

Secure Precoding in MIMO-NOMA: A Deep Learning Approach

Jordan Pauls¹, *Student Member, IEEE*, and Mojtaba Vaezi¹, *Senior Member, IEEE*

Abstract—A novel signaling design for secure transmission over two-user multiple-input multiple-output non-orthogonal multiple access channel using deep neural networks (DNNs) is proposed. The goal of the DNN is to form the covariance matrix of users' signals such that the message of each user is transmitted reliably while being confidential from its counterpart. The proposed DNN linearly precodes each user's signal before superimposing them and achieves near-optimal performance with significantly lower run time. Simulation results show that the proposed models reach about 98% of the secrecy capacity rates. The spectral efficiency of the DNN precoder is much higher than that of existing analytical linear precoders,—e.g., generalized singular value decomposition—and its on-the-fly complexity is several times less than the existing iterative methods.

Index Terms—Deep learning, DNN, MIMO, NOMA, physical layer security, wiretap, precoding, covariance, GSVD.

I. INTRODUCTION

NON-ORTHOGONAL multiple access (NOMA) is a promising candidate for connecting massively increasing devices to fifth-generation and beyond wireless networks [1]. NOMA is the optimal transmission strategy for both the single-input, single-output (SISO) and multiple-input, multiple-output (MIMO) cases in a single-cell network. In the SISO case, superposition coding at the transmitter with successive interference cancellation at the receiver is optimal. In contrast, in the MIMO case, *dirty-paper coding* (DPC) is the optimal solution. Nonetheless, in both cases, the base station (BS) *broadcasts* a superimposed signal of multiple users. This makes secure communications challenging in the presence of adversarial users as signals can be eavesdropped on by such users.

Physical layer security enables the exchange of confidential messages over a wireless medium in the presence of internal or unauthorized eavesdroppers [2]. Specifically, in two-user MIMO-NOMA networks, both users can transmit their messages concurrently and confidentially via *secret dirty-paper coding* (S-DPC) [3]. While S-DPC is the most spectral efficient precoding for the two-user MIMO-NOMA, it is excessively complex for practical uses. Alternatively, S-DPC region can be achieved by *linear precoding* and power allocation schemes. In the past years, various linear precoding schemes have been introduced [4]–[7]. *Generalized singular value decomposition* (GSVD)-based precoder [4] is a fast analytical precoder, but it falls short of getting the capacity region when the users have a

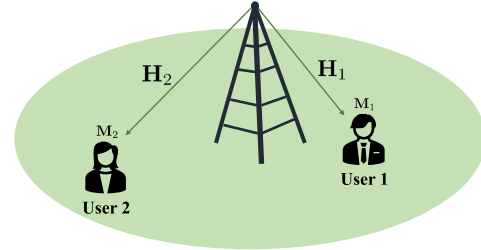


Fig. 1. Two-user MIMO-NOMA with confidential messages. M_1 and M_2 are the messages for user 1 and user 2, respectively. Each user is intended to decode its own message but not the other one.

single antenna. Also, weighted sum-rate maximization [5] and power-splitting [6] approaches are still too time-consuming to be used in practice. More accurately, these approaches require much higher time than the *coherence time* of wireless channels which is about a few milliseconds [8].

This letter exploits deep learning (DL) to design the covariance matrices of the channel input vectors—or equivalently, to design precoding and power allocation matrices—for secure MIMO-NOMA transmission. Embedding DL into the mobile and wireless networks is well justified in various cases, e.g., when closed-form solutions require poor approximation or the complexity of existing techniques is high [9]–[11].

We leverage supervised deep neural networks (DNNs) for secure communication for the two-user MIMO-NOMA channel (see Fig. 1), resulting in a significantly faster solution while almost reaching the spectral efficiency of the S-DPC. To fulfill the task, we first decompose the two-user MIMO-NOMA with confidential messages into two wiretap channels [6] and use a wiretap channel solution for generating and labeling the training data set. We then build and train DNN models that learn to approximate the function mapping channel matrices, base station power, and required secrecy rate of the users to the covariance matrix of the channel input vector for each user. Simulation results prove the efficacy of the developed model since the proposed DL-based precoding scheme has near-optimal performance and outperforms GSVD-based precoding with a large margin. It also brings mapping time below the coherence time of the wireless channel. Therefore, the channel input signal can be designed before the channel coefficients become stale, which is crucial in practice.

II. SYSTEM MODEL AND EXISTING SOLUTIONS

Consider a single-cell two-user MIMO-NOMA network, as shown in Fig. 1. Assume that the transmitter, user 1, and user 2 are equipped with n_t , n_1 , and n_2 antennas, respectively. The transmitter wishes to send two messages M_1 and M_2 to user 1, and user 2, respectively. When PHY security is a concern, each

Manuscript received October 5, 2021; accepted October 12, 2021. Date of publication October 15, 2021; date of current version January 10, 2022. The associate editor coordinating the review of this article and approving it for publication was B. Makki. (Corresponding author: Mojtaba Vaezi.)

The authors are with the Department of Electrical and Computer Engineering, Villanova University, Villanova, PA 19085 USA (e-mail: jpauls1@villanova.edu; mvaezi@villanova.edu).

Digital Object Identifier 10.1109/LWC.2021.3120594

message must be kept *confidential* from the other user [12].¹ That is, user 1 should not be able to decode M_2 and vice versa. As such, the transmitter *securely* encodes M_1 and M_2 to codewords $\mathbf{x}_1 \in \mathbb{R}^{n_1 \times 1}$ and $\mathbf{x}_2 \in \mathbb{R}^{n_2 \times 1}$, *superimposes* them $\mathbf{x} = \mathbf{x}_1 + \mathbf{x}_2$, and *broadcasts* \mathbf{x} [12], [13]. Let $\mathbf{H}_1 \in \mathbb{R}^{n_1 \times n_t}$ and $\mathbf{H}_2 \in \mathbb{R}^{n_2 \times n_t}$ be the channel matrices corresponding to user 1 and user 2. Then, the received signals at user 1 and user 2, respectively, can be represented as

$$\mathbf{y}_1 = \mathbf{H}_1(\mathbf{x}_1 + \mathbf{x}_2) + \mathbf{z}_1, \quad (1a)$$

$$\mathbf{y}_2 = \mathbf{H}_2(\mathbf{x}_1 + \mathbf{x}_2) + \mathbf{z}_2, \quad (1b)$$

in which $\mathbf{z}_1 \in \mathbb{R}^{n_1 \times 1}$ and $\mathbf{z}_2 \in \mathbb{R}^{n_2 \times 1}$ are two independent identically distributed (i.i.d) Gaussian noise vectors with mean zero and identity covariance matrices.

A. Secrecy Capacity Region

The secrecy capacity region of this channel under a matrix constraint on the covariance matrix of the input \mathbf{x} is proved in [12, Th. 1]. However, in practical MIMO systems, a total power constraint P at the transmitter is more common. Under this assumption, using [3, Corollary 1], the *secrecy capacity region* of this channel can be represented as

$$R_1 \leq \frac{1}{2} \log |\mathbf{I}_1 + \mathbf{H}_1 \mathbf{Q}_1 \mathbf{H}_1^T| - \frac{1}{2} \log |\mathbf{I}_2 + \mathbf{H}_2 \mathbf{Q}_1 \mathbf{H}_2^T|, \quad (2a)$$

$$R_2 \leq \frac{1}{2} \log \left| \mathbf{I}_2 + \frac{\mathbf{H}_2 \mathbf{Q}_2 \mathbf{H}_2^T}{\mathbf{I}_2 + \mathbf{H}_2 \mathbf{Q}_1 \mathbf{H}_2^T} \right| - \frac{1}{2} \log \left| \mathbf{I}_1 + \frac{\mathbf{H}_1 \mathbf{Q}_2 \mathbf{H}_1^T}{\mathbf{I}_1 + \mathbf{H}_1 \mathbf{Q}_1 \mathbf{H}_1^T} \right|, \quad (2b)$$

$$\text{s.t.} \quad \text{tr}(\mathbf{Q}_1) + \text{tr}(\mathbf{Q}_2) \leq P, \quad \mathbf{Q}_1 \succeq \mathbf{0}, \quad \mathbf{Q}_2 \succeq \mathbf{0}, \quad (2c)$$

in which R_i , $i \in \{1, 2\}$ is the secure achievable rate at user i , \mathbf{I}_i is an identity matrix of size n_i , and \mathbf{Q}_i is the covariance matrix of \mathbf{x}_i . By definition, $\mathbf{Q}_i = \mathbb{E}(\mathbf{x}_i \mathbf{x}_i^T)$ where $\mathbb{E}(\cdot)$ denotes expectation, and thus, \mathbf{Q}_i is positive semidefinite, i.e., $\mathbf{Q}_i \succeq \mathbf{0}$. Further, since total transmit power cannot be higher than P , we have $P \geq \mathbb{E}(\mathbf{x} \mathbf{x}^T) = \text{tr}(\mathbf{Q}_1) + \text{tr}(\mathbf{Q}_2)$.

B. Existing Solutions

Although the secure capacity region of the two-user MIMO-NOMA channel is given in (2), it is still unknown how to analytically form \mathbf{Q}_1 and \mathbf{Q}_2 to achieve the capacity region. This is because the right-hand side expressions both in (2a) and (2b) are non-convex, and thus, the corresponding optimization problems are challenging. Early works like [12] use an exhaustive search over all possible \mathbf{Q}_1 and \mathbf{Q}_2 , satisfying the constraints in (2c). Such an approach is, however, prohibitively complex for practical systems. Later, Fakoorian and Swindlehurst [4] proposed a GSVD-based precoder for this problem. The rate region of this method is, however, far from the capacity region when n_1 and/or n_2 are small numbers and, in particular, when the users have a single antenna. Intriguingly, such cases are very prevalent and thus important in practice.

¹In information theory, this channel is known as the MIMO broadcast channel (BC) with two confidential messages [12].

Lately, another approach was proposed to solve this problem [6] whose achievable rate region is very close to the optimal solution for any number of antennas at each node. The main observation is that the two-users MIMO-NOMA channel can be decomposed into two MIMO *wiretap channels* by splitting power between the two users. Then, the associated optimization problems are solved one at a time. Despite the fact that the rate region obtained from this solution is very close to the secure capacity region of the channel, this solution incurs an unacceptable amount of delay, which hinders it from being used for practical systems. To be specific, finding optimal \mathbf{Q}_1 and \mathbf{Q}_2 could take several hundred milliseconds, [6] where as the *coherence time* of the wireless channel can be as small as a few milliseconds [8]. That is, the solution assumes that the channel is constant for several hundred milliseconds while it changes much faster in practice.

In this letter, we propose a DL-based signaling design to approach the secure capacity of the MIMO-NOMA channel within a practically acceptable delay. This is obtained at the expense of a slightly smaller achievable rate region. In the following, we describe the structure of the DNN, the training process, and the test results.

III. DEEP LEARNING-BASED SOLUTION

In this section, we build a supervised deep learning model to determine suitable covariance matrices for (2). Put differently, we describe signaling design (precoding and power allocation) for secure transmission over the MIMO-NOMA channel. We present a DNN structure that includes data generation, training methodology, DNN structure, and hyper-parameters.

A. Data Generation and Labeling

In *supervised learning*, labeled data is used for drawing inferences, i.e., for classification, regression, or approximation. In this letter, labeled training data is used to build predictive models to learn the functions mapping inputs to outputs, i.e., to approximate optimal signaling in the secure MIMO-NOMA by regression. Specifically, for each set of \mathbf{H}_1 and \mathbf{H}_2 we find \mathbf{Q}_1 and \mathbf{Q}_2 and use them for training a DNN and determining a model.

We decompose the secure MIMO-NOMA channel into two MIMO wiretap channels [6]. For $\alpha \in [0, 1]$, we allot αP and $(1 - \alpha)P$ to user 1 and user 2, respectively. Then, we find the covariance matrix \mathbf{Q}_1 from (2a), i.e.,

$$\mathbf{Q}_1^* = \arg \max_{\mathbf{Q}_1} \frac{1}{2} \log \frac{|\mathbf{I}_1 + \mathbf{H}_1 \mathbf{Q}_1 \mathbf{H}_1^T|}{|\mathbf{I}_2 + \mathbf{H}_2 \mathbf{Q}_1 \mathbf{H}_2^T|}, \quad (3a)$$

$$\text{s.t.} \quad \mathbf{Q}_1 \succeq \mathbf{0}, \quad \text{tr}(\mathbf{Q}_1) \leq P_1 = \alpha P. \quad (3b)$$

Now, this problem can be seen as a wiretap channel in which user 1 is the legitimate user and user 2 is an eavesdropper. Thus, we can apply any wiretap channel solutions to solve it. Alternating optimization and water filling (AOWF) algorithm [14] and rotation-based [15] method are two of them. Once \mathbf{Q}_1^* is obtained, we plug it in (2b) and manipulate it to get

$$\mathbf{Q}_2^* = \arg \max_{\mathbf{Q}_2} \frac{1}{2} \log \frac{|\mathbf{I}_2 + \mathbf{H}_2 \mathbf{Q}_2 \mathbf{H}_2^T|}{|\mathbf{I}_1 + \mathbf{H}_1 \mathbf{Q}_2 \mathbf{H}_1^T|}, \quad (4a)$$

$$\text{s.t.} \quad \mathbf{Q}_2 \succeq \mathbf{0}, \quad \text{tr}(\mathbf{Q}_2) \leq P_2 = (1 - \alpha)P, \quad (4b)$$

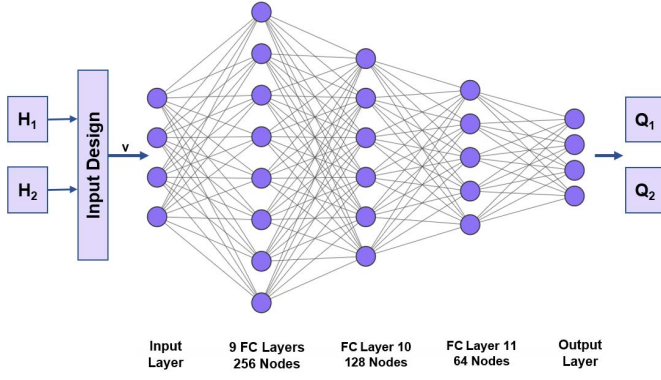


Fig. 2. The structure of the multi-layer perceptrons network used.

TABLE I
HYPER-PARAMETERS

Hyper-parameter	Value	Hyper-parameter	Value
Initial learning rate	0.001	Mini batch size	256
Learn rate drop factor	0.5	Learn rate drop period	5
Training set size	5×10^5	Validation set size	10^5
Validation frequency	10^3 iters	Validation patience	5

in which $\mathbf{H}'_i \triangleq \mathbf{\Lambda}_i^{-\frac{1}{2}} \mathbf{V}_i^T \mathbf{H}_i$, $i \in \{1, 2\}$, where $\mathbf{\Lambda}_i$ and \mathbf{V}_i are obtained from eigenvalue decomposition of $\mathbf{I} + \mathbf{H}_i \mathbf{Q}_1^* \mathbf{H}_i^T$, i.e., $\mathbf{I} + \mathbf{H}_i \mathbf{Q}_1^* \mathbf{H}_i^T = \mathbf{V}_i \mathbf{\Lambda}_i \mathbf{V}_i^T$. Then again, (4) is the rate for a MIMO wiretap channel where \mathbf{H}'_2 and \mathbf{H}'_1 are the channels corresponding to the legitimate user and eavesdropper, respectively. Thus, we solve it using a wiretap solution. Although this approach is suboptimal, the resulting rate region is close to the optimal solution—obtained by a brute-force search.

We next describe the structure of the DNN used for finding suitable covariance matrices for the MIMO-NOMA networks.

B. Network Structure

We use a *multi-layer perceptrons* (MLP) DNN in this letter. As feed-forward neural nets, MLPs are less complex, easy to design, and have quick run time. The structure of the network is shown in Fig. 2. As we will see in Section III-C, the input is a feature mapping of \mathbf{H}_1 and \mathbf{H}_2 —the channel matrices of user 1 and user 2. *Rectified linear unit* (ReLU) [16] serves as the activation function. ReLUs are sparse and have a reduced likelihood of vanishing gradient which reduce training and inference time for neural networks. The network has nine fully connected (FC) layers, each with a width of 256 nodes. The network then funnels through a 128 node layer and a 64 node layer before reaching the output layer. The output layer is the upper triangular elements of the covariance matrices (\mathbf{Q}_1 and \mathbf{Q}_2) that the network is trying to learn how to predict. We note that since \mathbf{Q}_1 and \mathbf{Q}_2 are symmetric, once we get the upper triangular elements, we know all elements. The size of the output layer depends on n_t and is equal to $n_t(n_t + 1)$.

We have investigated the effect of various hyperparameters including learning rate, drop factor, Adam optimizer, validation frequency, mini-batch size, and validation patience in order to most effectively train the network. The final, tuned hyperparameters are shown in Table I.

C. Pre-Processing

1) *Input Design*: A big advantage of DL algorithms is that they reduce/eliminate the need for feature engineering as they try to learn high-level features from data. Hence, this problem the input could simply be the channel matrices \mathbf{H}_1 and \mathbf{H}_2 . However, we have observed that some nonlinear combinations of the channels improve the network performance. Specifically, observing that $|\mathbf{I}_i + \mathbf{H}_i \mathbf{Q}_i \mathbf{H}_i^T| = |\mathbf{I}_i + \mathbf{H}_i^T \mathbf{H}_i \mathbf{Q}_i|$, we can rewrite (3) and (4) as functions of $\mathbf{H}_i^T \mathbf{H}_i$. Then, the input is designed based on $\mathbf{H}_i^T \mathbf{H}_i$, not \mathbf{H}_i [10]. This makes our design independent of the number of antennas at the users since the size of $\mathbf{H}_i^T \mathbf{H}_i$ is $n_t \times n_t$ which does not depend on n_1 and n_2 , unlike the size of \mathbf{H}_i . This also simplifies the inputs.

2) *Scaling*: Before feeding the data to the network, we scale it to avoid over-fitting and improve the performance [17]. It makes back-propagation more efficient [18], and allows the network to more quickly learn the optimal parameters for each input node. Normalizing or standardizing the inputs are the two common ways of scaling. We normalize the input variables. To summarize, the input vector \mathbf{v} is designed as

$$\mathbf{v} = [0.05\mathbf{v}_1, 0.002\mathbf{v}_2]^T, \quad (5)$$

in which \mathbf{v}_1 and \mathbf{v}_2 are given by

$$\mathbf{v}_1 = \text{vec}([\mathbf{H}_1^T \mathbf{H}_1 \quad \mathbf{H}_2^T \mathbf{H}_2]), \quad (6a)$$

$$\mathbf{v}_2 = \text{vec}([\mathbf{H}_1^T \mathbf{H}_1 \quad \mathbf{H}_2^T \mathbf{H}_2]^T [\mathbf{H}_1^T \mathbf{H}_1 \quad \mathbf{H}_2^T \mathbf{H}_2]), \quad (6b)$$

where $\text{vec}(\mathbf{A})$ converts matrix \mathbf{A} to a vector. The coefficients of \mathbf{v}_1 and \mathbf{v}_2 in (5) are chosen based on the histogram of all elements of \mathbf{v}_1 and \mathbf{v}_2 for 10^3 input channels. After this normalization, with a high probability, the elements of \mathbf{v} will be in the range of $[-1 \ 1]$.

IV. NUMERICAL RESULTS

We next evaluate the performance of the proposed DNN-based covariance matrix design (precoding and power allocation) in different antenna settings. A large dataset was used to have the network generalize for many different channels, and cross-validation data was used to prevent over-fitting. The test dataset was based on 1,000 channel matrices \mathbf{H}_1 and \mathbf{H}_2 whose elements were generated randomly based on $\mathcal{N}(0, 1)$.

The performance of the proposed solution can be evaluated in different ways. We may find the mean square error (MSE) of elements \mathbf{Q}_1 and \mathbf{Q}_2 provided by the DNN and those obtained from a traditional capacity-approaching. The lower the MSEs, the better the regression. Alternatively, we can substitute \mathbf{Q}_1 and \mathbf{Q}_2 to (2) and compare the secure rate region achieved by the DNN with analytical methods like GSVD and maximum achievable rates (capacity) for this channel. We also compare the performance of the DL and traditional iterative methods in terms of computation time.

Figures 3 and 4 illustrate the accuracy of the proposed DNN model compared to GSVD and capacity region. The rate region plots were made for $n_t = 2$ and $n_t = 3$. The users have a single antenna, and the BS power is 10 Watts. Our DNN is able to find covariance matrices for any given $\alpha \in [0, 1]$. To demonstrate this, rate region plots were made. These plots show the different rates simultaneously achieved by user 1 and user 2 as the power splitting factor α is changed. In order to be

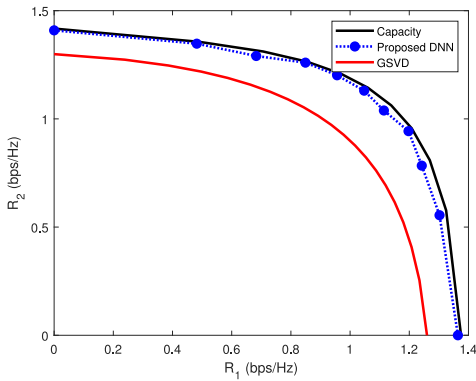


Fig. 3. Secrecy rate regions for $n_t = 2$, $n_1 = n_2 = 1$.

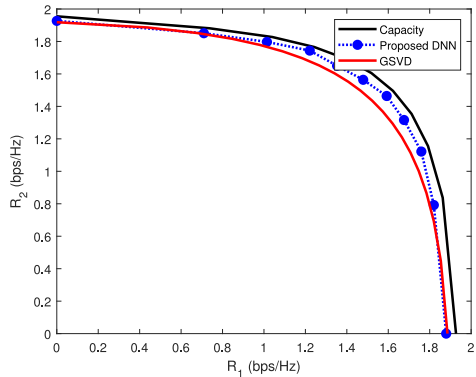


Fig. 4. Secrecy rate regions for $n_t = 3$, $n_1 = n_2 = 1$.

accurate at different power splitting factors, eleven networks were trained, each at a different α starting from $\alpha = 0$ with a step size of 0.1. This produces a smooth, piecewise linear secure rate region curve and proves that the neural network can be generalized for any α . It is important to note that different values of α correspond to different services and result in very different covariance matrices. For example, $\alpha = 0$ implies $\mathbf{Q}_1 = \mathbf{0}$ which gives $R_1 = 0$, i.e., security is important only for user 2, whereas $\alpha = 1$ has the reverse implication. That being said, we choose the value of α and the associated DNN based on the users' quality of service. Averaging over all α s, the DNN achieves %98.9 and %97.7 of the capacity rates for $n_t = 2$ and $n_t = 3$, respectively. We have used the same regularization parameters on all networks. A finer tune would increase this accuracy.

As can be seen in Fig. 3 and 4, the DNN highly outperforms analytical methods like GSVD in terms of rate region. Further, the proposed method largely outperforms existing iterative solutions [5], [6] in terms of computation time. Table II shows the input signaling design time gain that we obtain with the DNN. It is several times faster than the rotation algorithm and AOWF. All algorithms are tested on the same machine in MATLAB. The time difference becomes much larger as n_t increases. In short, the DNN is able to get very close to the capacity region of the MIMO-NOMA and achieves this much faster, and thus, it is a viable solution.

V. CONCLUSION

A novel deep learning assisted covariance matrix design for the two-user MIMO-NOMA with confidential messages

TABLE II
SOLUTION TIME IN MILLISECONDS

n_t	Rotation	AOWF	DNN
2	6.6	22.7	3.7
3	26.1	31.4	6.3

has been developed, trained, and tested in this letter. The proposed DNN, which is used to approximate the capacity region of this channel, is able to achieve nearly perfect accuracy for maximum secure rates for this channel. Remarkably, using the DNN for signaling design significantly reduces the solution time versus existing iteratively solutions and brings this time low enough that it can be used in practice. It also significantly outperforms GSVD precoding in achievable secure rates.

REFERENCES

- [1] M. Vaezi, Z. Ding, and H. V. Poor, *Multiple Access Techniques for 5G Wireless Networks and Beyond*. Cham, Switzerland: Springer, 2019.
- [2] A. Mukherjee, S. A. A. Fakoorian, J. Huang, and A. L. Swindlehurst, "Principles of physical layer security in multiuser wireless networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 3, pp. 1550–1573, 3rd Quart., 2014.
- [3] E. Ekrem and S. Ulukus, "Capacity region of Gaussian MIMO broadcast channels with common and confidential messages," *IEEE Trans. Inf. Theory*, vol. 58, no. 9, pp. 5669–5680, Sep. 2012.
- [4] S. A. A. Fakoorian and A. L. Swindlehurst, "On the optimality of linear precoding for secrecy in the MIMO broadcast channel," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 9, pp. 1701–1713, Sep. 2013.
- [5] D. Park, "Weighted sum rate maximization of MIMO broadcast and interference channels with confidential messages," *IEEE Trans. Wireless Commun.*, vol. 15, no. 3, pp. 1742–1753, Mar. 2016.
- [6] Y. Qi and M. Vaezi, "Secure transmission in MIMO-NOMA networks," *IEEE Commun. Lett.*, vol. 24, no. 12, pp. 2696–2700, Dec. 2020.
- [7] M. F. Hanif and Z. Ding, "Robust power allocation in MIMO-NOMA systems," *IEEE Wireless Commun. Lett.*, vol. 8, no. 6, pp. 1541–1545, Dec. 2019.
- [8] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge, U.K.: Cambridge Univ. Press, 2005.
- [9] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2224–2287, 3rd Quart., 2019.
- [10] X. Zhang and M. Vaezi, "Deep learning based precoding for the MIMO Gaussian wiretap channel," in *Proc. IEEE Global Commun. Conf. Workshops (GC Workshops)*, Waikoloa, HI, USA, 2019, pp. 1–6.
- [11] M. Gümüř and T. M. Duman, "Deep neural network based precoding for wiretap channels with finite alphabet inputs," *IEEE Wireless Commun. Lett.*, vol. 10, no. 8, pp. 1652–1656, Aug. 2021.
- [12] R. Liu, T. Liu, H. V. Poor, and S. Shamai, "Multiple-input multiple-output Gaussian broadcast channels with confidential messages," *IEEE Trans. Inf. Theory*, vol. 56, no. 9, pp. 4215–4227, Sep. 2010.
- [13] E. Ekrem and S. Ulukus, "The secrecy capacity region of the Gaussian MIMO multi-receiver wiretap channel," *IEEE Trans. Inf. Theory*, vol. 57, no. 4, pp. 2083–2114, Apr. 2011.
- [14] Q. Li, M. Hong, H.-T. Wai, Y.-F. Liu, W. Ma, and Z.-Q. Luo, "Transmit solutions for MIMO wiretap channels using alternating optimization," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 9, pp. 1714–1727, Sep. 2013.
- [15] X. Zhang, Y. Qi, and M. Vaezi, "A rotation-based method for precoding in Gaussian MIMOME channels," *IEEE Commun. Lett.*, vol. 69, no. 2, pp. 1189–1200, Feb. 2021.
- [16] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical evaluation of rectified activations in convolutional network," 2015. [Online]. Available: arXiv:1505.00853.
- [17] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford, U.K.: Oxford Univ. Press, 1995.
- [18] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, "Efficient BackProp," in *Neural Networks: Tricks of the Trade*. Heidelberg, Germany: Springer, 2012, pp. 9–48.