# The WMMSE Algorithm Based on BFGS Quasi-Newton Method in Massive MIMO Systems

Yuhuan Zhao[1], Bin Yan[1], Zheng Wang[1], Yongming Huang[1] and Qingjiang Shi[2]
[1]School of Information Science and Engineering, Southeast University, Nanjing, China
[2]School of Software Engineering, Tongji University, Shanghai, China
Email: {yuhuan_zhao, bin_yan, zheng_wang, huangym}@seu.edu.cn; shiqj@tongji.edu.cn

***Abstract*—In this paper, a novel low-complexity weighted minimum mean squared error (WMMSE) algorithm for downlink massive multiple-input multiple-output (MIMO) systems is studied in detail. Specifically, we firstly adopt the generalized Broyden-Fletcher-Goldfarb-Shanno (BFGS) quasi-Newton method into the WMMSE algorithm, where the convergence of the introduced generalized BFGS method is guaranteed. Then, based on it, we achieve a significant reduction in the complexity of the proposed WMMSE-BFGS algorithm, whose computational complexity only grows linearly with the number of transmit antennas. Finally, simulation results are given to show that the proposed WMMSE-BFGS achieves lower complexity cost with competitive performance compared to the classic WMMSE.**

***Index Terms*—Massive MIMO, Precoding, WMMSE, Quasi-Newton.**

## I. Introduction

Massive MIMO enables significant gains in spectral and energy efficiency for 5G and future beyond 5G/6G systems [1]. To maximize the sum rate of the downlink massive MIMO systems, the celebrated WMMSE algorithm is proposed in [2], which tries to solve the constrained weighted sum rate (WSR) maximization problem via block coordinate descent (BCD) method. Moreover, by making use of the full power property, the work in [3] transforms the constrained WSR maximization problem into an equivalent unconstrained problem for massive MIMO downlink systems. Nevertheless, the computational complexity of WMMSE grows cubically, which becomes unaffordable with the rapid increment of the number of the transmit antennas. To this end, a number of works about the low-complexity WMMSE algorithm have been given [4]–[6].

Specifically, a variant of WMMSE algorithm is given in [4], which bypasses the matrix inversion by applying gradient descent and Schulz iterations. To accelerate the convergence of WMMSE, techniques of projection and extrapolation are employed in [5], which facilitate the next iteration by generating a predicted point in the feasible set. In [6], the quasi-Newton strategy is applied to WMMSE. More specifically, as one of the methods in quasi-Newton framework, the symmetric-rank-1 (SR1) quasi-Newton method is introduced to reduce the complexity order of WMMSE from cubic to quadratic in the number of transmit antennas. Unfortunately, since the convergence of the introduced SR1 quasi-Newton method in [6] cannot be guaranteed, the performance loss turns out to be inevitable. Meanwhile, the SR1-based WMMSE algorithm proposed in [6] only works for multiple-input single-output (MISO) systems, rendering it limited in many cases of interest.

Corresponding author: Zheng Wang (email: wznuaa@gmail.com)

In this paper, under the framework of quasi-Newton, we introduce the generalized BFGS method into WMMSE, which not only works for massive MIMO systems, but also overcomes the convergence issue. Based on it, a low-complexity WMMSE-BFGS algorithm is proposed, whose complexity scales linearly with respect to the number of transmit antennas. Finally, the simulations verify the proposed WMMSE-BFGS exhibits the lower complexity than WMMSE at the same WSR.

## II. WMMSE For Downlink Massive MIMO Systems

We consider a massive MIMO system, where a single base station (BS) with $M$ transmit antennas simultaneously serves $K$ users. Each user $k$, $k = 1, 2, ..., K$, has $N_k$ receive antennas. Let $\mathbf{H}_k \in \mathbb{C}^{N_k \times M}$ denote the channel between BS and user $k$. $\mathbf{s}_k \in \mathbb{C}^{L_k}$ represents the desired symbol vector for user $k$ and $\bar{\mathbf{V}}_k \in \mathbb{C}^{M \times L_k}$ is the linear precoder for user $k$. Then the transmitted signal of BS can be written as $\mathbf{x} = \sum_{k=1}^{K} \bar{\mathbf{V}}_k \mathbf{s}_k$, where we assume $\mathbf{s}_k \sim \mathcal{CN}(\mathbf{0}, \mathbf{I})$. Accordingly, the received signal for user $k$ can be expressed as

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{x} + \mathbf{n}_k = \mathbf{H}_k \bar{\mathbf{V}}_k \mathbf{s}_k + \sum_{i=1, i\neq k}^{K} \mathbf{H}_k \bar{\mathbf{V}}_i \mathbf{s}_i + \mathbf{n}_k, \quad (1)$$

where $\mathbf{n}_k \in \mathbb{C}^{N_k}$ is the additive white Gaussian noise (AWGN) with distribution $\mathcal{CN}(\mathbf{0}, \sigma_k^2 \mathbf{I})$ for user $k$.

Given the received signal shown in (1), the signal to interference plus noise ratio (SINR) of user $k$ is written as

$$\mathrm{SINR}_k = \mathbf{H}_k \bar{\mathbf{V}}_k \bar{\mathbf{V}}_k^H \mathbf{H}_k^H \left( \sum_{i=1, i\neq k}^{K} \mathbf{H}_k \bar{\mathbf{V}}_i \bar{\mathbf{V}}_i^H \mathbf{H}_k^H + \sigma_k^2 \mathbf{I} \right)^{-1}, \quad (2)$$

leading to the achievable rate of user $k$ given by

$$R_k = \log\det(\mathbf{I} + \mathrm{SINR}_k). \quad (3)$$

Based on $R_k$ in (3), the WSR maximization problem under a sum power constraint is formulated as [2]

$$\max_{\{\bar{\mathbf{V}}_k\}_{k=1}^K} \quad \sum_{k=1}^{K} \alpha_k R_k \quad (4)$$

**Algorithm 1:** The WMMSE Algorithm

**Input:** $\mathbf{H}_k, \alpha_k, \sigma_k^2, P_{\max}, \forall k$.
**Initialize:** $\{\mathbf{V}_k\}_{k=1}^K$ with $\sum_{k=1}^K \mathrm{Tr}(\mathbf{V}_k\mathbf{V}_k^H) \le P_{\max}$ and $\mathbf{W}_k = \mathbf{I}, \forall k$.

```
repeat
    update U_k = (N_k + H_k V_k V_k^H H_k^H)^{-1} H_k V_k, ∀k;
    update W_k = (I − U_k^H H_k V_k)^{-1}, ∀k;
    update V_k by (8), ∀k;
until convergence;
```

**Output:** $\bar{\mathbf{V}}_k = \sqrt{w}\mathbf{V}_k, \forall k$.

$$\text{s.t.} \quad \sum_{k=1}^{K} \mathrm{Tr}(\bar{\mathbf{V}}_k\bar{\mathbf{V}}_k^H) \le P_{\max}.$$

Here, the weight $\alpha_k > 0$ denotes the priority of user $k$ and $P_{\max}$ is the power budget of BS. Moreover, the problem in (4) can be equivalently reformulated as [3]

$$\min_{\{\mathbf{U}_k,\mathbf{W}_k,\mathbf{V}_k\}_{k=1}^K} \quad \sum_{k=1}^{K} \alpha_k \left( \mathrm{Tr}(\mathbf{W}_k\mathbf{E}_k) - \log\det(\mathbf{W}_k) \right) \tag{5}$$

with the variable $\mathbf{V}_k = \frac{1}{\sqrt{w}}\bar{\mathbf{V}}_k$ and $w = \frac{P_{\max}}{\sum_{k=1}^K \mathrm{Tr}\left(\mathbf{V}_k\mathbf{V}_k^H\right)}$. Here, $\mathbf{W}_k \in \mathbb{C}^{L_k\times L_k}$ and $\mathbf{U}_k \in \mathbb{C}^{N_k\times L_k}$ denote the positive definite weight matrix and the receiving matrix of user $k$ respectively. Meanwhile, the mean squared error (MSE) matrix $\mathbf{E}_k \in \mathbb{C}^{L_k\times L_k}$ of user $k$ is expressed as

$$\mathbf{E}_k = (\mathbf{I} - \mathbf{U}_k^H\mathbf{H}_k\mathbf{V}_k)(\mathbf{I} - \mathbf{U}_k^H\mathbf{H}_k\mathbf{V}_k)^H + \mathbf{U}_k^H\mathbf{N}_k\mathbf{U}_k \tag{6}$$

with $\mathbf{N}_k \triangleq \sum_{i=1,i\neq k}^K \mathbf{H}_k\mathbf{V}_i\mathbf{V}_i^H\mathbf{H}_k^H + \frac{\sigma_k^2}{P_{\max}}\sum_{i=1}^K \mathrm{Tr}(\mathbf{V}_i\mathbf{V}_i^H)\mathbf{I}$.

As the problem in (5) becomes a convex and unconstrained minimization problem with respect to each variable $\mathbf{U}_k$, $\mathbf{W}_k$ and $\mathbf{V}_k$, the classic WMMSE can be applied to address it, which employs the BCD method to sequentially update variables $\mathbf{U}_k$, $\mathbf{W}_k$ and $\mathbf{V}_k$ by fixing the other two variables [2]. For instance, in WMMSE, by fixing variables $\mathbf{U}_k$ and $\mathbf{W}_k$, the optimization problem specified in (5) turns out to be

$$\min_{\{\mathbf{V}_k\}_{k=1}^K} \quad \sum_{k=1}^{K} \alpha_k \mathrm{Tr}(\mathbf{W}_k\mathbf{E}_k), \tag{7}$$

and the optimal $\mathbf{V}_k$ can be obtained by

$$\begin{aligned}\mathbf{V}_k = \Bigg( &\sum_{i=1}^{K} \alpha_i \mathbf{H}_i^H\mathbf{U}_i\mathbf{W}_i\mathbf{U}_i^H\mathbf{H}_i \\ &+ \sum_{i=1}^{K}\alpha_i \mathrm{Tr}\left(\frac{\sigma_i^2}{P_{\max}}\mathbf{U}_i\mathbf{W}_i\mathbf{U}_i^H\right)\mathbf{I}\Bigg)^{-1} \alpha_k\mathbf{H}_k^H\mathbf{U}_k\mathbf{W}_k, \forall k.\end{aligned} \tag{8}$$

As for complexity, compared to the updates of $\mathbf{W}_k$ and $\mathbf{U}_k$, the complexity cost of WMMSE is dominated by the update of $\mathbf{V}_k$. To be more specific, due to the matrix inversion in (8), updating $\mathbf{V}_k$ reaches a complexity order $\mathcal{O}(M^3)$, and becomes prohibitive in practice when $M$ is sufficiently large. Consequently, the quasi-Newton strategy is introduced in [6] to exclusively solve the problem in (7) with reduced complexity cost. However, the introduced SR1 method in [6] is designed only for MISO systems and not guaranteed to converge.

## III. The WMMSE-BFGS Algorithm

In this section, the generalized BFGS method is introduced into WMMSE to solve the problem in (7), which works for massive MIMO systems with guaranteed convergence.

First of all, for notational simplicity, we define the objective function of (7) as follows

$$f(\mathbf{V}_k) \triangleq \sum_{k=1}^{K} \alpha_k \mathrm{Tr}(\mathbf{W}_k\mathbf{E}_k), \tag{9}$$

and its gradient with respect to $\mathbf{V}_k$ is given by

$$\begin{aligned}\nabla_{\mathbf{V}_k} f(\mathbf{V}_k) = &-2\alpha_k\mathbf{H}_k^H\mathbf{U}_k\mathbf{W}_k + 2\sum_{i=1}^{K}\alpha_i\mathbf{H}_i^H\mathbf{U}_i\mathbf{W}_i(\mathbf{U}_i^H\mathbf{H}_i\mathbf{V}_k) \\ &+ 2\sum_{i=1}^{K}\alpha_i\mathrm{Tr}\left(\frac{\sigma_i^2}{P_{\max}}\mathbf{U}_i\mathbf{W}_i\mathbf{U}_i^H\right)\mathbf{V}_k \in \mathbb{C}^{M\times L_k}.\end{aligned} \tag{10}$$

Subsequently, let $\mathbf{V}_k \triangleq [\mathbf{v}_{k1}, \ldots, \mathbf{v}_{kl}, \ldots, \mathbf{v}_{kL_k}]$ with $\mathbf{v}_{kl} \in \mathbb{C}^M$ for $l = 1, \ldots, L_k$. Given $\mathbf{U}_k \neq \mathbf{0}$ for some $k$, the positive definite Hessian matrix $\nabla^2_{\mathbf{v}_{kl}} f \in \mathbb{C}^{M\times M}$ with respect to $\mathbf{v}_{kl}$ is written as

$$\begin{aligned}&\nabla^2_{\mathbf{v}_{k1}} f = \cdots = \nabla^2_{\mathbf{v}_{kl}} f = \cdots = \nabla^2_{\mathbf{v}_{kL_k}} f \\ &= 2\sum_{i=1}^{K}\alpha_i\mathbf{H}_i^H\mathbf{U}_i\mathbf{W}_i\mathbf{U}_i^H\mathbf{H}_i + 2\sum_{i=1}^{K}\alpha_i\mathrm{Tr}\left(\frac{\sigma_i^2}{P_{\max}}\mathbf{U}_i\mathbf{W}_i\mathbf{U}_i^H\right)\mathbf{I}.\end{aligned} \tag{11}$$

Then, given (10) and (11), to solve the problem in (7), the traditional Newton method for seeking $\mathbf{V}_k$ works iteratively by

$$\mathbf{V}_k^{t+1} = \mathbf{V}_k^t - a\cdot\left(\nabla^2_{\mathbf{v}_{kl}} f\right)^{-1}\nabla_{\mathbf{V}_k} f(\mathbf{V}_k^t), \tag{12}$$

where $t$ is the iteration index and $a$ is the positive step size. To bypass the costly computation of $\nabla^2_{\mathbf{v}_{kl}} f$ and its inverse in (12), we now introduce the generalized BFGS method from the strategy of quasi-Newton to solve the problem in (7) by

$$\mathbf{V}_k^{t+1} = \mathbf{V}_k^t - a\cdot\mathbf{G}^t\nabla_{\mathbf{V}_k} f(\mathbf{V}_k^t), \tag{13}$$

where the approximation matrix $\mathbf{G}^t \in \mathbb{C}^{M\times M}$ to $\left(\nabla^2_{\mathbf{v}_{kl}} f\right)^{-1}$ is generated in an iterative way [7]

$$\begin{aligned}\mathbf{G}^t = &\left(\mathbf{I} - \mathbf{S}^t\mathbf{\Phi}^t\left(\mathbf{Y}^t\right)^H\right)\mathbf{G}^{t-1}\left(\mathbf{I} - \mathbf{Y}^t\mathbf{\Phi}^t\left(\mathbf{S}^t\right)^H\right) \\ &+ \mathbf{S}^t\mathbf{\Phi}^t\left(\mathbf{S}^t\right)^H\end{aligned} \tag{14}$$

with $\mathbf{S}^t \triangleq \mathbf{V}_k^t - \mathbf{V}_k^{t-1}$, $\mathbf{Y}^t \triangleq \nabla_{\mathbf{V}_k} f(\mathbf{V}_k^t) - \nabla_{\mathbf{V}_k} f(\mathbf{V}_k^{t-1})$ and $\mathbf{\Phi}^t \triangleq \left(\left(\mathbf{Y}^t\right)^H\mathbf{S}^t\right)^{-1}$. Here, the initial approximation $\mathbf{G}^0$ is set to the identity matrix $\mathbf{I}$, and the initial precoder $\mathbf{V}_k^0$ is the current BCD iterate $\mathbf{V}_k$.

Since it is straightforward to verify $\frac{\partial\mathrm{vec}\left(\nabla_{\mathbf{V}_k} f(\mathbf{V}_k)\right)}{\partial\mathrm{vec}^T(\mathbf{V}_k^*)} = \mathbf{0}$ with the vectorization operator $\mathrm{vec}(\cdot)$, we can introduce the Hessian matrix of $f(\mathbf{V}_k)$ in (9) with respect to $\mathbf{V}_k$, i.e., $\mathcal{H}_{\mathbf{V}_k,\mathbf{V}_k^*} f \triangleq \frac{\partial\mathrm{vec}\left(\nabla_{\mathbf{V}_k} f(\mathbf{V}_k)\right)}{\partial\mathrm{vec}^T(\mathbf{V}_k)}$, to judge its convexity [8].

**Lemma 1.** *The function $f(\mathbf{V}_k)$ in (9) is $\mu$-strongly convex, since the Hessian matrix of $f(\mathbf{V}_k)$ with respect to $\mathbf{V}_k$, i.e., $\mathcal{H}_{\mathbf{V}_k,\mathbf{V}_k^*}f$, satisfies*

$$\mathcal{H}_{\mathbf{V}_k,\mathbf{V}_k^*}f - \mu\mathbf{I}_{ML_k} \succeq \mathbf{0} \tag{15}$$

*with*

$$\mu = 2\sum_{i=1}^{K}\alpha_i \mathrm{Tr}\left(\frac{\sigma_i^2}{P_{\max}}\mathbf{U}_i\mathbf{W}_i\mathbf{U}_i^H\right) > 0. \tag{16}$$

*Proof.* From (10) and (11), we can derive

$$\nabla_{\mathbf{V}_k}f(\mathbf{V}_k) = -2\alpha_k\mathbf{H}_k^H\mathbf{U}_k\mathbf{W}_k + \nabla^2_{\mathbf{v}_{kl}}f\cdot\mathbf{V}_k. \tag{17}$$

Then, based on (17), the Hessian matrix $\mathcal{H}_{\mathbf{V}_k,\mathbf{V}_k^*}f$ follows

$$\begin{aligned}\mathcal{H}_{\mathbf{V}_k,\mathbf{V}_k^*}f &= \frac{\partial \mathrm{vec}\left(-2\alpha_k\mathbf{H}_k^H\mathbf{U}_k\mathbf{W}_k + \nabla^2_{\mathbf{v}_{kl}}f\cdot\mathbf{V}_k\right)}{\partial\mathrm{vec}^T(\mathbf{V}_k)}\\ &= \frac{\partial\left(\left(\mathbf{I}_{L_k}\otimes\nabla^2_{\mathbf{v}_{kl}}f\right)\mathrm{vec}(\mathbf{V}_k)\right)}{\partial\mathrm{vec}^T(\mathbf{V}_k)}\\ &= \mathbf{I}_{L_k}\otimes\nabla^2_{\mathbf{v}_{kl}}f\\ &\succeq \mu\mathbf{I}_{ML_k}\end{aligned} \tag{18}$$

with Kronecker product $\otimes$. Specifically, the second equality holds by the property $\mathrm{vec}(\mathbf{XYZ}) = (\mathbf{Z}^T\otimes\mathbf{X})\mathrm{vec}(\mathbf{Y})$ [8]. □

Next, to proceed the following analysis, let us invoke the Lemmas 2 and 3 from [7], [9], which confirm the convergence of generalized BFGS method.

**Lemma 2.** [7] *For a $\mu$-strongly convex problem, the generalized BFGS method generates symmetric and positive definite approximations if the initial approximation $\mathbf{G}^0 \succ \mathbf{0}$.*

**Lemma 3.** [9] *For a $L$-smooth and $\mu$-strongly convex problem, the quasi-Newton method converges if the approximations are symmetric and positive definite and the fixed step size is chosen from $a\in(0,\frac{2}{L/\lambda_{min}+\mu/\Lambda_{min}}]$, where $\lambda_{min}\triangleq\min_{t\in\mathcal{L}}\lambda^t$ and $\Lambda_{min}\triangleq\min_{t\in\mathcal{L}}\Lambda^t$ during the $\mathcal{L}$ quasi-Newton iterations. Here, $\lambda^t$ and $\Lambda^t$ are the minimum and maximum eigenvalues of $\mathbf{G}^t$ at the $t$-th quasi-Newton iteration respectively.*

Based on Lemmas 1, 2 and 3, since $f(\mathbf{V}_k)$ is $L$-smooth [4], by defining $\mathbf{V}_k^*$ as the unique minimum of the problem in (7) at the current BCD iteration, we can readily arrive at the following result.

**Theorem 1.** *For the subproblem in (7), the WMMSE-BFGS algorithm converges to the unique minimum, i.e.,*

$$\lim_{t\to\infty}\mathbf{V}_k^t = \mathbf{V}_k^* \tag{19}$$

*with the positive definite initial approximation $\mathbf{G}^0$ and step size $a\in(0,\frac{2}{L/\lambda_{min}+\mu/\Lambda_{min}}]$.*

## IV. Complexity Reduction of the WMMSE-BFGS Algorithm

With regard to the complexity of the proposed generalized BFGS for WMMSE, the complexity order of updating $\mathbf{G}^t$ in (14) is $\mathcal{O}(M^3)$, followed by the matrix multiplication between $\mathbf{G}^t$ and $\nabla_{\mathbf{V}_k}f(\mathbf{V}_k^t)$ in (13) with complexity order $\mathcal{O}(M^2L_k)$. To alleviate the complexity burden, we propose to compute and update the term $\mathbf{G}^t\nabla_{\mathbf{V}_k}f(\mathbf{V}_k^t)\in\mathbb{C}^{M\times L_k}$ as a whole, so that the iteration in (13) becomes

$$\mathbf{V}_k^{t+1} = \mathbf{V}_k^t - a\cdot\mathbf{D}^t, \tag{20}$$

where

$$\mathbf{D}^t \triangleq \mathbf{G}^t\nabla_{\mathbf{V}_k}f(\mathbf{V}_k^t) \tag{21}$$

is also known as the quasi-Newton direction. Moreover, for the sake of notational simplicity, the update of $\mathbf{G}^t$ in (14) can be recast as

$$\mathbf{G}^t = \mathbf{M}^t\mathbf{G}^{t-1}\mathbf{N}^t + \mathbf{S}^t\mathbf{\Phi}^t\left(\mathbf{S}^t\right)^H \tag{22}$$

with

$$\mathbf{M}^t \triangleq \mathbf{I} - \mathbf{S}^t\mathbf{\Phi}^t\left(\mathbf{Y}^t\right)^H, \tag{23a}$$

$$\mathbf{N}^t \triangleq \mathbf{I} - \mathbf{Y}^t\mathbf{\Phi}^t\left(\mathbf{S}^t\right)^H. \tag{23b}$$

Then, given $\mathbf{G}^0$, the computation of $\mathbf{D}^t$ in (21) can be expressed as follows

$$\begin{aligned}\mathbf{D}^t =& \left(\prod_{j=0}^{t-1}\mathbf{M}^{t-j}\right)\mathbf{G}^0\left(\prod_{j=1}^{t}\mathbf{N}^j\right)\nabla_{\mathbf{V}_k}f(\mathbf{V}_k^t)\\ &+\sum_{i=1}^{t}\left(\prod_{j=0}^{t-1-i}\mathbf{M}^{t-j}\right)\mathbf{S}^i\mathbf{\Phi}^i\left(\mathbf{S}^i\right)^H\left(\prod_{j=i+1}^{t}\mathbf{N}^j\right)\nabla_{\mathbf{V}_k}f(\mathbf{V}_k^t).\end{aligned} \tag{24}$$

Clearly, by virtue of the low-dimensional matrix $\nabla_{\mathbf{V}_k}f(\mathbf{V}_k^t)\in\mathbb{C}^{M\times L_k}$, the complexity order of computing $\mathbf{D}^t$ in (24) is reduced to $\mathcal{O}(M^2L_k)$.

Next, to further reduce the complexity cost of $\mathbf{D}^t$ in (24), motivated by L-BFGS, we define

$$\mathbf{Q}^i \triangleq \left(\prod_{j=i}^{t}\mathbf{N}^j\right)\nabla_{\mathbf{V}_k}f(\mathbf{V}_k^t) \tag{25}$$

and

$$\mathbf{A}^i \triangleq \mathbf{\Phi}^i\left(\mathbf{S}^i\right)^H\mathbf{Q}^{i+1}, \tag{26}$$

so that $\mathbf{D}^t$ in (24) can be reexpressed by

$$\mathbf{D}^t = \left(\prod_{j=0}^{t-1}\mathbf{M}^{t-j}\right)\mathbf{G}^0\mathbf{Q}^1 + \sum_{i=1}^{t}\left(\prod_{j=0}^{t-1-i}\mathbf{M}^{t-j}\right)\mathbf{S}^i\mathbf{A}^i. \tag{27}$$

Then, based on (23b) and (26), $\mathbf{Q}^i$ in (25) follows

$$\begin{aligned}\mathbf{Q}^i &= \mathbf{N}^i\left(\prod_{j=i+1}^{t}\mathbf{N}^j\right)\nabla_{\mathbf{V}_k}f(\mathbf{V}_k^t)\\ &= \mathbf{N}^i\mathbf{Q}^{i+1}\\ &= \mathbf{Q}^{i+1} - \mathbf{Y}^i\mathbf{\Phi}^i\left(\mathbf{S}^i\right)^H\mathbf{Q}^{i+1}\\ &= \mathbf{Q}^{i+1} - \mathbf{Y}^i\mathbf{A}^i.\end{aligned} \tag{28}$$

By doing this, the results of $\mathbf{Q}^i$ and $\mathbf{A}^i$ can be efficiently obtained in an iterative way, where the complexities of computing $\mathbf{Q}^i$ and $\mathbf{A}^i$ are $ML_k^2$ and $2ML_k^2$ respectively.

**Function 1:** The Two Loops of WMMSE-BFGS

**Input:** $\mathbf{G}^0, \nabla_{\mathbf{V}_k} f(\mathbf{V}_k^t), \{\mathbf{S}^i\}_{i=1}^t, \{\mathbf{Y}^i\}_{i=1}^t, \{\mathbf{\Phi}^i\}_{i=1}^t$.

**for** $i = t, t-1, \ldots, 1$ **do**
  $\mathbf{A}^i = \mathbf{\Phi}^i \left(\mathbf{S}^i\right)^H \mathbf{Q}^{i+1}$;
  $\mathbf{Q}^i = \mathbf{Q}^{i+1} - \mathbf{Y}^i \mathbf{A}^i$;
**end**
$\mathbf{R}^0 = \mathbf{G}^0 \mathbf{Q}^1$;
**for** $p = 1, 2, \ldots, t$ **do**
  $\mathbf{R}^p = \mathbf{R}^{p-1} - \mathbf{S}^p \left(\mathbf{\Phi}^p \left(\mathbf{Y}^p\right)^H \mathbf{R}^{p-1} - \mathbf{A}^p\right)$;
**end**
**Return:** $\mathbf{D}^t = \mathbf{R}^t$.

Subsequently, defining $\mathbf{R}^0 \triangleq \mathbf{G}^0 \mathbf{Q}^1$, the update of $\mathbf{D}^t$ in (27) can be expressed as

$$\mathbf{D}^t = \left(\prod_{j=0}^{t-2} \mathbf{M}^{t-j}\right) \mathbf{M}^1 \mathbf{R}^0 + \left(\prod_{j=0}^{t-2} \mathbf{M}^{t-j}\right) \mathbf{S}^1 \mathbf{A}^1 + \sum_{i=2}^{t} \left(\prod_{j=0}^{t-1-i} \mathbf{M}^{t-j}\right) \mathbf{S}^i \mathbf{A}^i. \tag{29}$$

Then, by factoring out the common factor over the first two terms on the right-hand side of (29), we have

$$\mathbf{D}^t = \left(\prod_{j=0}^{t-2} \mathbf{M}^{t-j}\right) (\mathbf{M}^1 \mathbf{R}^0 + \mathbf{S}^1 \mathbf{A}^1) + \sum_{i=2}^{t} \left(\prod_{j=0}^{t-1-i} \mathbf{M}^{t-j}\right) \mathbf{S}^i \mathbf{A}^i. \tag{30}$$

Similarly, by defining

$$\mathbf{R}^p \triangleq \mathbf{M}^p \mathbf{R}^{p-1} + \mathbf{S}^p \mathbf{A}^p, \quad p = 1, 2, \ldots, t, \tag{31}$$

the update of $\mathbf{D}^t$ in (30) can be inductively expressed as

$$\begin{aligned} \mathbf{D}^t &= \left(\prod_{j=0}^{t-2} \mathbf{M}^{t-j}\right) \mathbf{R}^1 + \sum_{i=2}^{t} \left(\prod_{j=0}^{t-1-i} \mathbf{M}^{t-j}\right) \mathbf{S}^i \mathbf{A}^i \\ &= \ldots \\ &= \left(\prod_{j=0}^{t-p-1} \mathbf{M}^{t-j}\right) \mathbf{R}^p + \sum_{i=p+1}^{t} \left(\prod_{j=0}^{t-1-i} \mathbf{M}^{t-j}\right) \mathbf{S}^i \mathbf{A}^i \\ &= \ldots \\ &= \mathbf{M}^t \mathbf{R}^{t-1} + \mathbf{S}^t \mathbf{A}^t \\ &= \mathbf{R}^t, \quad p = 1, 2, \ldots, t. \end{aligned} \tag{32}$$

Interestingly, in this way, $\mathbf{D}^t$ can be directly obtained by the computation of $\mathbf{R}^t$, which can be iteratively calculated by (31) but with a more efficient formulation as

$$\mathbf{R}^p = \mathbf{R}^{p-1} - \mathbf{S}^p \left(\mathbf{\Phi}^p \left(\mathbf{Y}^p\right)^H \mathbf{R}^{p-1} - \mathbf{A}^p\right), p = 1, 2, \ldots, t. \tag{33}$$

Consequently, the computational cost of computing $\mathbf{D}^t$ in (29) is reduced to $3tML_k^2$ by decreasing the number of matrix multiplications and leveraging the low-dimensional matrices. On the other hand, to improve the performance, we apply the momentum technique to accelerate the convergence. Specifically, given the momentum coefficient $b \in [0, 1]$, the momentum variant of the update in (20) is given by

$$\mathbf{V}_k^{t+1} = \mathbf{V}_k^t - a \cdot \mathbf{D}^t + b\left(\mathbf{V}_k^t - \mathbf{V}_k^{t-1}\right), \ t \geqslant 1. \tag{34}$$

Besides, recalling $\mu$ in (16) and noting that the second term $\mu\mathbf{I}$ on the right-hand side of (11) is trivially invertible, we employ the more accurate initial approximation

$$\mathbf{G}^0 = \frac{1}{\mu}\mathbf{I}, \tag{35}$$

which further improves the convergence. To summarize, the WMMSE-BFGS algorithm is outlined in Algorithm 2.

**Algorithm 2:** The WMMSE-BFGS Algorithm

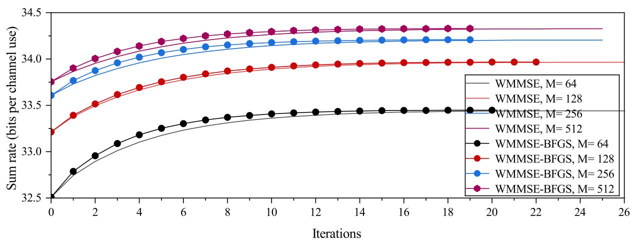
**Input:** $\mathbf{H}_k$, $\alpha_k$, $\sigma_k^2$, $P_{\max}$, $a$, $b$, iteration number $\mathcal{L}$ and accuracy $\epsilon$, $\forall k$.
**Initialize:** $\{\mathbf{V}_k\}_{k=1}^K$ with $\sum_{k=1}^K \mathrm{Tr}(\mathbf{V}_k \mathbf{V}_k^H) \leq P_{\max}$ and $\mathbf{W}_k = \mathbf{I}, \forall k$.

**repeat**
  update $\mathbf{U}_k = (\mathbf{N}_k + \mathbf{H}_k \mathbf{V}_k \mathbf{V}_k^H \mathbf{H}_k^H)^{-1} \mathbf{H}_k \mathbf{V}_k, \forall k$;
  update $\mathbf{W}_k = (\mathbf{I} - \mathbf{U}_k^H \mathbf{H}_k \mathbf{V}_k)^{-1}, \forall k$;
  compute $\mathbf{G}^0$ by (35);
  $\mathbf{V}_k^0 = \mathbf{V}_k$;
  **for** $k = 1 : K$ **do**
    compute $\nabla_{\mathbf{V}_k} f(\mathbf{V}_k^0)$ by (10);
    **for** $t = 0 : \mathcal{L} - 1$ **do**
      **if** $||\nabla_{\mathbf{V}_k} f(\mathbf{V}_k^t)||_F \leq \epsilon$ **then**
        **break**;
      **end**
      **if** $t = 0$ **then**
        update $\mathbf{V}_k^1$ by (20);
      **else**
        compute $\nabla_{\mathbf{V}_k} f(\mathbf{V}_k^t)$ by (10);
        compute $\mathbf{S}^t$, $\mathbf{Y}^t$ and $\mathbf{\Phi}^t$;
        compute $\mathbf{D}^t$ by Function 1;
        update $\mathbf{V}_k^{t+1}$ by (34);
      **end**
    **end**
    $\mathbf{V}_k = \mathbf{V}_k^{\mathcal{L}}$;
  **end**
**until** convergence;
**Output:** $\bar{\mathbf{V}}_k = \sqrt{w}\mathbf{V}_k, \forall k$.

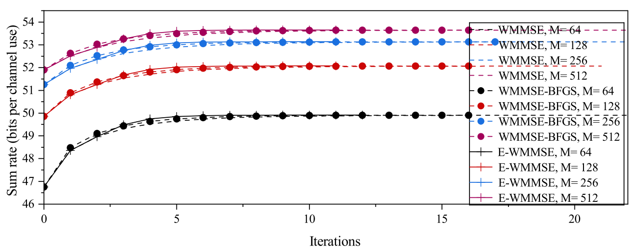
Now, we examine the complexity of the proposed WMMSE-BFGS for solving the problem in (7). For simplicity, we assume $L_k = L$ and $N_k = N, \forall k$. In particular, at each iteration of BCD, the overall computation of $\{\mathbf{D}^t\}_{t=1}^{\mathcal{L}-1}$ in Function 1 needs $3KML^2\mathcal{L}(\mathcal{L}-1) + KML(\mathcal{L}-1)$ multiplications. Moreover, the computation of $\{\nabla_{\mathbf{V}_k} f(\mathbf{V}_k^t)\}_{t=0}^{\mathcal{L}-1}$ in (10) requires $K\left(2MNL + 3ML^2 + ML\right) + (K-1)(KMNL + 2KML^2) + K^2(MNL + 2ML^2)(\mathcal{L}-1) + KML\mathcal{L}$ multiplications. The complexities of updating $\mathbf{V}_k^1$ in (20) and $\{\mathbf{V}_k^{t+1}\}_{t=1}^{\mathcal{L}-1}$ in (34) are $KML + 2KML(\mathcal{L}-1)$ multiplications. Besides, the computation of $\{\mathbf{\Phi}^t\}_{t=1}^{\mathcal{L}-1}$ in (14) requires

TABLE I
THE COMPUTATIONAL COMPLEXITY OF WMMSE-BFGS AND WMMSE IN A BCD ITERATION WITH RESPECT TO THE PROBLEM IN (7).

| Algorithm | Computational Complexity | $M=256,\ \mathcal{L}=5$ | $M=512,\ \mathcal{L}=6$ |
|---|---|---|---|
| WMMSE | $2M^3+2KM^2L+KM^2+2KML^2+KMNL$ $+KML+KN^2L+KNL^2$ [4] | $3.48\times10^7$ | $2.73\times10^8$ |
| WMMSE-BFGS | $K^2MNL\mathcal{L}+2K^2ML^2\mathcal{L}+KML^2(3\mathcal{L}^2-2\mathcal{L})+KMNL$ $+KML(4\mathcal{L}-1)+KN^2L+KNL^2+2KL^3(\mathcal{L}-1)$ | $8.27\times10^5$ | $2.28\times10^6$ |




(a) $K=2$




(b) $K=4$

Fig. 1. The convergence performance of the WMMSE-BFGS algorithm.

$K\left(ML^2+2L^3\right)(\mathcal{L}-1)$ multiplications, while $\mathbf{G}^0$ in (35) requires $K\left(N^2L+NL^2\right)$ multiplications. Consequently, the complexity of WMMSE-BFGS is as shown in Table I.

## V. SIMULATION

We consider a BS with $M$ transmit antennas and $K$ users each equipped with $N=4$ receive antennas in a massive MIMO system. Each user has $L=4$ transmit streams and equal priority $\alpha=1$. Besides, the channel $\mathbf{H}_k,\forall k$ follows the circularly-symmetric standard complex normal distribution, where the pathloss is $128.1+37.6\log_{10}(d)$ [dB] with distance $d\,(\text{km})\in[0.2,0.3]$ between the BS and the users. The equal noise power is $\sigma^2=10^{\frac{1}{K}\sum_k\log_{10}\frac{1}{N}\|\mathbf{H}_k\|_F^2}\times10^{-\frac{\text{SNR}}{10}},\forall k$, where $\text{SNR}=10\,[\text{dB}]$ is the average receive SNR for all users without precoding. The maximum power of BS is $P=15\,[\text{W}]$.

Fig.1 mainly compares the convergence performance of WMMSE-BFGS and WMMSE under different transmit antennas $M=[64,128,256,512]$, both initialized with the ZF precoder. In contrast, Fig.1(a) considers $K=2$ users, while Fig.1(b) considers $K=4$ users. Specifically, the simulation is performed in MATLAB with maximum quasi-Newton iterations $\mathcal{L}=[8,7,5,6]$ and momentum coefficient $b=0.4$. The step size is $a=[0.04,0.03,0.04,0.04]$ for $K=2$ and $a=[0.06,0.06,0.07,0.07]$ for $K=4$. In addition, the convergence and performance are sensitive to the choice of these parameters. Notably, these parameters can be learned via deep unfolding [10], reducing the required quasi-Newton iterations and thereby further lowering the complexity. Additionally, we compare the numerical complexity in Table I. From Fig.1, WMMSE-BFGS requires less iterations when reaches the same WSR. Meanwhile, based on the comparison of numerical complexity in Table I, WMMSE-BFGS achieves a reduction in complexity relative to WMMSE.

Moreover, Fig.1(b) also compares WMMSE-BFGS with E-WMMSE proposed in [5]. For fairness, we adapt E-WMMSE to the problem (5), which still requires $\mathcal{O}(M^3)$ matrix inversions. Hence, E-WMMSE is included as a baseline for assessing the convergence speed of our proposed algorithm.


## ACKNOWLEDGMENT

This work was supported in part by the National Key R&D Program of China under Grants No.2023YFC2205501, and in part by National Natural Science Foundation of China under Grants No.62371124.


## REFERENCES


[1] Z. Gao, M. Ke, Y. Mei, L. Qiao, S. Chen, D. W. K. Ng, and H. V. Poor, "Compressive-sensing-based grant-free massive access for 6G massive communication," *IEEE Internet of Things Journal*, vol. 11, no. 5, pp. 7411–7435, 2024.

[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 Transactions on Signal Processing*, vol. 59, no. 9, pp. 4331–4340, 2011.

[3] X. Zhao, S. Lu, Q. Shi, and Z.-Q. Luo, "Rethinking WMMSE: Can its complexity scale linearly with the number of BS antennas?" *IEEE Transactions on Signal Processing*, vol. 71, pp. 433–446, 2023.

[4] L. Pellaco and J. Jaldén, "A matrix-inverse-free implementation of the MU-MIMO WMMSE beamforming algorithm," *IEEE Transactions on Signal Processing*, vol. 70, pp. 6360–6375, 2022.

[5] K. Zhou, Z. Chen, G. Liu, and Z. Chen, "A novel extrapolation technique to accelerate WMMSE," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[6] M. Sun, Z. Li, S. Wu, R. Ma, and L. Jiang, "A matrix-inverse-free WMMSE algorithm to MISO beamforming based on quasi-Newton," *IEEE Communications Letters*, vol. 28, no. 12, pp. 2809–2813, 2024.

[7] R. B. Schnabel, "Quasi-Newton methods using multiple secant equations," University of Colorado at Boulder, Department of Computer Science, Technical Report, 1983.

[8] G. Scutari, F. Facchinei, J.-S. Pang, and D. P. Palomar, "Real and complex monotone communication games," *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 4197–4231, 2014.

[9] A. Uschmajew and B. Vandereycken, "A note on the optimal convergence rate of descent methods with fixed step sizes for smooth strongly convex functions," *Journal of Optimization Theory and Applications*, vol. 194, no. 1, pp. 364–373, 2022.

[10] G. Yiqun, C. Qiqiang, H. Lanxin, W. Zheng, H. Yongming, Z. Jianjun, G. Yu, and X. Zhikun, "Artificial intelligence enabled joint channel estimation and signal detection for massive MIMO systems," *Chinese Journal of Electronics*, vol. 35, no. 1, pp. 178–195, 2026.