DIMENSIONALITY REDUCTION FOR BEAMFORMING BY CHANGE OF VARIABLE

Yannan Chen†, Wenyu Wang†, Licheng Zhao‡, Lei Huang*, and Kaiming Shen†*

*State Key Laboratory of Radio Frequency Heterogeneous Integration, Shenzhen University, China
†School of Science and Engineering, The Chinese University of Hong Kong (Shenzhen), China
‡Shenzhen Research Institute of Big Data, China
e-mail: `shenkaiming@cuhk.edu.cn`

ABSTRACT

This work aims to reduce the complexity of wireless beamforming when massive antennas are deployed. The main idea is to design a linear transform that maps the beamforming matrix to a low-dimension new space. While some special cases of this change-of-variable approach have been considered in the literature, this paper investigates its full generality. Two basic questions are: (i) What is the smallest dimension of the new space with the problem equivalence preserved? (ii) How is the new space constructed given the target dimension? The paper partially answers the first question by bounding the dimension limit from above. The second question is numerically difficult because it involves the high-dimension nonconvex optimization. We propose using the learning technique to extract features of the channel tensor and thereby construct the new space. Simulations show that the proposed method runs faster and achieves higher utility.

1. INTRODUCTION

A research hotspot in wireless communications is to redesign the beamforming algorithm to handle the soaring number of antennas at base station (BS). This work proposes a change-of-variable approach to the dimensionality reduction of the beamforming problem via optimization and learning jointly.

We focus on the weighted-sum-rate maximization problem for beamforming, which has been extensively studied in the existing literature over the past two decades. The mainstream approaches nowadays include the weighted minimum mean square error (WMMSE) algorithm [1, 2] and the fractional programming (FP) method [3, 4]. The two methods both work in an iterative fashion and require inverting an $M \times M$ matrix per iterate, where M is the number of antennas at the BS, so the computational complexity poses a formidable challenge when massive antennas are deployed.

A recent progress achieved in [5] is to alleviate the matrix inverse operation in the WMMSE algorithm by the following change of variable: $\boldsymbol{V} = \boldsymbol{G}\boldsymbol{X}$, where $\boldsymbol{V}$ is the beamforming variable, $\boldsymbol{G}$ is a prescribed matrix, and $\boldsymbol{X}$ is the new variable. As a consequence, the primal beamforming problem $\max_{\boldsymbol{V}} f(\boldsymbol{V})$ is converted to $\max_{\boldsymbol{X}} f(\boldsymbol{G}\boldsymbol{X})$. It turns out that the new problem can still be addressed by WMMSE and FP. Most importantly, when $\boldsymbol{G} \in \mathbb{C}^{M\times L}$, the dimension of the matrix inverse is changed from $M \times M$ to $L \times L$, so one wishes to make L small. Nevertheless, L also needs to be sufficiently large so that the new problem is still equivalent to (or at least close to) the original problem. As a theoretical contribution of this work, we provide an upper bound on the limit of L that guarantees the problem equivalence.

Only knowing the range of L is not enough. We further need to find a way to construct $\boldsymbol{G}$ efficiently given L. The previous work [5] has provided an elegant way, but it requires L to be fairly large, sometimes even larger than M so that the change of variable becomes meaningless. It is numerically difficult to find the optimal $\boldsymbol{G}$ for a small L, so we resort to machine learning. Specifically, a deep unfolding paradigm is adopted in our case to capture the iteration structure of WMMSE and FP. Because the $\boldsymbol{G}$ construction aims at the optimal column space of $\boldsymbol{G}$ where the beamforming variable resides in, we refer to this method as *basis learning*.

Aside from change of variable, other approaches to the high-dimensional beamforming problems can be found in the literature to date. The authors of [6] propose an eigenmode-based improvement of the classical low-complexity zero forcing algorithm, with a provable asymptotic gain for the sum rate as the number of user terminals tends to infinity. A more recent work [7] aims to completely eliminate the matrix inverse operation from WMMSE and FP by altering their iteration structures, albeit at the risk of slowing down convergence. Moreover, a line of works [8–10] suggest using neural networks to mimic the behavior of the WMMSE algorithm.

2. WIRELESS BEAMFORMING PROBLEM

Following [5], we focus on a single-cell downlink network where one BS sends independent data streams to $K > 1$ user terminals simultaneously via spatial multiplexing. The BS

This work was supported by NSFC No. 12426306, Guangdong Basic and Applied Basic Research Foundation No. 2024A1515010154, and the Project of Department of Education of Guangdong Province No.2021KCXTD008. *(Corresponding author: Kaiming Shen.)*

has M transmit antennas while each user terminal has N receive antennas. Note that M is typically much larger than N. Thus, the number of data streams of each user terminal, written d, is bounded by N from above. Assign a beamforming matrix $\boldsymbol{V}_k \in \mathbb{C}^{M\times d}$ to each user terminal k. Moreover, let $\boldsymbol{H}_k \in \mathbb{C}^{N\times M}$ be the MIMO channel matrix from the BS to user terminal k. The resulting achievable data rate of user terminal k is given by

$$R_k = \log\left|\boldsymbol{I} + \boldsymbol{V}_k^*\boldsymbol{H}_k^*\left(\sum_{j\neq k}\boldsymbol{H}_k\boldsymbol{V}_j\boldsymbol{V}_j^*\boldsymbol{H}_k^* + \eta\boldsymbol{I}\right)^{-1}\boldsymbol{H}_k\boldsymbol{V}_k\right|$$

with $\boldsymbol{I}$ denoting an identity matrix, η denoting the thermal noise power, and $(\cdot)^*$ denoting the conjugate transpose of a matrix. Furthermore, assign a positive rate weight $\omega_k > 0$ to each user terminal k in accordance with its priority. Assuming that the channel state information (CSI) is available, we optimize $\{\boldsymbol{V}_k\}$ to maximize the weighted sum rates:

$$\underset{\{\boldsymbol{V}_k\}}{\text{maximize}} \quad f_o(\{\boldsymbol{V}_k\}) := \sum_{k=1}^{K}\omega_k R_k \tag{1a}$$

$$\text{subject to} \quad \sum_{k=1}^{K}\|\boldsymbol{V}_k\|^2 = P, \tag{1b}$$

where $\|\cdot\|$ is the Frobenius norm, and P is the power constraint. It is easy to see that replacing "$=$" with "$\leq$" in (1b) does not impact the solution.

3. LIMITATIONS OF EXISTING METHODS

WMMSE [1, 2] and FP [3] are two common approaches[1] to problem (1), both recasting the primal objective $f_o(\{\boldsymbol{V}_k\})$ to

$$f_Q(\{\boldsymbol{V}_k\}) = \sum_{k=1}^{K}\mathrm{tr}\big(\boldsymbol{B}_k^*\boldsymbol{V}_k + \boldsymbol{V}_k^*\boldsymbol{B}_k - \boldsymbol{V}_k^*\boldsymbol{A}\boldsymbol{V}_k\big), \tag{2}$$

where the matrix $\boldsymbol{B}_k \in \mathbb{C}^{M\times d}$ and the positive semidefinite (PSD) matrix $\boldsymbol{A} \in \mathbb{H}_+^{M\times M}$ are iteratively updated as in [3]. For fixed $\boldsymbol{A}$ and $\boldsymbol{B}_k$, each $\boldsymbol{V}_k$ can be optimally obtained as

$$\boldsymbol{V}_k = (\boldsymbol{A} + \lambda\boldsymbol{I})^{-1}\boldsymbol{B}_k, \tag{3}$$

where the Lagrange multiplier $\lambda \geq 0$ is chosen to enforce $\sum_{k=1}^{K}\|\boldsymbol{V}_k\|^2 = P$. However, the solution in (3) becomes computationally daunting when M is large, since it requires inverting an $M\times M$ matrix $\boldsymbol{A}+\lambda\boldsymbol{I}$. To make matters worse, we must invert a sequence of such $M\times M$ matrices in order to decide λ (e.g., by the bisection search).

To get rid of the Lagrange multiplier tuning, the authors of [5] suggest another new objective function

$$f_S(\{\boldsymbol{V}_k\}) = \sum_{k=1}^{K}\mathrm{tr}\big(\boldsymbol{F}_k^*\boldsymbol{V}_k + \boldsymbol{V}_k^*\boldsymbol{F}_k - \boldsymbol{V}_k^*\boldsymbol{E}\boldsymbol{V}_k\big), \tag{4}$$

[1] Actually, FP amounts to a generalization of WMMSE, and can strictly outperform WMMSE when it involves discrete constraints as shown in [11].

where $\boldsymbol{F}_k \in \mathbb{C}^{M\times d}$ and $\boldsymbol{E} \in \mathbb{H}_+^{M\times M}$ are iteratively updated as specified later in Algorithm 1.

Lemma 1 *$\{\boldsymbol{V}_k'\}$ is a stationary point of problem (1) if and only if it is a stationary point of $\max f_S(\{\boldsymbol{V}_k\})$ after the iterative updates of $\boldsymbol{E}$ and $\boldsymbol{F}_k$ converge.*

Regarding the dimensionality issue, the authors of [5] further propose the following substitution:

$$\boldsymbol{V}_k = \boldsymbol{G}\boldsymbol{X}_k, \tag{5}$$

where $\boldsymbol{G} \in \mathbb{C}^{M\times NK}$ stacks the channel matrices as

$$\boldsymbol{G} = [\boldsymbol{H}_1^*, \boldsymbol{H}_2^*, \ldots, \boldsymbol{H}_K^*]. \tag{6}$$

Thus, with each $\boldsymbol{V}_k$ replaced by a new variable $\boldsymbol{X}_k \in \mathbb{C}^{NK\times d}$, the new objective $f_S(\{\boldsymbol{V}_k\})$ is rewritten as

$$\begin{aligned} f_Z(\{\boldsymbol{X}_k\}) &= f_S(\{\boldsymbol{G}\boldsymbol{X}_k\}) \\ &= \sum_{k=1}^{K}\mathrm{tr}(\boldsymbol{F}_k^*\boldsymbol{G}\boldsymbol{X}_k + \boldsymbol{X}_k^*\boldsymbol{G}^*\boldsymbol{F}_k - \boldsymbol{X}_k^*\boldsymbol{G}^*\boldsymbol{E}\boldsymbol{G}\boldsymbol{X}_k). \end{aligned} \tag{7}$$

The main result of [5] is to convert the constrained problem of $f_o(\{\boldsymbol{V}_k\})$ to an unconstrained problem of $f_S(\{\boldsymbol{X}_k\})$, as stated below.

Proposition 1 (Theorem 1 in [5]) *For $\boldsymbol{G}$ as defined in (6), $\{\boldsymbol{X}_k'\}$ is a stationary point of $\max f_Z(\{\boldsymbol{X}_k\})$ if and only if $\{\boldsymbol{V}_k' = \sqrt{\alpha}\boldsymbol{G}\boldsymbol{X}_k' : \alpha = P/\sum_{k=1}^{K}\|\boldsymbol{G}\boldsymbol{X}_k'\|^2\}$ is a stationary point of problem (1).*

Thus, we solve $\max f_S(\{\boldsymbol{X}_k\})$ as $\boldsymbol{X}_k = (\boldsymbol{G}^*\boldsymbol{E}\boldsymbol{G})^{-1}\boldsymbol{G}^*\boldsymbol{F}_k$ with $\boldsymbol{E}$ and $\boldsymbol{F}_k$ iteratively updated till convergence, and then scale all the $\boldsymbol{G}\boldsymbol{X}_k$'s simultaneously to enforce the power constraint P, namely the RWMMSE algorithm [5]. Importantly, the $M\times M$ matrix inverse in (3) is now turned to the $NK\times NK$ matrix inverse, so the computational complexity becomes much lower when $NK \ll M$.

However, $NK \ll M$ need not hold in practice, since K is also a large number. Actually, it can happen that $NK > M$, in which case RWMMSE is even less efficient than WMMSE. As such, this paper aims to reduce the problem dimensionality in general without any special assumptions from [5].

4. PROPOSED CHANGE OF VARIABLE

4.1. Fundamental Limit Analysis

We return to the variable substitution $\boldsymbol{V}_k = \boldsymbol{G}\boldsymbol{X}_k$ in (5) but no longer require $\boldsymbol{G}$ to be constructed as in (5). Rather, $\boldsymbol{G}$ can be an arbitrary $M\times L$ matrix, where L is a design parameter, i.e., $\boldsymbol{G} = [\boldsymbol{g}_1, \boldsymbol{g}_2, \ldots, \boldsymbol{g}_L]$. As a result, each column of $\boldsymbol{V}_k$ falls in the column space $\mathrm{Col}(\boldsymbol{G}) = \mathrm{span}\{\boldsymbol{g}_1, \boldsymbol{g}_2, \ldots, \boldsymbol{g}_L\}$. Thus, if $\boldsymbol{G}$ is fixed and we are only

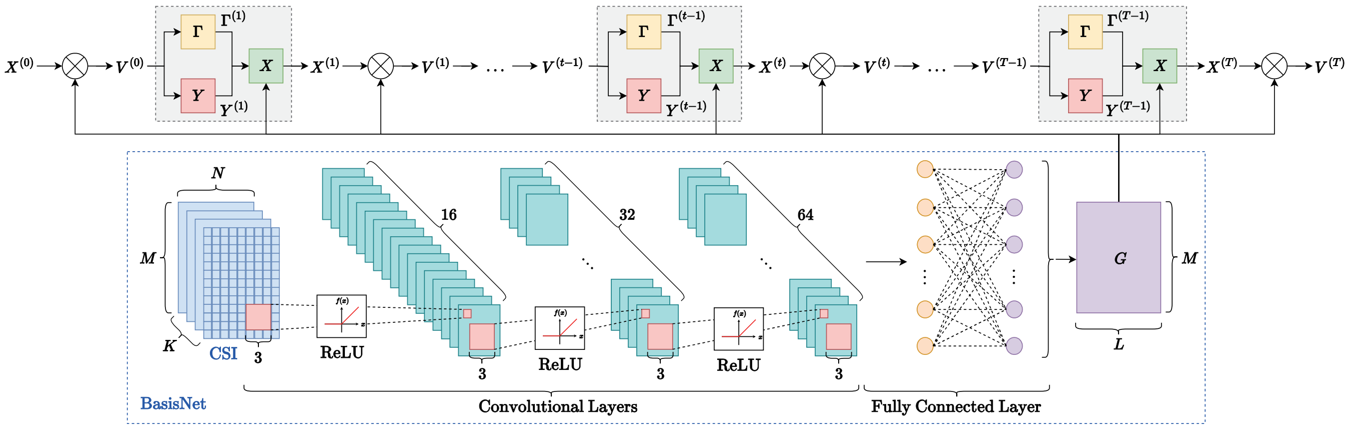

Fig. 1. The architecture of BasisNet. The features of channels are extracted by the convolutional layers, and then are fed to the fully connected layer to yield $\boldsymbol{G}$. The generated $\boldsymbol{G}$ is provided to each block for the iterative updates as shown in Algorithm 1.

allowed to change $\boldsymbol{X}_k$, then the possible values of $\boldsymbol{V}_k$ are completely determined by $\mathrm{Col}(\boldsymbol{G})$.

Note that the choice of L plays a key role. For the dimensionality reduction purpose, we wish to make L as small as possible. However, when L is too small, the new problem of $\boldsymbol{X}_k$ is no longer equivalent to the original problem of $\boldsymbol{V}_k$. A trivial choice is to let $L = M$ and $\boldsymbol{G} = \boldsymbol{I}$, so there is an identity mapping between $\boldsymbol{V}_k$ and $\boldsymbol{X}_k$. But we are most interested in this question: is it possible to let $L < M$ while guaranteeing that the primal optimal solution will not be missed out?

Thus, the main result from [5] can then be thought of as a partial answer to the above question. The authors of [5] show that L can be reduced to NK whenever $NK < M$. We now give a strengthened result in the following theorem.

Theorem 1 *When $L \geq Kd$, there exists some $\boldsymbol{G} \in \mathbb{C}^{M\times L}$ such that problem* (1) *is equivalent to* $\max f_Z(\{\boldsymbol{X}_k\})$ *in the sense that they have the same global optimum after the iterative updates of $\boldsymbol{E}$ and $\boldsymbol{F}_k$ converge.*

Proof 1 (Sketched) *Clearly, if the theorem holds for a particular L, then it also holds when L becomes larger since we can augment $\boldsymbol{G}$ by zero padding. Thus, it suffices to consider the case of $L = Kd$. Assume that $\{\boldsymbol{V}'_k\}$ is a global solution to problem* (1)*. According to the FP theory [3], we must have $\boldsymbol{V}'_k = (\boldsymbol{A} + \lambda\boldsymbol{I})^{-1}\boldsymbol{B}_k$, where $\boldsymbol{A}$, $\boldsymbol{B}_k$, and λ are described at the beginning of Section 3. Now we construct $\boldsymbol{G}$ as*

$$\boldsymbol{G} = [\boldsymbol{B}_1, \boldsymbol{B}_2, \ldots, \boldsymbol{B}_K]. \tag{8}$$

For $\boldsymbol{V}' = [\boldsymbol{V}'_1, \boldsymbol{V}'_2, \ldots, \boldsymbol{V}'_K]$, we have $\boldsymbol{V}' = (\boldsymbol{A} + \lambda\boldsymbol{I})^{-1}\boldsymbol{G}$. A key observation is that $\boldsymbol{A}$ can be factorized into $\boldsymbol{A} = \boldsymbol{G}\boldsymbol{U}$ with some $\boldsymbol{U} \in \mathbb{C}^{Kd\times M}$. As a result, we can rewrite $\boldsymbol{V}'$ as

$$\boldsymbol{V}' = (\boldsymbol{G}\boldsymbol{U} + \lambda\boldsymbol{I})^{-1}\boldsymbol{G} = \boldsymbol{G}(\boldsymbol{U}\boldsymbol{G} + \lambda\boldsymbol{I})^{-1},$$

where the second equality follows by the Woodbury matrix identity. We then recover the solution of $\{\boldsymbol{X}_k\}$ as

$$[\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_K] = (\boldsymbol{U}\boldsymbol{G} + \lambda\boldsymbol{I})^{-1}.$$

Algorithm 1 Proposed High-Dimensional Beamforming

1: construct $\boldsymbol{G}$ by BasisNet
2: initialize $\boldsymbol{X}$ to a feasible value
3: **repeat**
4: compute $\alpha = P/\sum_k \|\boldsymbol{V}_k\|^2$ and $\boldsymbol{V}_k = \boldsymbol{G}\boldsymbol{X}_k$
5: update each $\boldsymbol{\Gamma}_k$ to $\boldsymbol{V}^*\boldsymbol{H}_k^*(\sum_{j\neq k}\boldsymbol{H}_k\boldsymbol{V}_j\boldsymbol{V}_j^*\boldsymbol{H}_k^* + \frac{\eta}{\alpha}\boldsymbol{I})^{-1}\boldsymbol{H}_k\boldsymbol{V}_k$
6: update each $\boldsymbol{Y}_k$ to $(\sum_j \boldsymbol{H}_k\boldsymbol{V}_j\boldsymbol{V}_j^*\boldsymbol{H}_k^* + \frac{\eta}{\alpha}\boldsymbol{I})^{-1}\boldsymbol{H}_k\boldsymbol{V}_k$
7: update each $\boldsymbol{X}_k$ to $(\boldsymbol{G}^*\boldsymbol{E}\boldsymbol{G})^{-1}\boldsymbol{G}^*\boldsymbol{F}_k$, where $\boldsymbol{E} = \sum_j \omega_j\boldsymbol{H}_j^*\boldsymbol{Y}_j(\boldsymbol{I} + \boldsymbol{\Gamma}_j)\boldsymbol{Y}_j^*\boldsymbol{H}_j + \nu\boldsymbol{I}, \boldsymbol{F}_k = \omega_k\boldsymbol{H}_k^*\boldsymbol{Y}_k(\boldsymbol{I} + \boldsymbol{\Gamma}_k)$, and $\nu = \frac{\eta}{P}\sum_j \omega_j \mathrm{tr}\big(\boldsymbol{Y}_j(\boldsymbol{I} + \boldsymbol{\Gamma}_j)\boldsymbol{Y}_j^*\big)$
8: **until** convergence
9: output $\{\boldsymbol{V}_k = \sqrt{\alpha}\boldsymbol{V}_k\}$

Note that the condition $L = Kd$ is used when we construct $\boldsymbol{G}$ in (8)*. Furthermore, it can be shown (e.g., by the FP theory from [3]) that $f_Z(\{\boldsymbol{X}_k\}) = f_o(\{\boldsymbol{V}'_k\})$ after the iterative updates of $\boldsymbol{A}$ and $\boldsymbol{B}_k$ converge. Thus, the global optimality of $\{\boldsymbol{X}_k\}$ can be verified immediately.*

Remark 1 *The choice of $\boldsymbol{G}$ in Theorem 1 may not be unique. For example, when $L = KN$, both* (6) *and* (8) *are feasible, and they lead to distinct choices of $\boldsymbol{G}$ in general.*

Remark 2 *Theorem 1 is mainly about the existence of $\boldsymbol{G}$. The way we construct $\boldsymbol{G}$ as in* (8) *is of limited practical interest because it entails iterative update of $\boldsymbol{B}_k$, which leads us back to the complexity of WMMSE and FP.*

4.2. Basis Learning

It remains to find a “good” $\boldsymbol{G}$ given L. As implied in Remark 2, it is difficult to decide $\boldsymbol{G}$ by the optimization tools. We propose using a neural network architecture to learn the $\boldsymbol{G}$ construction based on the current CSI. Recall that $\boldsymbol{G}$ serves to give a column space $\mathrm{Col}(\boldsymbol{G})$ that limits $\boldsymbol{V}_k$, so our neural

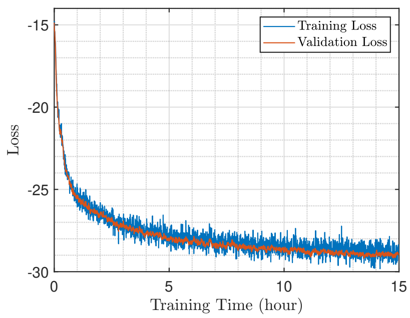

Fig. 2. Loss convergence for training.

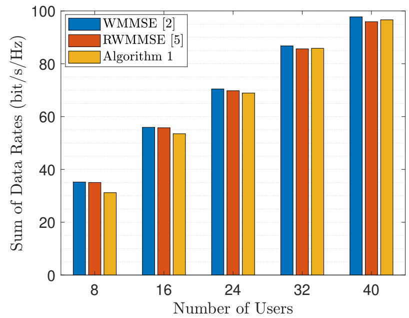

Fig. 3. Sum rate vs. user number.

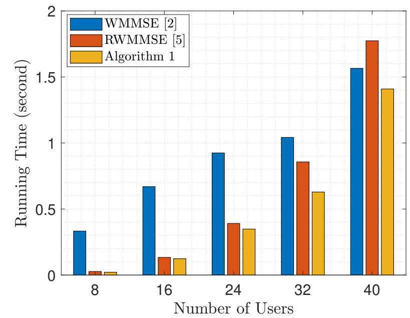

Fig. 4. Running time vs. user number.

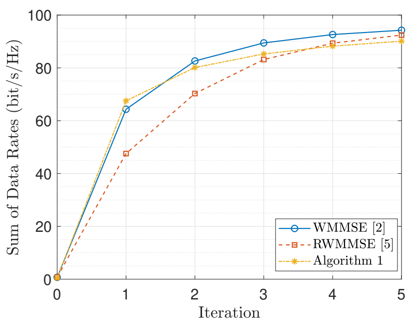

Fig. 5. Sum rate vs. iterations.

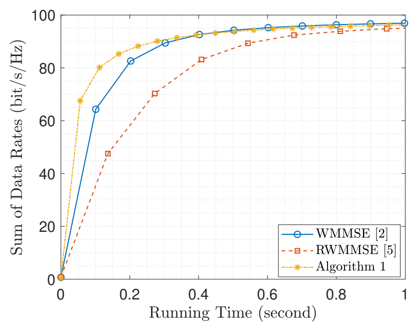

Fig. 6. Sum rate vs. running time.

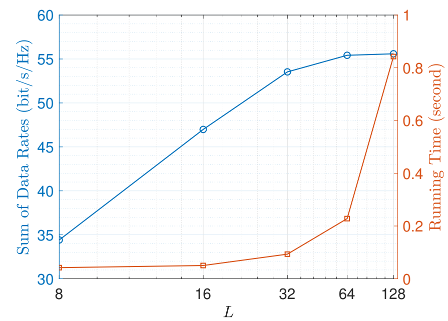

Fig. 7. Influence of L on Algorithm 1.

network in essence learns how to choose the basis of a column space. Thus, the proposed neural network architecture is termed *BasisNet*. The proposed beamforming algorithm is described in Algorithm 1, with BasisNet illustrated in Fig. 1.

As shown in Fig. 1, we adopt a deep unfolding paradigm with T blocks for BasisNet, where each block corresponds to one iterate of updates in Algorithm 1. Since it is the last iterate that determines the output result, the loss function for training is based on $f_Z(\{\boldsymbol{X}_k\})$ associated with the last block:

$$\text{Loss} = -\sum_{k=1}^{K} \mathrm{tr}\Big((\boldsymbol{F}_k^{(T)})^* \boldsymbol{G}\boldsymbol{X}_k^{(T)} + (\boldsymbol{X}_k^{(T)})^* \boldsymbol{G}^* \boldsymbol{F}_k^{(T)} - (\boldsymbol{X}_k^{(T)})^* \boldsymbol{G}^* \boldsymbol{E}^{(T)} \boldsymbol{G}\boldsymbol{X}_k^{(T)}\Big). \quad (9)$$

Moreover, the lower part of Fig. 1 shows how to generate $\boldsymbol{G}$ with the CSI as input. We extract features from the CSI tensor through the convolutional layers, and then decide $\boldsymbol{G}$ based on these features through the fully connected layer. At the training stage, we tune the kernels of the convolutional layers and the link weights of the fully connected layer.

5. SIMULATION RESULTS

We generate channels by the QuaDRiGa simulator [12] for the *3GPP TR 38.901 UMa LOS* scenario at 6.7 GHz. The receivers are randomly distributed within a $500\text{m} \times 500\text{m}$ square area, with the BS deployed at the center. Let $P = 20$ dBm and $\eta = -80$ dBm. We assume by default that $d = 2$, $M = 128$, $N = 4$, and $K = 16$. Set $L = Kd$.

As shown in Fig. 2, it takes about half day time for Algorithm 1 to finish the training session to obtain $\boldsymbol{G}$. Observe from Fig. 3 that the proposed algorithm achieves similar performance to that of WMMSE [2] and RWMMSE [5]. According to Fig. 4, we see that RWMMSE and our algorithm are much more efficient than WMMSE when K is small. When K becomes large, then this efficiency advantage shrinks, but our algorithm still runs fastest. Note that RWMMSE is even slower than WMMSE when $K = 40$; this result shows the limitation of RWMMSE as stated in Section 3. Moreover, we observe from Fig. 5 and Fig. 6 that the proposed algorithm enhances the sum-rate objective much more efficiently than WMMSE, either in terms of the number of iterations or in terms of running time. Lastly, we plot the sum rate and the running time of Algorithm 1 with respect to different values of L. When $L = 128$ without any dimensionality reduction, our algorithm and WMMSE are similar. When L is reduced to 64, the running time of our algorithm drops sharply but the sum-rate metric is barely impacted.

6. CONCLUSION

This work concerns the high-dimension wireless beamforming problem, with two main results. First, we analyze the fundamental limit of dimensionality reduction for the beamforming matrices. Second, we devise a basis learning approach to the practical implementation of this dimensionality reduction.

7. REFERENCES

[1] S. S. Christensen, R. Agarwal, E. D. Carvalho, and J. M. Cioffi, “Weighted sum-rate maximization using weighted MMSE for MIMO-BC beamforming design,” *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 4792–4799, Dec. 2008.

[2] Q. Shi, M. Razaviyayn, Z.-Q. Luo, and C. He, “An iteratively weighted MMSE approach to distributed sum-utility maximization for a MIMO interfering broadcast channel,” *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4331–4340, Sept. 2011.

[3] K. Shen and W. Yu, “Fractional programming for communication systems—Part I: Power control and beamforming,” *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2616–2630, Mar. 2018.

[4] K. Shen, Z. Zhao, Y. Chen, Z. Zhang, and H. V. Cheng, “Accelerating quadratic transform and WMMSE,” *IEEE J. Sel. Areas Commun.*, vol. 42, no. 11, pp. 3110–3124, Nov. 2024.

[5] X. Zhao, S. Lu, Q. Shi, and Z.-Q. Luo, “Rethinking WMMSE: Can its complexity scale linearly with the number of BS antennas?” *IEEE Trans. Signal Process.*, vol. 71, pp. 433–446, Feb. 2023.

[6] L. Sun and M. R. McKay, “Eigen-based transceivers for the MIMO broadcast channel with semi-orthogonal user selection,” *IEEE Trans. Signal Process.*, vol. 58, no. 10, pp. 5246–5261, Oct. 2010.

[7] J. Zhu, T.-H. Chang, L. Xiang, and K. Shen, “DeepFP: Deep-unfolded fractional programming for massive MIMO beamforming,” in *IEEE Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, July 2025.

[8] 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.

[9] M. Zhu, T.-H. Chang, and M. Hong, “Learning to beamform in heterogeneous massive MIMO networks,” *IEEE Trans. Wireless Commun.*, vol. 22, no. 7, pp. 4901–4915, July 2022.

[10] Q. Xu, J. Sun, and Z. Xu, “Efficient MU-MIMO beamforming based on majorization-minimization and deep unfolding,” *IEEE Trans. Wireless Commun.*, vol. 24, no. 5, pp. 3949–3963, May 2025.

[11] K. Shen and W. Yu, “Fractional programming for communication systems—Part II: Uplink scheduling via matching,” *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2631–2644, Mar. 2018.

[12] S. Jaeckel, L. Raschkowski, K. Börner, and L. Thiele, “Quadriga: A 3-D multi-cell channel model with time evolution for enabling virtual field trials,” *IEEE Trans. Antennas Propag.*, vol. 62, no. 6, pp. 3242–3256, June 2014.