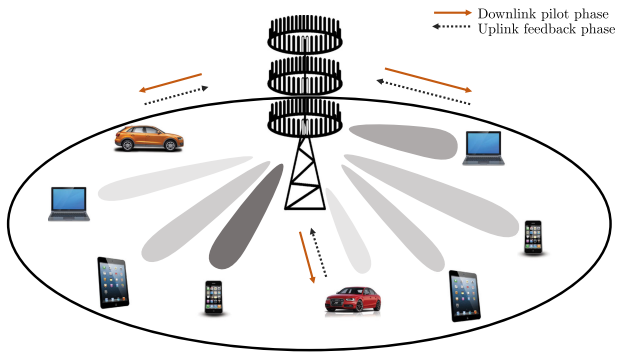


Figure 6

FDD massive MIMO system, in which the BS transmits the pilots, the users estimate their channels then feedback a quantized version of the channels to BS, and the BS designs the precoders based on the feedback from all the users [22].

who observe and then quantize a noisy version of the channels. The BS is the centralized source decoder, which aims to compute a function of the channels, namely the precoder.

The optimal design of distributed source encoders and decoders is highly nontrivial. Information-theoretic optimal coding strategies involve concepts, such as binning, which can be thought of as a multiuser codebook. While it is unlikely for a neural network to learn structured binning, it can help design good codebook-based quantization and feedback strategies that reap the benefit of distributed source coding. This is an example in which a data-driven approach can play an important role in designing short block-length quantization codes under rate constraints.

### System Model and Problem Formulation

Consider an FDD multiuser MIMO system in which a BS equipped with  $M$  antennas serves  $K$  single-antenna users. Analogues to the previous section, we consider the downlink scenario in which the BS aims to communicate the data symbol  $s_k \in \mathbb{C}$  with  $\mathbb{E}[|s_k|^2] = 1$  to user  $k$  using a precoding vector  $\mathbf{w}_k \in \mathbb{C}^M$ , which satisfies a total power constraint  $\sum_{k=1}^K \|\mathbf{w}_k\|_2^2 \leq P_d$ . Assuming a narrowband block-fading channel model, the received signal at the  $k$ th user in the data transmission phase can be written as

$$r_k = \mathbf{h}_k^\top \mathbf{w}_k s_k + \sum_{i \neq k} \mathbf{h}_k^\top \mathbf{w}_i s_i + z_k \quad (5)$$

where  $\mathbf{h}_k \in \mathbb{C}^M$  is the channel between the BS and user  $k$  and  $z_k \sim \mathcal{CN}(0, \sigma_0^2)$  is the AWGN. The achievable rate of user  $k$  is given by

$$R_k = \log_2 \left( 1 + \frac{|\mathbf{h}_k^\top \mathbf{w}_k|^2}{\sum_{i \neq k} |\mathbf{h}_k^\top \mathbf{w}_i|^2 + \sigma_0^2} \right). \quad (6)$$

The aim is to maximize a network utility function  $\mathcal{U}(R_1, \dots, R_K)$ , which is a function of the precoding vectors  $\{\mathbf{w}_k\}_{k=1}^K$ . To design the optimal precoding vectors, the BS must first acquire the instantaneous CSI. We consider a pilot phase for the FDD system in which the BS sends pilots  $\mathbf{X} \in \mathbb{C}^{M \times L}$  of length  $L$ , and the  $k$ th user receives  $\mathbf{y}_k \in \mathbb{C}^{1 \times L}$  as

$$\mathbf{y}_k = \mathbf{h}_k^\top \mathbf{X} + \mathbf{n}_k \quad (7)$$

where the pilots in the  $\ell$ th transmission satisfy the power constraint, i.e.,  $\|\mathbf{x}_\ell\|_2^2 \leq P_d$  with  $\mathbf{x}_\ell$  being the  $\ell$ th column of  $\mathbf{X}$ , and  $\mathbf{n}_k \sim \mathcal{CN}(\mathbf{0}, \sigma_0^2 \mathbf{I})$  is the AWGN at user  $k$ . Subsequently, the  $k$ th user abstracts the useful information in the received pilots  $\mathbf{y}_k$  for the purpose of multiuser downlink precoding, and feeds back that information to the BS under a feedback constraint of  $B$  bits, i.e.,

$$\mathbf{q}_k = \mathcal{F}_k(\mathbf{y}_k) \quad (8)$$

where the function  $\mathcal{F}_k: \mathbb{C}^{1 \times L} \rightarrow \{\pm 1\}^B$  is the  $k$ th user's feedback scheme. Finally, the BS designs the multiuser precoding matrix  $\mathbf{W} = [\mathbf{w}_1, \dots, \mathbf{w}_K]$  based on the feedback bits received from all  $K$  users (i.e.,  $\mathbf{q} = [\mathbf{q}_1^\top, \mathbf{q}_2^\top, \dots, \mathbf{q}_K^\top]^\top$ ), i.e.,

$$\mathbf{W} = \mathcal{P}(\mathbf{q}) \quad (9)$$

where the function  $\mathcal{P}: \{\pm 1\}^{KB} \rightarrow \mathbb{C}^{M \times K}$  denotes the multiuser downlink precoding scheme.

The overall problem formulation is therefore

$$\text{maximize}_{\mathbf{x}, \{\mathcal{F}_k(\cdot)\}_{k=1}^K, \mathcal{P}(\cdot)} \mathcal{U}(R_1, \dots, R_K) \quad (10a)$$

$$\text{subject to } \mathbf{W} = \mathcal{P}([\mathbf{q}_1^\top, \dots, \mathbf{q}_K^\top]^\top) \quad (10b)$$

$$\mathbf{q}_k = \mathcal{F}_k(\mathbf{h}_k^\top \mathbf{X} + \mathbf{n}_k), \quad \forall k \quad (10c)$$

$$\sum_k \|\mathbf{w}_k\|_2^2 \leq P_d \quad (10d)$$

$$\|\mathbf{x}_\ell\|_2^2 \leq P_d, \quad \forall \ell \quad (10e)$$

in which the training pilots  $\mathbf{X}$ , all  $K$  users' feedback schemes  $\{\mathcal{F}_k(\cdot)\}_{k=1}^K$ , and the multiuser precoding scheme  $\mathcal{P}(\cdot)$  can be designed to optimize the overall utility function of the system.

This problem can be viewed as a distributed source coding problem with the network utility as the "distortion" metric, because channel estimation and quantization are performed across  $K$  distributed users, and the feedback bits from all  $K$  users are centrally processed at the BS for the purpose of designing the multiuser precoder, as illustrated in Figure 7(a). Obtaining the optimal distributed source coding strategy by directly solving the optimization problem (10) is challenging. As shown in Figure 7(b), the conventional design of FDD massive MIMO system is based on independent quantization and feedback of the channel vector (or channel parameters) at each user. However, such independent quantization and feedback approach is quite suboptimal, especially in the short pilot regime. In this section,

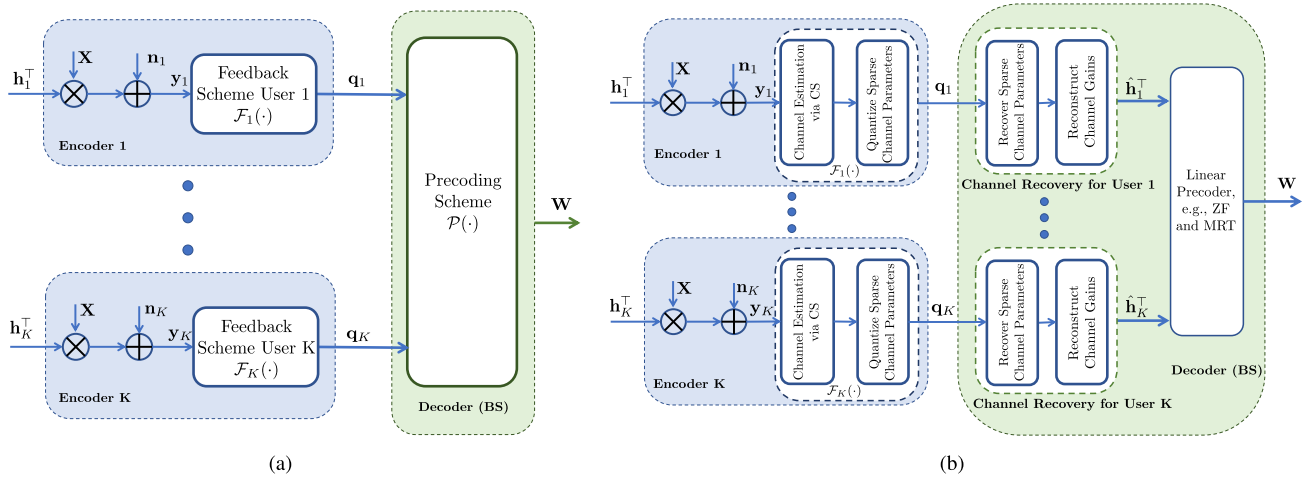


Figure 7

Comparison between end-to-end design and conventional scheme in FDD downlink precoding problem. (a) FDD downlink precoding design problem can be viewed as a distributed source coding problem in which the downlink pilots and the feedback schemes adopted at the users can be thought of as the source encoders and the precoding scheme adopted at the BS can be thought of as the decoder. (b) Conventional channel feedback scheme can be regarded as a separate source coding strategy of independent quantization of each user's channel. In the machine-learning approach, the feedback scheme at the user side and the precoding scheme at the BS side are replaced by DNNs that can be trained in an end-to-end fashion [22].

we show that a deep learning approach can be used to design a more efficient distributed source coding codebook for the FDD massive MIMO systems.

### Learning Distributed Channel Estimation and Feedback

The idea is to use DNNs to model the feedback scheme  $\{\mathcal{F}_k(\cdot)\}_{k=1}^K$  and the multiuser precoding scheme  $\mathcal{P}(\cdot)$  in Figure 7(a). The rest of this section briefly explains how we solve the overall optimization problem (10) by employing such a deep-learning framework.

As the first step of the downlink training phase, the BS sends  $L$  training pilots and the  $k$ th user observes the pilots through its channel as  $\mathbf{y}_k = \mathbf{h}_k^\top \mathbf{X} + \mathbf{n}_k$ . Since the received signal  $\mathbf{y}_k$  is a linear function of the channel  $\mathbf{h}_k$ , we can simply model it as the output of a single-layer neural network with a linear activation function in which the input is the channel  $\mathbf{h}_k$ . In this single-layer neural network, the weight matrix is the pilot  $\mathbf{X}$  and the bias vector is the noise vector  $\mathbf{n}_k$ . To enforce the total power constraint on each pilot transmission, we adopt a weight constraint under which each column of  $\mathbf{X}$  satisfies  $\|\mathbf{x}_\ell\|_2^2 \leq P_\ell$ . It is worth mentioning that such a weight constraint is often used in the machine-learning literature for regularization in order to reduce overfitting, e.g., [32]. Here, we use the weight constraint to capture the physical constraint on the downlink power level of transmit antennas of a cellular BS.

On the user side, upon receiving  $\mathbf{y}_k$ , the user seeks to summarize the useful information in  $\mathbf{y}_k$  and to feedback that information to the BS in a form of  $B$  information bits. We can simply model this process by a DNN which maps  $\mathbf{y}_k$  to feedback bits  $\mathbf{q}_k$ . To make sure that the final output of the DNN is in the form of binary bits, we use the sign activation function at the last layer of the user-side DNNs.

Finally, assuming an error-free feedback channel between each user and the BS, the BS designs precoding vectors as a function of the received feedback bits from all  $K$  users. We propose to use another DNN to map the received feedback bits  $\mathbf{q}$  to the design of the multiuser precoding matrix  $\mathbf{W}$ . To ensure that the precoding matrix designed by the DNN satisfies the total power constraint, we employ a normalization layer at the last layer of the BS-side DNN.

The overall distributed source coding strategy is designed by training the end-to-end deep-learning framework to maximize the network utility using stochastic gradient descent. But care must be taken, due to the fact that the derivative of the sign activation function is always zero, so the conventional backpropagation method cannot be directly used to train the overall network. It is possible to circumvent this difficulty by adopting the straight-through approximation in which the sign activation function is approximated by another smooth differentiable function for the backpropagation step [33]. By gradually tightening the approximation, we eventually arrive at a beamforming codebook that maps the noisy version of the channels from all the users to an optimized set of downlink beamformers.

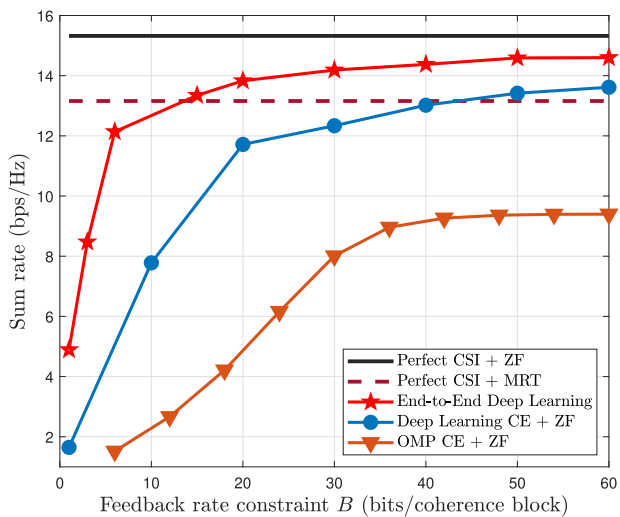


Figure 8

Sum rate versus feedback rate constraint in a two-user FDD massive MIMO system with 64 antennas, sparse channels with two dominant paths, pilot length of 8 and SNR of 10 dB [22].

## Numerical Results

We now present the performance evaluation of the end-to-end deep-learning framework in a scenario where a BS with  $M = 64$  antennas serves  $K = 2$  users in a mmWave propagation environment with two dominant paths as reported in [22].<sup>2</sup> The fading coefficient of each path is modeled by a Gaussian random variable and the corresponding angle of departure is modeled by a uniform random variable in the range of  $[-30^\circ, 30^\circ]$ . The signal-to-noise ratio (SNR)  $P_d/\sigma_0^2$  is set to 10 dB and the pilot length  $L = 8$ .

Figure 8 plots the average sum rate versus per-user feedback rate constraint  $B$ . It can be seen that the end-to-end deep-learning framework with relatively low rate feedback links (i.e., about 15 bits per user) can already outperform the maximum-ratio transmission (MRT) precoding baseline with full CSI. The MRT precoding design does not take the interuser interference into account. This shows that the trained DNN has actually learned a precoding mechanism capable of alleviating interuser interference in a multiuser FDD massive MIMO system.

Furthermore, we compare the performance of the end-to-end deep-learning framework with that of the conventional design methodology based on channel estimation followed by linear precoding schemes, such as zero forcing. For the channel estimation part of the conventional

approach, two different methods are used: 1) a compressed sensing algorithm called orthogonal matching pursuit (OMP) and 2) deep learning-based channel estimation method.

Figure 8 shows that the end-to-end deep learning framework can achieve significantly better performance as compared to the conventional channel estimation based design methodology (either when the channel estimation is implemented by OMP or by deep learning). This confirms the intuition that in practical massive MIMO systems in which the pilot length is much smaller than the number of antennas, the conventional approach of first estimating and then quantizing the sparse channel parameters is quite suboptimal. The end-to-end deep learning framework can achieve much better performance, because it is able to better explore the channel sparsity. It implicitly estimates the channel and designs the quantization codebooks jointly across the multiple users in order to maximize an overall true system objective, i.e., the sum rate in this case.

## Active Sensing for mmWave Channel Initial Alignment

Machine learning also has an important role to play in solving high-dimensional nonconvex optimization problems in sensing applications. To illustrate this point, we consider the mmWave initial alignment problem for a BS equipped with a hybrid massive MIMO architecture consisting of an analog beamformer and a low-dimensional digital beamformer. The user transmits a sequence of pilot signals; the BS makes a corresponding sequence of observations, via the analog beamformers, which it can design, but the observations reside only in the low-dimensional digital domain. The question is in which analog directions should the BS choose to observe in a sequential manner in order to obtain the most accurate channel information for a communication or sensing task of interest?

Because the sensing direction in each stage can be designed as a function of the previous observations, this is an active sensing problem for which the analytic solution is highly nontrivial and the conventional codebook-based approach is highly suboptimal [34], [35]. Specifically, [34] proposed a bisection search algorithm to gradually narrow down the angle-of-arrival (AoA) range. However, the performance of the bisection algorithm is very sensitive to the noise power, so it is suitable for the high SNR scenario only. To address this issue, [35] proposed to select the next sensing vector from a predefined codebook based on the posterior distribution of the AoA. Furthermore, [17] eliminated the codebook constraint by directly mapping the posterior distribution to the next sensing vector using a DNN. However, as the computation of posterior distribution is applicable only to the single-path channel model, the generalization of these ideas to the multipath channel is challenging.

<sup>2</sup>The code for this simulation is available at <https://github.com/foadsobrabi/DL-DSC-FDD-Massive-MIMO>.

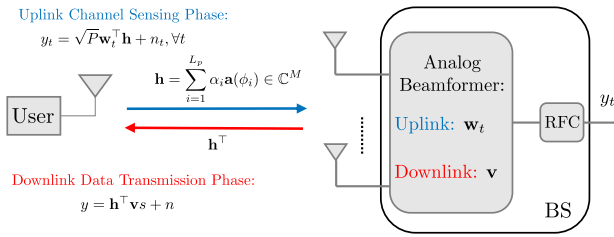


Figure 9

Active sensing for mmWave initial alignment at a BS with a single RF chain. The goal is to design the analog sensing beamformers  $\mathbf{w}_t$  adaptively as a function of the previous observations over multiple sensing stages  $t = 1, \dots, T$  for the purpose of maximizing a utility function, e.g., the eventual downlink transmission beamforming gain  $|\mathbf{h}^\top \mathbf{v}|^2$  after the sensing stage [23].

Instead, we show that an excellent solution can be obtained by training a DNN to learn the sensing direction in an end-to-end manner without needing to compute the posterior. Furthermore, we explore the active nature of the problem and show that by using a long short-term memory (LSTM)-based architecture [36], the state representation in each observation stage can be learned then used to design the sensing direction in the next stage. The results show that machine learning can offer a significant advantage over the current state-of-the-art.

### System Model and Problem Formulation

Consider a TDD mmWave system in which a BS equipped with  $M$  antennas and a single RF chain serves a single-antenna user. The user transmits a sequence of pilots to the BS, and the BS seeks to estimate the channel or to design a subsequent downlink beamformer to maximize the beamforming gain, based on the received pilots. Due to the limited RF chain, the BS can only sense the channel through an analog beamformer (or combiner), but it can design the analog beamformers sequentially to sense different directions over time. Specifically, in time frame  $t \in \{1, \dots, T\}$ , let  $\mathbf{w}_t \in \mathbb{C}^M$  denote the sensing (i.e., combining) vector with  $\|\mathbf{w}_t\|_2^2 = 1$  and let  $x_t = \sqrt{P_u}$  be the pilot symbol, then the received pilot at the BS is given by

$$y_t = \mathbf{w}_t^\top \mathbf{h} x_t + n_t \quad (11)$$

where  $n_t \sim \mathcal{CN}(0, \sigma_n^2)$  is the effective noise, and  $\mathbf{h} \in \mathbb{C}^M$  is the channel from the user to the BS. In a mmWave environment, the channel  $\mathbf{h}$  is often sparse, and can typically be modeled in the form of a multipath channel as follows:

$$\mathbf{h} = \sum_{i=1}^{L_p} \alpha_i \mathbf{a}_i(\phi_i) \quad (12)$$

where  $L_p$  is the number of paths,  $\alpha_i \in \mathcal{CN}(0, 1)$  is the fading coefficient of the  $i$ th path, and  $\phi_i \in [\phi_{\min}, \phi_{\max}]$  is the AoA of

the  $i$ th path, and  $\mathbf{a}(\phi) = [1, e^{j\pi \sin \phi}, \dots, e^{j(M-1)\pi \sin \phi}]^\top$  is the array response vector.

Assuming a fixed total number of pilot stages  $T$ , the objective in the active sensing problem is to sequentially design the sensing beamformers  $\{\mathbf{w}_t\}_{t=1}^T$  to maximize some utility function, i.e.,  $\mathcal{J}(\boldsymbol{\theta}, \mathbf{v})$ , where  $\boldsymbol{\theta} = \{\alpha_i, \phi_i\}_{i=1}^{L_p}$  contains all the channel parameters and  $\mathbf{v} \in \mathbb{C}^M$  is the parameter to be designed or estimated after receiving all the pilots. For example, as illustrated in Figure 9,  $\mathbf{v} \in \mathbb{C}^M$  can be the subsequent downlink data transmission beamformer and the goal can be to maximize the beamforming gain, i.e.,  $\mathcal{J}(\boldsymbol{\theta}, \mathbf{v}) := |\mathbf{h}^\top \mathbf{v}|^2$ . In other applications, e.g., AoA-based localization, we might be interested in estimating the AoAs of the multipath channel, i.e.,  $\mathcal{J}(\boldsymbol{\theta}, \mathbf{v}) := -\sum_{i=1}^{L_p} (\hat{\phi}_i - \phi_i)^2$ , where  $\mathbf{v} := [\hat{\phi}_1, \dots, \hat{\phi}_{L_p}]^\top$ . The key characteristic of such problems is that the sensing vector  $\mathbf{w}_{t+1}$  can be designed based on the historical observations at stage  $t$ . Accordingly, the overall problem can be formulated as

$$\text{maximize } E[\mathcal{J}(\boldsymbol{\theta}, \mathbf{v})] \quad (13a)$$

$$\{\mathcal{G}_t(\cdot, \cdot)\}_{t=0}^{T-1}, \mathcal{F}(\cdot, \cdot)$$

$$\text{subject to } \mathbf{w}_{t+1} = \mathcal{G}_t(\mathbf{y}_{1:t}, \mathbf{w}_{1:t}), t = 0, \dots, T-1 \quad (13b)$$

$$\mathbf{v} = \mathcal{F}(\mathbf{y}_{1:T}, \mathbf{w}_{1:T}) \quad (13c)$$

where  $\mathcal{G}_t : \mathbb{R}^t \times \mathbb{R}^{LM} \rightarrow \mathbb{R}^M$  is the adaptive sensing strategy adopted by the BS in time frame  $t$  and  $\mathcal{F} : \mathbb{R}^T \times \mathbb{R}^{TM} \rightarrow \mathbb{R}^V$  is the function for designing the vector  $\mathbf{v}$ .

The active sensing problem (13) is challenging to solve, because both the active sensing strategy  $\{\mathcal{G}_t(\cdot, \cdot)\}_{t=0}^{T-1}$  and the mapping  $\mathcal{F}(\cdot, \cdot)$  are functions in high-dimensional spaces. Moreover, the input dimension of the function  $\mathcal{G}_t(\cdot, \cdot)$  increases as the number of sensing stages increases, making the sensing strategy particularly difficult to design when  $T$  is large. The conventional strategies are codebook based. For example, a hierarchical beamforming codebook [34] can be designed based on the principle of bisection as mentioned before. A posterior matching based approach for sequentially selecting the appropriate analog combiners from the hierarchical codebook is proposed in [35]. But these approaches are by no means optimal and are restricted to single-path channels. For the multipath channel, nonadaptive sensing strategies which exploit the channel sparsity are usually adopted [34].

In this section, we show that instead of using a model-based approach, a codebook-free data-driven approach can be used to design the analog combiners to sense a multipath channel. Specifically, the sequential nature of the problem suggests that a recurrent neural network (RNN) is an appropriate network architecture. We show that a deep active sensing framework based on the LSTM network, which is a variation of RNN, can be used to efficiently solve the active sensing problem (13).

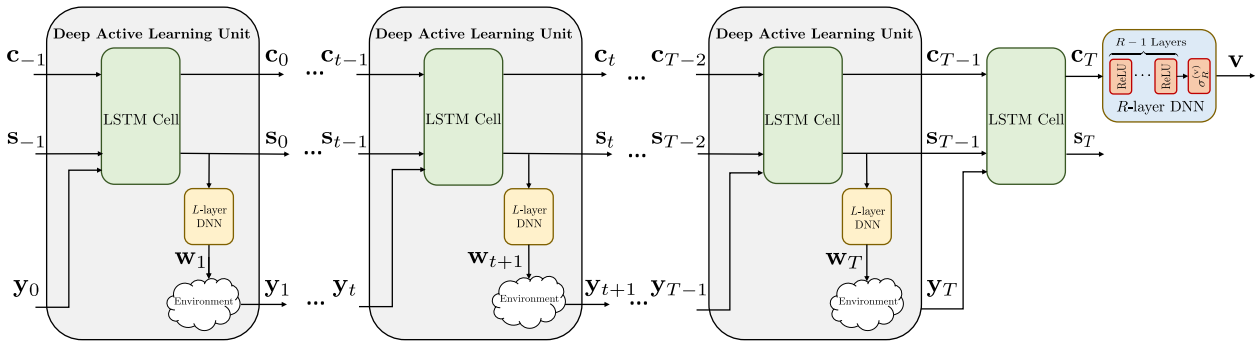


Figure 10

LSTM-based sequential learning architecture for solving an active sensing problem in mmWave initial alignment, in which the LSTM cells aim to summarize the system state based on the observations made so far, and a DNN is used to produce the analog combiner to be used in the next sensing stage [23].

### Learning Active Sensing Strategy

The proposed active sensing framework is shown in Figure 10. It consists of  $T$  deep active learning units, corresponding to  $T$  different sensing stages. Each active sensing stage is designed based on an LSTM cell and a fully connected DNN. Specifically, in the  $t$ th active sensing stage, the LSTM cell takes the previous cell state vector  $\mathbf{c}_{t-1}$ , the previous hidden state vector  $\mathbf{s}_{t-1}$ , and the current measurement  $y_t$  as input, and outputs the next cell state vector  $\mathbf{c}_t$  and hidden state vector  $\mathbf{s}_t$ . The LSTM cell is capable of automatically summarizing the previous observations into state vectors. At each stage, we use the fully connected DNN to map the hidden state vector  $\mathbf{s}_t$  to the sensing vector  $\mathbf{w}_t$ . After receiving the last pilot symbol  $y_T$ , the LSTM cell updates its cell state to  $\mathbf{c}_T$ , which is then mapped to the desired parameter  $\mathbf{v}$  using another DNN. This active sensing framework is trained end-to-end to maximize the objective function in (13), with neural network weights tied together across the sensing stages. Such an end-to-end training approach enables the learning of an active sensing policy that accounts for the ultimate design or estimation objective after the  $T$  sensing stages.

### Numerical Results

To illustrate the performance of the active sensing framework, we now present the simulation results<sup>3</sup> in [23] for a downlink beamforming gain maximization problem in a setting with  $M = 64$ ,  $L_p = 3$ , and  $\text{SNR} = 0\text{dB}$ . The AoAs are randomly generated from the range  $[-60^\circ, 60^\circ]$ . We compare the proposed active sensing method with the channel estimation based approach as well as a design using a DNN to map the received pilots to the beamforming vector, but the sensing beamformers are fixed, either at random or learned from the statistics of the channel. In Figure 11, we see that the deep learning

methods outperform the channel estimation based method with OMP. This shows the benefit of bypassing channel estimation. The active sensing method achieves better performance than deep learning with fixed sensing vectors. This shows the benefits of adaptive sensing and the ability of the LSTM network to optimize the sensing vectors.

To see where the performance gain comes from, we examine the output of the LSTM framework for an AoA estimation problem in a single-path channel, and plot the posterior distribution of the AoA at each stage  $t$  as well as the array response of the sensing vectors designed by the LSTM and the DNN. As can be seen from Figure 12, the posterior distribution gradually converges to a distribution concentrated at the true AoA  $\phi = 25.82^\circ$ . In the

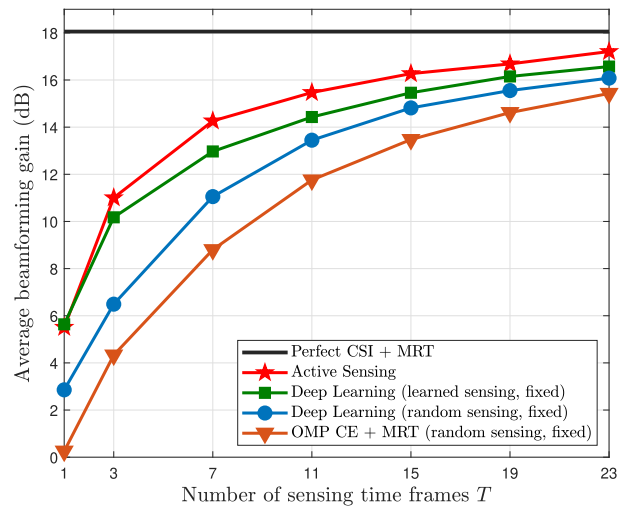


Figure 11

Average beamforming gain in dB versus the number of sensing stages  $T$  for different methods after beam alignment in a TDD mmWave system with  $M_r = 64$ ,  $\text{SNR} = 0\text{ dB}$ ,  $L_p = 3$ , and  $\phi_1, \phi_2, \phi_3 \in [-60^\circ, 60^\circ]$  [23].

<sup>3</sup>The code for this simulation is available at <https://github.com/foadsohrabi/DL-ActiveSensing>.



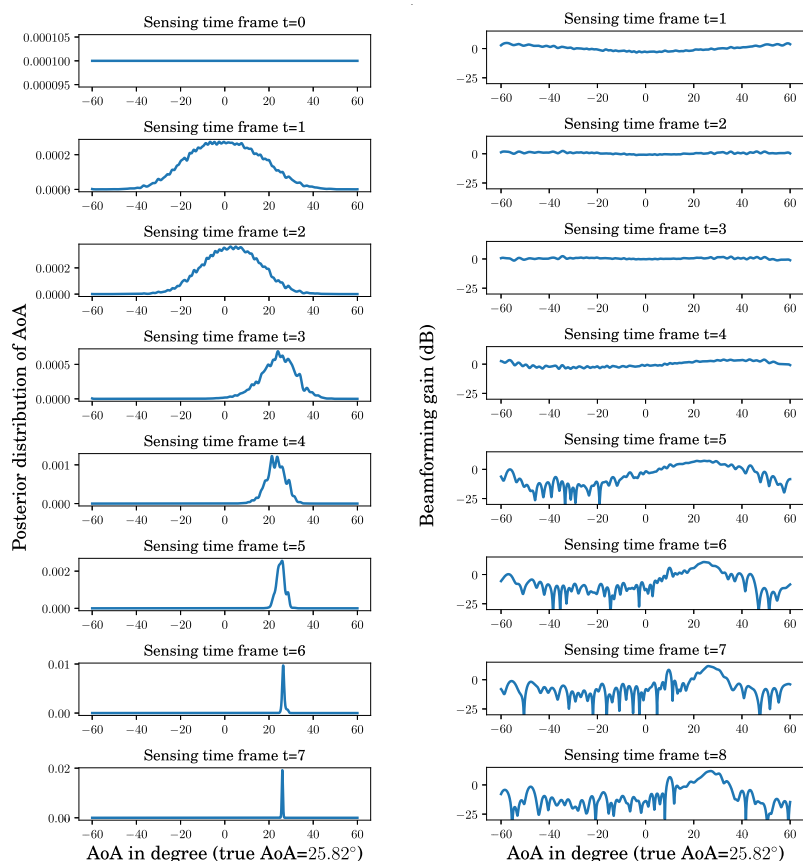


Figure 12

Posterior distributions of the AoA (left) and the beamforming patterns of the sensing vectors (right) learned from the proposed active sensing framework over eight stages in a mmWave alignment problem for a single-path channel where  $\text{SNR} = 0$  dB,  $M_r = 64$ , and  $T = 12$  [23].

meanwhile, the array response of the sensing vectors designed by the active sensing framework is relatively flat across the angles at the beginning, indicating that it is exploring all directions in searching for the AoA, but gradually narrows down to around the direction of the true AoA as the sensing operation progresses. This shows that the active sensing framework indeed learns a meaningful sensing strategy. It is capable of quickly converging to the true AoA. It is remarkable that although finding the truly optimal sensing vectors is extremely difficult to do computationally, the LSTM framework is able to learn an excellent sensing strategy based on training over millions of channel instances.

## Standardization Impact

While the experimental results reported in this article are still generated using widely accepted wireless channel propagation models so the proposed framework should be regarded as a proof-of-concept rather than as field-tested, the standardization of the wireless communications bodies have recognized the potential for using machine-learning techniques in future

cellular networks and have taken steps toward standardizing the communication protocols between the BS and the users in order to enable learning-based system-level optimization.

Specifically, the 3rd Generation Partnership Project (3GPP) has recognized channel estimation and feedback, mmWave beam management, and positioning as the three initial areas where machine learning can have a significant impact [37]. One of the target scenarios related to the CSI feedback enhancement that 3GPP aims to study is CSI compression and feedback for FDD massive MIMO systems where a wireless device has already obtained the entire high-dimensional channel matrix and it needs to compress and feedback this CSI to the BS. Such a CSI acquisition process can be modeled by an autoencoder consisting of a DNN encoder at the device and a DNN decoder at the BS. In particular, the DNN encoder first maps the high-dimensional channel to a low-dimensional quantized signal, then the compressed signal is sent to the BS via the uplink feedback channel, and finally, the BS reconstructs the channel by using the DNN decoder. The goal of DNNs here is to capture the spatial-domain and frequency-domain correlations in the channel matrix, so a

convolutional neural network is an excellent candidate as an autoencoder. Preliminary results reported by the different companies suggest that machine learning can outperform the existing 5G codebook-based CSI compression methods, e.g., [38]. This use case is closely related to the CSI estimation and feedback problem studied in an earlier section of this article.

The second use case is about beam management procedure (e.g., alignment) to find the best transmit-receive beam pair. The conventional practical beam management is based on exhaustive beam sweeping. While these linear beam search strategies lead to excellent performance, they suffer from significant time delay and power consumption issues. To address these concerns, sparse beam sweeping has been introduced in 3GPP [39] in which a beam pair is selected by employing multiple-stage beam narrowing strategies. However, the existing algorithms developed for sparse beam sweeping are quite suboptimal, especially in higher frequency bands and with high-mobility users. Data-driven methods, on the other hand, can utilize the training datasets to construct a mapping from sparse beam measurements to the best beam pair. This use case is closely related to the initial beam alignment problem addressed in an earlier section using a deep active sensing approach. The use case can actually be thought of as a nonactive version of the problem.

Accurate positioning is a crucial component in several 5G industrial Internet of Things applications, such as smart factories, and is another promising area for data-driven designs. The traditional model-based positioning relies on explicit mappings from timing/angle measurements to the position of the user. These mappings are effective when there are multiple line-of-sight paths between the target user and different reception points of the BS. But, practical application scenarios usually have to deal with nonline-of-sight conditions. In these scenarios, the traditional model-based approach is not always feasible. Learning-based methods are promising solutions for these difficult positioning tasks since they can easily learn a good mapping from the radio measurements to the position by discerning patterns in the available training datasets. Preliminary results already show significant positioning accuracy enhancement over the conventional methods, e.g., [40]. While this article has not addressed the localization problem specifically, the techniques presented are quite applicable to localization [41].

## Conclusion

In conclusion, the modern machine-learning approach is opening new opportunities in the optimization of physical-layer wireless communication systems. It challenges the conventional wisdom of always first modeling the channel, then optimizing wireless system design given the estimated channel. This article shows that much can be gained by bypassing explicit channel modeling, by designing the overall system in

an end-to-end manner, and by formulating and solving optimization problems in a data-driven fashion.

## Acknowledgments

This work was supported by Natural Sciences and Engineering Research Council (NSERC) via the Canada Research Chairs program.

## References

- [1] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, no. 5, pp. 359–366, 1989, doi: [10.1016/0893-6080\(89\)90020-8](https://doi.org/10.1016/0893-6080(89)90020-8).
- [2] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Trans. Cogn. Commun. Netw.*, vol. 3, no. 4, pp. 563–575, Dec. 2017, doi: [10.1109/TCCN.2017.2758370](https://doi.org/10.1109/TCCN.2017.2758370).
- [3] Z. Qin, H. Ye, G. Y. Li, and B.-H. F. Juang, "Deep learning in physical layer communications," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 93–99, Apr. 2019, doi: [10.1109/MWC.2019.1800601](https://doi.org/10.1109/MWC.2019.1800601).
- [4] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Artificial neural networks-based machine learning for wireless networks: A tutorial," *IEEE Commun. Surv. Tut.*, vol. 21, no. 4, pp. 3039–3071, Oct.–Dec. 2019, doi: [10.1109/COMST.2019.2926625](https://doi.org/10.1109/COMST.2019.2926625).
- [5] Y. C. Eldar, A. Goldsmith, D. Gündüz, and H. V. Poor, *Machine Learning and Wireless Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2022, doi: [10.1017/9781108966559](https://doi.org/10.1017/9781108966559).
- [6] E. Bourtsoulatzé, D. B. Kurka, and D. Gündüz, "Deep joint source-channel coding for wireless image transmission," *IEEE Trans. Cogn. Commun. Netw.*, vol. 5, no. 3, pp. 567–579, Sep. 2019, doi: [10.1109/TCCN.2019.2919300](https://doi.org/10.1109/TCCN.2019.2919300).
- [7] E. Nachmani et al., "Deep learning methods for improved decoding of linear codes," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 119–131, Feb. 2018, doi: [10.1109/JSTSP.2017.2788405](https://doi.org/10.1109/JSTSP.2017.2788405).
- [8] H. Kim, S. Oh, and P. Viswanath, "Physical layer communication via deep learning," *IEEE J. Sel. Areas Commun.*, vol. 1, no. 1, pp. 5–18, May 2020, doi: [10.1109/JSAIT.2020.2991562](https://doi.org/10.1109/JSAIT.2020.2991562).
- [9] F. A. Aoudia and J. Hoydis, "Waveform learning for next-generation wireless communication systems," *IEEE Trans. Commun.*, vol. 70, no. 6, pp. 3804–3817, Jun. 2022, doi: [10.1109/TCOMM.2022.3164060](https://doi.org/10.1109/TCOMM.2022.3164060).
- [10] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 114–117, Feb. 2018, doi: [10.1109/LWC.2017.2757490](https://doi.org/10.1109/LWC.2017.2757490).
- [11] N. Farsad and A. Goldsmith, "Neural network detection of data sequences in communication systems," *IEEE Trans. Signal Process.*, vol. 66, no. 21, pp. 5663–5678, Nov. 2018, doi: [10.1109/TSP.2018.2868322](https://doi.org/10.1109/TSP.2018.2868322).

- [12] F. Sahrabi, T. Jiang, and W. Yu, "Learning progressive distributed compression strategies from local channel state information," *IEEE J. Sel. Topics Signal Process.*, vol. 16, no. 3, pp. 573–584, Apr. 2022, doi: [10.1109/JSTSP.2022.3158820](https://doi.org/10.1109/JSTSP.2022.3158820).
- [13] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos, "Learning to optimize: Training deep neural networks for interference management," *IEEE Trans. Signal Process.*, vol. 66, no. 20, pp. 5438–5453, Oct. 2018, doi: [10.1109/TSP.2018.2866382](https://doi.org/10.1109/TSP.2018.2866382).
- [14] W. Cui, K. Shen, and W. Yu, "Spatial deep learning for wireless scheduling," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1248–1261, Jun. 2019, doi: [10.1109/JSAC.2019.2904352](https://doi.org/10.1109/JSAC.2019.2904352).
- [15] M. Lee, G. Yu, and G. Y. Li, "Graph embedding based wireless link scheduling with few training samples," *IEEE Trans. Wireless Commun.*, vol. 20, no. 4, pp. 2282–2294, Apr. 2021, doi: [10.1109/TWC.2020.3040983](https://doi.org/10.1109/TWC.2020.3040983).
- [16] Y. Shen, Y. Shi, J. Zhang, and K. B. Letaief, "Graph neural networks for scalable radio resource management: Architecture design and theoretical analysis," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 1, pp. 101–115, Jan. 2020, doi: [10.1109/JSAC.2020.3036965](https://doi.org/10.1109/JSAC.2020.3036965).
- [17] F. Sahrabi, Z. Chen, and W. Yu, "Deep active learning approach to adaptive beamforming for mmWave initial alignment," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2347–2360, Aug. 2021, doi: [10.1109/JSAC.2021.3087234](https://doi.org/10.1109/JSAC.2021.3087234).
- [18] Z. Zhang, T. Jiang, and W. Yu, "Learning based user scheduling in reconfigurable intelligent surface assisted multiuser downlink," *IEEE J. Sel. Topics Signal Process.*, vol. 16, no. 5, pp. 1026–1039, Aug. 2022, doi: [10.1109/JSTSP.2022.3178213](https://doi.org/10.1109/JSTSP.2022.3178213).
- [19] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "Deep learning-based channel estimation for beamspace mmWave massive MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 852–855, Oct. 2018, doi: [10.1109/LWC.2018.2832128](https://doi.org/10.1109/LWC.2018.2832128).
- [20] D. Neumann, T. Wiese, and W. Utschick, "Learning the MMSE channel estimator," *IEEE Trans. Signal Process.*, vol. 66, no. 11, pp. 2905–2917, Jun. 2018, doi: [10.1109/TSP.2018.2799164](https://doi.org/10.1109/TSP.2018.2799164).
- [21] T. Jiang, H. V. Cheng, and W. Yu, "Learning to reflect and to beamform for intelligent reflecting surface with implicit channel estimation," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 7, pp. 1931–1945, Jul. 2021, doi: [10.1109/JSAC.2021.3078502](https://doi.org/10.1109/JSAC.2021.3078502).
- [22] F. Sahrabi, K. M. Attiah, and W. Yu, "Deep learning for distributed channel feedback and multiuser precoding in FDD massive MIMO," *IEEE Trans. Wireless Commun.*, vol. 20, no. 7, pp. 4044–4057, Jul. 2021, doi: [10.1109/TWC.2021.3055202](https://doi.org/10.1109/TWC.2021.3055202).
- [23] F. Sahrabi, T. Jiang, W. Cui, and W. Yu, "Active sensing for communications by learning," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 6, pp. 1780–1794, Jun. 2022, doi: [10.1109/JSAC.2022.3155496](https://doi.org/10.1109/JSAC.2022.3155496).
- [24] 3GPP TR 38.901, "5G; Study on channel model for frequencies from 0.5 to 100 GHz," 3GPP, Tech. Rep., Dec. 2018, version 14.3.0 (Rel. 14).
- [25] J. Nickolls and W. J. Dally, "The GPU computing era," *IEEE Micro*, vol. 30, no. 2, pp. 56–69, Mar./Apr. 2010, doi: [10.1109/MM.2010.41](https://doi.org/10.1109/MM.2010.41).
- [26] M. Abadi et al., "TensorFlow: A system for large-scale machine learning," in *Proc. USENIX Symp. Oper. Syst. Des. Implementation*, 2016, pp. 265–283.
- [27] M. Di Renzo et al., "Smart radio environments empowered by reconfigurable intelligent surfaces: How it works, state of research, and road ahead," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 11, pp. 2450–2525, Jul. 2019, doi: [10.1109/JSAC.2020.3007211](https://doi.org/10.1109/JSAC.2020.3007211).
- [28] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?," in *Proc. Int. Conf. Learn. Representation*, 2019.
- [29] D. J. Love, R. W. Heath, V. K. N. Lau, D. Gesbert, B. D. Rao, and M. Andrews, "An overview of limited feedback in wireless communication systems," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 8, pp. 1341–1365, Oct. 2008, doi: [10.1109/JSAC.2008.081002](https://doi.org/10.1109/JSAC.2008.081002).
- [30] Z. Gao, L. Dai, Z. Wang, and S. Chen, "Spatially common sparsity based adaptive channel estimation and feedback for FDD massive MIMO," *IEEE Trans. Signal Process.*, vol. 63, no. 23, pp. 6169–6183, Dec. 2015, doi: [10.1109/TSP.2015.2463260](https://doi.org/10.1109/TSP.2015.2463260).
- [31] X. Rao and V. K. N. Lau, "Distributed compressive CSIT estimation and feedback for FDD multi-user massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3261–3271, Jun. 2014, doi: [10.1109/TSP.2014.2324991](https://doi.org/10.1109/TSP.2014.2324991).
- [32] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," Jul. 2012. *arXiv:1207.0580*.
- [33] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "Hierarchical multiscale recurrent neural networks," in *Proc. Int. Conf. Learn. Representation*, 2017.
- [34] A. Alkhateeb, O. El Ayach, G. Leus, and R. W. Heath, "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 831–846, Oct. 2014, doi: [10.1109/JSTSP.2014.2334278](https://doi.org/10.1109/JSTSP.2014.2334278).
- [35] S. Chiu, N. Ronquillo, and T. Javidi, "Active learning and CSI acquisition for mmWave initial alignment," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 11, pp. 2474–2489, Nov. 2019, doi: [10.1109/JSAC.2019.2933967](https://doi.org/10.1109/JSAC.2019.2933967).
- [36] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: [10.1162/neco.1997.9.8.1735](https://doi.org/10.1162/neco.1997.9.8.1735).
- [37] RP-213599, "New SI: Study on artificial intelligence (AI)/ machine learning (ML) for NR air interface," 3GPP, Tech. Rep., Specification Group, 2021, RAN Meeting#94e.

- [38] R1-2204238, Apple, Inc., "Evaluation for AI/ML based CSI feedback enhancement," 3GPP, Tech. Rep., Specification Group, May 2022, RAN1#109-e.
- [39] R1-2203142, Huawei, HiSilicon, "Evaluation on AI/ML for beam management," 3GPP, Tech. Rep., Specification Group, May 2022, RAN1#109-e.
- [40] R1-2203901, Samsung, "Discussion on the evaluation on AI/ML for positioning accuracy enhancement," 3GPP, Tech. Rep., Specification Group, May 2022, RAN1#109-e.
- [41] Z. Zhang, T. Jiang, and W. Yu, "Active sensing for localization with reconfigurable intelligent surface," 2022, preprint.



**Wei Yu** (Fellow, IEEE) received the B.A.Sc. degree in computer engineering and mathematics from the University of Waterloo, Waterloo, ON, Canada, and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, USA. He is currently a Professor and Canada Research Chair in Information Theory and Wireless Communications in the Electrical and Computer

Engineering Department, University of Toronto, Toronto, ON, Canada.

Dr. Yu is a Fellow of the Canadian Academy of Engineering. He was the recipient of the IEEE Marconi Prize Paper Award in Wireless Communications in 2019, the IEEE Communications Society Award for Advances in Communication in 2019, the IEEE Signal Processing Society Best Paper Award in 2008, 2017, and 2021, and the IEEE Communications Society Best Tutorial Paper Award in 2015. He was the President of the IEEE Information Theory Society in 2021.



**Foad Sohrabi** (Member, IEEE) received the B.A.Sc. degree from the University of Tehran, Tehran, Iran, in 2011, the M.A.Sc. degree from McMaster University, Hamilton, ON, Canada, in 2013, and the Ph.D. degree from the University of Toronto, Toronto, ON, in 2018, all in electrical and computer engineering. From March 2018 to November 2021, he was a Postdoctoral Fellow with the

University of Toronto. From October 2021 to August 2022, he was a Senior Technical Staff with Ofinno, Reston, Virginia, USA. Since August 2022, he has been a Member of Technical Staff with Nokia Bell Labs, NJ, USA.

His research interests include MIMO communications, optimization theory, wireless communications, signal processing, and machine learning. Dr. Sohrabi was the recipient of the IEEE Signal Processing Society Best Paper Award in 2017.



**Tao Jiang** (Graduate Student Member, IEEE) received the B.E. degree in telecommunications engineering from Xidian University, Xi'an, China, in 2017, and the M.E. degree in computer science from ShanghaiTech University, Shanghai, China, in 2020. He is currently working toward the Ph.D. degree with the University of Toronto, Toronto, ON, Canada.

His research interests include optimization theory, machine learning, and wireless communications.