RIS-aided Beamforming Design for MIMO Systems via Unified Manifold Optimization

Kai Zhong, Jinfeng Hu, Huiyong Li, Ren Wang, Dongxu An, Gangyong Zhu, Kah Chan Teh, *IEEE Senior Member*, Cunhua Pan, *IEEE Senior Member*, and Yonina C. Eldar, *Fellow, IEEE*

***Abstract*—Reconfigurable Intelligent Surface (RIS) can enhance spectral efficiency in Multiple-Input Multiple-Output (MIMO) communication systems. A specific challenge in this context involves codesigning the nonconvex phase shifts for RIS and a precoding matrix with a complex sphere constraint, where these coupled variables are used to formulate the nonconvex objective of maximizing spectral efficiency. Most existing methods do not directly address the maximization of spectral efficiency problem but instead relax it into a sum-path-gain-maximization (SPGM) problem before solving it, which may degrade spectral efficiency due to large relaxation gap. We propose an efficient Unified Manifold Optimization (UMO) framework to directly solve the problem. This is achieved through utilizing the inherent constant modulus characteristic of the RIS and the complex sphere characteristic of the precoding matrix constraint. Specifically, we construct a unified manifold space (UMS) that can simultaneously satisfy the RIS and the precoding matrix constraints, enabling the problem to be rephrased as an unconstrained Riemannian problem over the UMS. Based on the UMS, we derive a parallel conjugate gradient algorithm for simultaneous optimization of a precoding matrix and RIS phase shifts. Simulation outcomes indicate that the proposed method excels when compared to current approaches in achieving spectral efficiency enhancement. Furthermore, our algorithm has lower computational cost than several existing techniques.**

***Index Terms*—Reconfigurable intelligent surfaces (RIS), MIMO, Spectral efficiency, Beamforming design, Unified Manifold Optimization, Parallel conjugate gradient algorithm**

This work was supported by the National Natural Science Foundation of China (NO.62231006), National Key R&D Program of China (2023YFF0717303) and the Municipal Government of Quzhou (NO.2023D040, 2023D009, 2022D009, 2022D013, and 2022D033). Sichuan Science and Technology Program (No. 2023YFG0176). *Jinfeng Hu and Cunhua Pan are Corresponding authors*.

Kai Zhong, Huiyong Li, Dongxu, An and Gangyong Zhu are with the School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu, 611731, China. They are also with the Yangtze Delta Region Institute (Quzhou), University of Electronic Science and Technology of China Quzhou, Zhejiang, China. (e-mail: 201921011206@std.uestc.edu.cn; hyli@uestc.edu.cn; 202311012301@std.uestc.edu.cn; 202122011909@std.uestc.edu.cn).

Jinfeng Hu is with Yibin Institute of UESTC, Yibin, China, and also with the School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu, 611731, China. (e-mail: hujf@uestc.edu.cn).

Ren Wang is with the China Academy of Aerospace Science and Innovation, Beijing, 100088, China. (wangren94@126.com).

Kah Chan Teh is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. (Email: ekcteh@ntu.edu.sg).

Cunhua Pan is with the National Mobile Communications Research Laboratory, Southeast University, China. (Email: cpan@seu.edu.cn).

Yonina C. Eldar is with the Faculty of Math and CS, Weizmann Institute of Science, Rehovot, 7610001, Israel (email: yonina.eldar@weizmann.ac.il).

I. Introduction

Conventional large-scale Multiple-Input Multiple-Output (MIMO) transmit beamforming is extensively employed in wireless communications for enhancing spectral efficiency [1]–[3]. However, its large-scale implementation requires specific radio frequency chains, resulting in significant hardware cost. Additionally, the increasing number of antennas contributes to higher power consumption [4].

In response to the aforementioned issues, Reconfigurable Intelligent Surfaces (RIS)-aided MIMO beamforming design has been proposed, utilizing cost-effective passive reflecting elements [5]–[8]. The primary purpose of RIS is to enhance wireless communication systems by offering additional control over signal propagation [9]–[13]. By adjusting signal reflection and phase, RIS can shape and steer wireless waves in desired directions [14]–[16]. RIS-aided beamforming helps mitigate path loss, multipath fading, and interference, leading to improved spectral efficiency. Therefore, RIS-aided MIMO beamforming design for spectral efficiency maximization has garnered widespread attention [17]–[20].

The problem is typically expressed as maximizing spectral efficiency while maintaining constant modulus constraints on RIS phase shifts and a complex sphere constraint on a precoding matrix [17]. The coupled effect of phase shifts and precoding matrix makes the problem challenging to solve. Furthermore, the nonconvex characteristic of the constant modulus constraints further exacerbates the complexity.

Most existing methods do not directly address the maximization of spectral efficiency problem but instead relax it into a sum-path-gain-maximization (SPGM) problem before solving it, and solve the SPGM problem with two-stage algorithms [17]–[20]. Specifically, in the first stage, optimize the RIS phase shifts for SPGM problem. Subsequently, the precoding matrix is acquired through singular value decomposition (SVD) with the specified RIS phase shifts. Typically, authors in [17] have derived an Alternating Direction Method of Multipliers (ADMM) framework to relax the SPGM problem by introducing several auxiliary variables. The Majorization Minimization (MM) approch was also used [18], [19]. To solve SPGM directly, in [20], the authors suggested a Gradient Ascent (GA) algorithm by deriving the phase-based gradient with a constant step size. The Dimension-wise Sinusoidal Maximization (DSM) method based on block coordinate descent was proposed in [20], [21], by exploiting the fact that the objective of RIS is sinusoidal for each RIS phase-shift. DSM method may be computationally prohibitive as the size

of the RIS phase shifts large. To reduce the computational cost, the Riemannian conjugate gradient (RCG) algorithm based on complex circle manifold (CCM) can be used to solve the problem, using the characteristic of CCM naturally satisfies the constant modulus constraint [22]. Nevertheless, in the context of MIMO communication, optimizing SPGM by the above algorithms only provides a lower bound for spectral efficiency. An upper limit of spectral efficiency is determined by the product of SPGM upper bound and the number of data streams [17]. Comparing direct optimization of spectral efficiency maximization with SPGM, the relaxation gap of SPGM lies between zero and the product of the maximum SPGM bound and the number of data streams minus one. As the number of data streams increases, the error bound becomes larger, indicating that SPGM may not effectively capture the characteristics of spectral efficiency maximization.

Here, we directly solve the spectral efficiency maximization problem by a parallel optimization framework. In our approach, both the phase shifts and precoding matrix are simultaneously optimized. Parallel implementation leads to faster convergence for problems with a large number of variables or complex structures, as it tackles multiple parts of the problem concurrently. Within our framework, we rely on the gradient projection (GP) algorithm [23], [24]. GP operates in Euclidean space, where it computes a Euclidean gradient of the nonconvex objective function, performs descent steps, and then projects the result onto the nonconvex feasible set. The direct projection from Euclidean space to a non-convex complex space may lead to potential performance degradation. To enhance the performance, we propose a Unified Manifold Optimization (UMO) method. Specifically, we construct a unified manifold space (UMS) designed to satisfy the RIS constraint and precoding matrix constraint, enabling the problem to be reformulated as an unconstrained Riemannian problem over the UMS. We then leverage the fact that the UMS can be locally linearized as a tangent space around every point. Based on this tangent space, we derive a parallel conjugate gradient algorithm to simultaneously update the RIS phase shifts and precoding matrix. By aligning its update steps with the structure of the UMS, UMO adapts to local features of the optimization landscape. This adaptability to local features may allow UMO to escape from poor local optima and converge more effectively towards global or better solutions.

Our main contribution is threefold. First, in contrast to relaxing the spectral efficiency maximization into an SPGM problem with a large relaxation gap [17]–[20], we directly solve the spectral efficiency maximization problem using the proposed parallel optimization framework. Second, different from GP, which performs a direct projection from Euclidean space to a complex non-convex space without adapting to the local features of the complex non-convex optimization landscape [23], [24], the UMO method is proposed to adapt to local features. This adaptability aligns its update steps with the structure of the UMS and allows UMO to converge more effectively towards better solutions. Third, in simulations, our algorithm exhibits the following advantages when compared to existing methods: 1) the proposed method achieves a 10.12 bits/Hz/s higher gain than ADMM [17], a 3.94 bits/Hz/s higher gain than the methods in [18]–[20], and a 1.65 bits/Hz/s higher gain than GP [23], [24]; 2) the execution time is about 2 orders of magnitude lower than GA [20], about 1 order of magnitude lower than DSM [20], and about 0.5 order of magnitude lower than GP [23].

The subsequent sections are organized as follows. The problem description is outlined in Section II. Our proposed UMO method, aimed at efficient problem resolution, is detailed in Section III. We present the simulation results in Section IV. In Section V, we give a conclusion.

Notation: Operators $(\cdot)^H$, $(\cdot)^*$, and $(\cdot)^T$ indicate the conjugate transpose, the conjugate and the transpose operations, respectively. Vector variable is denoted by lower-case bold letter, and matrix variable is denoted by upper-case bold letter. $|\cdot|$ is the modulus of a variable, and $\|\cdot\|_F$ is the Frobenius norm. We use $diag(\cdot)$ and $vec(\cdot)$ as the diagonal operator and vectorization operator, respectively. Operators $\otimes$ and $\odot$ indicate Kronecker product and Hadamard product, respectively.

II. Problem Description

The RIS-aided point-to-point MIMO communication system model is illustrated in Figure 1. The base station (BS) of N_t antennas sends N_s parallel data streams to the user with N_b antennas, where $N_s \geq N_b$ and $N_s \leq N_t$ in general. The communication system utilizes an RIS with N_r elements to assist in the transmission process. The BS processes the transmitted signal through a precoding matrix, denoted $\mathbf{W} \in \mathbb{C}^{N_t \times N_s}$, and simultaneously transmits it to both the RIS and the user. The incident signal is passively reflected by the RIS through

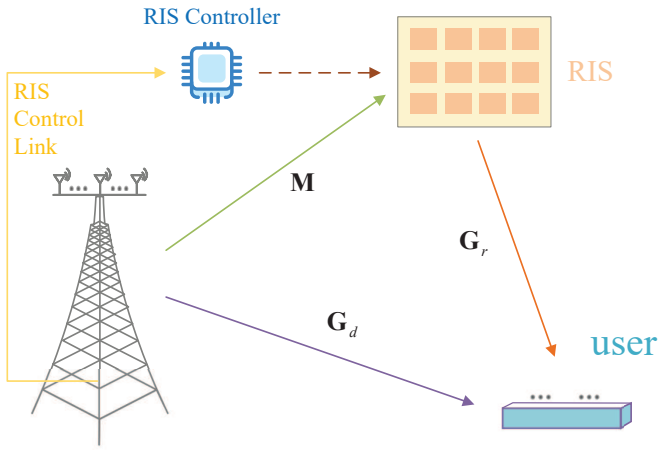


Fig. 1: RIS-aided point-to-point MIMO System Model.

the adjustment of phases $\theta_i \in [-\pi, \pi), i = 1, 2, \ldots, N_r$. The matrix of RIS phase shifts is [9], [12], [25], [26]

$$\mathbf{X} = \text{diag}(\boldsymbol{x}) = \text{diag}(e^{j\theta_1}, e^{j\theta_2}, \ldots, e^{j\theta_{N_r}}) \in \mathbb{C}^{N_r \times N_r}. \quad (1)$$

Let $\mathbf{G}_r \in \mathbb{C}^{N_b \times N_r}$, $\mathbf{M} \in \mathbb{C}^{N_r \times N_t}$ and $\mathbf{G}_d \in \mathbb{C}^{N_b \times N_t}$ respectively be the channels from the RIS to the user, the BS to the RIS, and the BS to the user. The received signal of the user is

$$\boldsymbol{y} = \sqrt{P/N_s}\,(\mathbf{G}_r\mathbf{X}\mathbf{M}\mathbf{W}\boldsymbol{s} + \mathbf{G}_d\mathbf{W}\boldsymbol{s}) + \boldsymbol{n}, \quad (2)$$

where $\mathbf{W}$ is the beamforming matrix that satisfies a power constraint $\|\mathbf{W}\|_F^2 = N_s$ at the source, and P represents transmitted power. The transmitted data streams are denoted as $\boldsymbol{s} \in \mathbb{C}^{N_s}$, with the condition $\mathbb{E}\left[\mathbf{s}\mathbf{s}^H\right] = \mathbf{I}_{N_s}$. Additionally, $\mathbf{n} \sim \mathcal{CN}(0, \sigma_n^2 \mathbf{I}_{N_b})$ is the zero-mean additive complex Gaussian noise.

We assume that the channel state information (CSI) is known [17], [20], [27], [28]. The problem is typically formulated as the maximization of spectral efficiency with a constant modulus constraint on the RIS phase shifts and a power constraint on a precoding matrix [17]. It is formulated as

$$\begin{aligned} \max_{\bar{\mathbf{W}}, \boldsymbol{x}} \quad & C(\bar{\mathbf{W}}, \boldsymbol{x}) = \log_2 \det\left(\mathbf{I}_{\mathrm{N_s}} + \frac{P}{\sigma_n^2 N_{\mathrm{s}}} \mathbf{G}_{\mathrm{eff}} \bar{\mathbf{W}} \bar{\mathbf{W}}^H \mathbf{G}_{\mathrm{eff}}^H\right) \\ s.t. \quad & |x_{n_r}| = 1, n_r = 1, \ldots, N_r, \\ & \left\|\bar{\mathbf{W}}\right\|_F^2 = N_s, \end{aligned} \tag{3}$$

where $\boldsymbol{x} = \left[e^{j\theta_1}, e^{j\theta_2}, \ldots, e^{j\theta_{N_r}}\right]^T \in \mathbb{C}^{N_r}$ represents the RIS phase shifts. $\mathbf{G}_{\mathrm{eff}} = \mathbf{G}_d + \mathbf{G} r \mathbf{X} \mathbf{M}$ denotes the effective channel. Equations (4b) and (4c) are the constant modulus constraints and the power constraint, respectively.

Problem (3) can be equivalently reformulated as

$$\min_{\mathbf{W}, \boldsymbol{x}} \quad -C(\mathbf{W}, \boldsymbol{x}) = -\log_2 \det\left(\mathbf{I}_{\mathrm{N_s}} + \frac{P}{\sigma_n^2} \mathbf{G}_{\mathrm{eff}} \mathbf{W} \mathbf{W}^H \mathbf{G}_{\mathrm{eff}}^H\right) \tag{4a}$$

$$s.t. \quad |x_{n_r}| = 1, n_r = 1, \ldots, N_r, \tag{4b}$$

$$\|\mathbf{W}\|_F^2 = 1, \tag{4c}$$

where $\|\mathbf{W}\|_F^2 = 1$ is the complex sphere constraint.

Due to the coupling effect between the precoding matrix $\mathbf{W}$ and the RIS phase shifts $\boldsymbol{x}$, solving (4) is challenging. The nonconvex charateristic of the constant modulus constraints on RIS phase shifts further increases the difficulty [29]–[34]. Typical methods used to address (4) are two-stage algorithms that rely on relaxing the spectral efficiency maximization into an SPGM problem [17]–[20]. However, these techniques may result in performance loss due to the objective function relaxation process. Furthermore, the existing methods that utilize manifold optimization primarily focus on single manifold constraints. These methods are effective for problems where the optimization variables lie on a single manifold, such as the complex circle manifold [22], [35]. However, they cannot directly handle problems involving multiple manifold variables.

Instead, we construct a Unified Manifold Space (UMS) that can simultaneously satisfy the constant modulus (CM) constraints of the RIS and the complex sphere constraint of the precoding matrix in (4). We then propose a Unified Manifold Optimization (UMO) framework, which is adaptable to the local features of the complex non-convex optimization landscape and does not rely on objective function relaxation, to address (4).

III. The Proposed UMO Method

To address (4), we propose the UMO framework. Specifically, we construct an UMS designed to satisfy the RIS constraint and precoding matrix constraint, enabling the problem to be reformulated as an unconstrained Riemannian problem over the UMS. We then leverage the characteristic that the UMS can be locally linearized as a tangent space around every point. Based on this tangent space, we derive a parallel conjugate gradient algorithm to simultaneously update the RIS phase shifts and precoding matrix.

A. Space Construction

1) Construction of UMS: In general, a constrained optimization problem can be transformed into an unconstrained optimization problem by designing a manifold space that incorporates all the required constraints. For (4), the solution space subject to CM constraint (4b) and complex sphere constraint (4c) can be represented as the UMS. This space is constructed by the product of the complex circle manifold (CM constraint) and the complex sphere manifold (complex sphere constraint).

The complex circle manifold for (4b) is

$$\mathcal{M}_{\boldsymbol{x}} = \left\{\boldsymbol{x} \in \mathbb{C}^{N_r}, s.t.\ |x_{n_r}| = 1, n_r = 1, \ldots, N_r.\right\}. \tag{5}$$

Similar to (4), the complex sphere manifold for (4c) is

$$\mathcal{M}_{\mathbf{W}} = \left\{\mathbf{W} \in \mathbb{C}^{N_t \times N_s}, s.t.\ \|\mathbf{W}\|^2 = 1\right\}. \tag{6}$$

The UMS $\mathcal{M}$ incorporating all the required constraints of (3) is denoted as the product of $\mathcal{M}_{\boldsymbol{x}}$ and $\mathcal{M}_{\mathbf{W}}$,

$$\begin{aligned} &\mathcal{M} = \mathcal{M}_{\boldsymbol{x}} \times \mathcal{M}_{\mathbf{W}} \\ &= \left\{(\boldsymbol{x}, \mathbf{W}), s.t.\ \boldsymbol{x} \in \mathcal{M}_{\boldsymbol{x}}, \mathbf{W} \in \mathcal{M}_{\mathbf{W}}\right\} \\ &= \left\{(\boldsymbol{x}, \mathbf{W}), s.t.\ |x_{n_r}| = 1, \|\mathbf{W}\|^2 = 1\right\}, \end{aligned} \tag{7}$$

where the dimension of $\mathcal{M}$ is $(N_r + N_t \times N_s)$.

Based on (7), problem (4) can be reformulated as an unconstrained coupled optimization problem over the UMS:

$$\begin{aligned} \min \quad & -C(\mathbf{W}, \boldsymbol{x}) = -\log_2 \det\left(\mathbf{I}_{\mathrm{N_s}} + \frac{P}{\sigma_n^2} \mathbf{G}_{\mathrm{eff}} \mathbf{W} \mathbf{W}^H \mathbf{G}_{\mathrm{eff}}^H\right) \\ s.t. \quad & (\boldsymbol{x}, \mathbf{W}) \in \mathcal{M} \end{aligned}. \tag{8}$$

2) Construction of unified tangent space: Given that the UMS space is curved, operations such as gradient calculations cannot be directly applied to (8). In order to enable these operations, we approximate the curved space with the unified tangent space (UTS) $\mathrm{T}_{(\boldsymbol{x}, \mathbf{W})}\mathcal{M}$ at the current point. UTS can be obtained by taking the product of the tangent space of $\mathcal{M}_{\boldsymbol{x}}$ (5) and the tangent space of $\mathcal{M}_{\mathbf{W}}$ (6).

Let x_{n_r} be an arbitrary element of $\boldsymbol{x}$. The tangent space of $\boldsymbol{x}$ at the point $x_{n_r} \in \boldsymbol{x}$ is defined as $\mathrm{T}_{x_{n_r}}\mathcal{M}_{\boldsymbol{x}} = \left\{\xi_{x_{n_r}} \in \mathbb{C} : \Re\left\{\xi_{x_{n_r}} x_{n_r}^*\right\} = 0\right\}$ [36], [37]. The tangent space of $\mathcal{M}_{\boldsymbol{x}}$ (5) is the product of N_r tangent spaces of those for the manifold, which is

$$\begin{aligned} \mathrm{T}_{\boldsymbol{x}}\mathcal{M}_{\boldsymbol{x}} &= \mathrm{T}_{x_1}\mathcal{M}_{\boldsymbol{x}} \times \mathrm{T}_{x_2}\mathcal{M}_{\boldsymbol{x}} \times \cdots \times \mathrm{T}_{x_{N_r}}\mathcal{M}_{\boldsymbol{x}} \\ &= \left\{\xi_{\boldsymbol{x}} \in \mathbb{C}^{N_r} : \Re\left\{\xi_{\boldsymbol{x}} \odot \boldsymbol{x}^*\right\} = \boldsymbol{0}_{N_r}\right\}, \end{aligned} \tag{9}$$

where $\xi_{\boldsymbol{x}}$ is the tangent vector at point $\boldsymbol{x}$. $\Re\left\{\xi_{\boldsymbol{x}} \odot \boldsymbol{x}^*\right\} = \boldsymbol{0}_{N_r}$ means that the elements at the corresponding positions of $\xi_{\boldsymbol{x}}$ and $\boldsymbol{x}^*$ are orthogonal to each other. The tangent space of $\mathcal{M}_{\mathbf{W}}$ (6) is given by

$$\mathrm{T}_{\mathbf{W}}\mathcal{M}_{\mathbf{W}} = \left\{\xi_{\mathbf{W}} \in \mathbb{C}^L : \Re\left\{Tr(\xi_{\mathbf{W}}^T \mathbf{W}^*)\right\} = 0\right\}, \tag{10}$$

where $L = N_t \times N_s$, and $\xi_{\mathbf{W}}$ is the tangent matrix at point $\mathbf{W}$. $\Re\left\{Tr(\xi_{\mathbf{W}}^T \mathbf{W}^*)\right\} = 0$ represents that $\xi_{\mathbf{W}}$ and $\mathbf{W}^*$ are orthogonal to each other. Using (9) and (10), the UTS $\mathrm{T}_{(\boldsymbol{x},\mathbf{W})}\mathcal{M}$ is obtained by

$$\begin{aligned} &\mathrm{T}_{(\boldsymbol{x},\mathbf{W})}\mathcal{M} \\ &= \mathrm{T}_{\boldsymbol{x}}\mathcal{M}_{\boldsymbol{x}} \times \mathrm{T}_{\mathbf{W}}\mathcal{M}_{\mathbf{W}} \\ &= \left\{ \begin{array}{l} (\xi_{\boldsymbol{x}}, \xi_{\mathbf{W}}) : \xi_{\boldsymbol{x}} \in \mathbb{C}^{N_r}, \Re\left\{\xi_{\boldsymbol{x}} \odot \boldsymbol{x}^*\right\} = \boldsymbol{0}_{N_r}, \\ \xi_{\mathbf{W}} \in \mathbb{C}^{N_t \times N_s}, \Re\left\{Tr(\xi_{\mathbf{W}}^T \mathbf{W}^*)\right\} = 0 \end{array} \right\}. \end{aligned} \tag{11}$$

Based on UMS (7) and UTS (11), the gradient projection (GP) algorithm in [23], [24] can be employed to solve (8) using parallel gradient descent while projecting onto the constraint set. However, GP typically relies on Euclidean geometry, which may not be as adaptable to the curved space of the problem. Additionally, GP's projection step may prioritize constraint satisfaction, potentially restricting exploration and making it more vulnerable to local optima in complex non-convex landscapes. To address these issues, we develop a parallel conjugate gradient algorithm for efficiently exploring the entire solution space, as presented in the next subsection.

B. The proposed PCG algorithm

The PCG algorithm is proposed to solve (8) in this subsection, where the precoding matrix $\mathbf{W}$ and RIS phase shifts $\boldsymbol{x}$ are simultaneously optimized. The proposed PCG algorithm primarily consists of four steps: 1) Calculation of the parallel Euclidean Gradient; 2) Calculation of the parallel Riemannian Gradient; 3) Update of the feasible solution; 4) Adaptive update of the step size using Armijo linear search.

1) Derive the parallel Euclidean gradient: The parallel Euclidean gradient of $C(\boldsymbol{x}, \mathbf{W})$ (8) is the combination of the Euclidean gradients w.r.t. $\boldsymbol{x}^*$ and $\mathbf{W}^*$.

The gradient of $C(\boldsymbol{x}, \mathbf{W})$ at $\boldsymbol{x}^*$ is

$$\begin{aligned} &\nabla_{\boldsymbol{x}^*} C(\boldsymbol{x}, \mathbf{W}) \\ &= \mathrm{vec}_d\left(\mathbf{G}_r^H \mathbf{K}(\boldsymbol{x}, \mathbf{W}) \mathbf{G}_{\mathrm{eff}} \mathbf{W} \mathbf{W}^H \mathbf{M}^H\right), \\ &= \mathrm{diag}\left(\mathbf{G}_r^H \mathbf{K}(\boldsymbol{x}, \mathbf{W}) \mathbf{G}_{\mathrm{eff}} \mathbf{W} \mathbf{W}^H \mathbf{M}^H\right) \end{aligned} \tag{12}$$

where

$$\mathbf{K}(\boldsymbol{x}, \mathbf{W}) = (\mathbf{I} + \mathbf{G}_{\mathrm{eff}} \mathbf{W} \mathbf{W}^H \mathbf{G}_{\mathrm{eff}}^H)^{-1}. \tag{13}$$

Similar to (12), the gradient of $C(\boldsymbol{x}, \mathbf{W})$ at $\mathbf{W}^*$ is

$$\nabla_{\mathbf{W}^*} C(\boldsymbol{x}, \mathbf{W}) = \mathbf{G}_{\mathrm{eff}}^H \mathbf{K}(\boldsymbol{x}, \mathbf{W}) \mathbf{G}_{\mathrm{eff}} \mathbf{W}. \tag{14}$$

Combining (12) and (14), the parallel Euclidean gradient at k-th iteration is

$$\begin{aligned} &\nabla C(\boldsymbol{x}_k, \mathbf{W}_k) \\ &= \left[\nabla_{\boldsymbol{x}_k^*} C(\boldsymbol{x}_k, \mathbf{W}_k), \nabla_{\mathbf{W}_k^*} C(\boldsymbol{x}_k, \mathbf{W}_k)\right] \\ &= \left[\begin{array}{l} \mathrm{diag}\left(\mathbf{G}_r^H \mathbf{K}(\boldsymbol{x}_k, \mathbf{W}_k) \mathbf{G}_{\mathrm{eff}} \mathbf{W}_k \mathbf{W}_k^H \mathbf{M}_k^H\right); \\ \mathbf{G}_{\mathrm{eff}}^H \mathbf{K}(\boldsymbol{x}_k, \mathbf{W}_k) \mathbf{G}_{\mathrm{eff}} \mathbf{W}_k \end{array} \right]. \end{aligned} \tag{15}$$

Proof: Please refer to the Appendix A.

2) Derive the Parallel Riemannian Gradient: The parallel Riemannian gradient of $C(\boldsymbol{x}_k, \mathbf{W}_k)$ (7) is the combination of the Riemannian gradients w.r.t. $\boldsymbol{x}_k^*$ and $\mathbf{W}_k^*$. Riemannian gradient is defined by orthogonally projecting Euclidean gradient onto the tangent space. According to this, the Riemannian gradient of $C(\boldsymbol{x}_k, \mathbf{W}_k)$ (8) w.r.t. $\boldsymbol{x}_k^*$ is given by

$$\begin{aligned} &\mathrm{grad}_{\boldsymbol{x}_k} C(\boldsymbol{x}_k, \mathbf{W}_k) \\ &= \mathrm{Proj}_{\boldsymbol{x}_k}(\nabla_{\boldsymbol{x}_k} C(\boldsymbol{x}_k, \mathbf{W}_k)) \\ &= \left(\begin{array}{l} \nabla_{\boldsymbol{x}_k} C(\boldsymbol{x}_k, \mathbf{W}_k) - \\ \Re\left\{\nabla_{\boldsymbol{x}_k} C(\boldsymbol{x}_k, \mathbf{W}_k) \odot \boldsymbol{x}_k^*\right\} \odot \boldsymbol{x}_k \end{array} \right), \end{aligned} \tag{16}$$

where $\mathrm{Proj}_{\boldsymbol{x}}(\cdot)$ is the orthogonal projection. Similar to (16), the Riemannian gradient of $C(\boldsymbol{x}_k, \mathbf{W}_k)$ (8) w.r.t. $\mathbf{W}_k^*$ is

$$\begin{aligned} &\mathrm{grad}_{\mathbf{W}_k} C(\boldsymbol{x}_k, \mathbf{W}_k) \\ &= \mathrm{Proj}_{\mathbf{W}_k}(\nabla_{\mathbf{W}_k} C(\boldsymbol{x}_k, \mathbf{W}_k)) \\ &= \left(\begin{array}{l} \nabla_{\mathbf{W}_k} C(\boldsymbol{x}_k, \mathbf{W}_k) - \\ \mathbf{W}_k \Re\left\{Tr(\mathbf{W}_k^H \nabla_{\mathbf{W}_k} C(\boldsymbol{x}_k, \mathbf{W}_k))\right\} \end{array} \right). \end{aligned} \tag{17}$$

Combining (16) and (17), the parallel Riemannian gradient of $C(\boldsymbol{x}_k, \mathbf{W}_k)$ on the UTS is

$$\begin{aligned} &\mathrm{grad}\, C(\boldsymbol{x}_k, \mathbf{W}_k) \\ &= [\mathrm{grad}_{\boldsymbol{x}_k} C(\boldsymbol{x}_k, \mathbf{W}_k); \mathrm{grad}_{\mathbf{W}_k} C(\boldsymbol{x}_k, \mathbf{W}_k)] \\ &= \left[\begin{array}{l} \left(\begin{array}{l} \nabla_{\boldsymbol{x}_k} C(\boldsymbol{x}_k, \mathbf{W}_k) - \\ \Re\left\{\nabla_{\boldsymbol{x}_k} C(\boldsymbol{x}_k, \mathbf{W}) \odot \boldsymbol{x}_k^*\right\} \odot \boldsymbol{x}_k \end{array} \right); \\ \left(\begin{array}{l} \nabla_{\mathbf{W}_k} C(\boldsymbol{x}_k, \mathbf{W}_k) - \\ \mathbf{W}_k \Re\left\{Tr(\mathbf{W}_k^H \nabla_{\mathbf{W}_k} C(\boldsymbol{x}_k, \mathbf{W}_k))\right\} \end{array} \right) \end{array} \right]. \end{aligned} \tag{18}$$

3) Update the feasible solution: In general, the steepest gradient descent method is applicable for addressing unconstrained optimization problems across $\mathcal{M}$. However, its practical convergence is impeded by its relatively slow rate and susceptibility to getting trapped in local minima.

The optimization procedure for nonconvex problems resembles the movement of a particle on a multi-dimensional plane. This suggests that the effective descent direction $\mathbf{d}_{(\boldsymbol{x}_k, \mathbf{W}k)} = [\mathbf{d}_{\boldsymbol{x}_k}; \mathbf{d}_{\mathbf{W}_k}]$ is a linear combination of the directions of initial momentum and steepest gradient descent, where $\mathbf{d}_{\boldsymbol{x}_k}$ and $\mathbf{d}_{\mathbf{W}_k}$ are the descent directions of $\boldsymbol{x}_k$ and $\mathbf{W}_k$, respectively. The descent direction of $\boldsymbol{x}_k$ is

$$\begin{aligned} \mathbf{d}_{\boldsymbol{x}_k} &= \mathrm{grad}_{\boldsymbol{x}_k} C(\boldsymbol{x}_k, \mathbf{W}_k) + \beta_k \mathrm{Trans}_{\boldsymbol{x}_k \leftarrow \boldsymbol{x}_{k-1}}\left(\mathbf{d}_{\boldsymbol{x}_{k-1}}\right) \\ &= \left(\begin{array}{l} \mathrm{grad}_{\boldsymbol{x}_k} C(\boldsymbol{x}_k, \mathbf{W}_k) + \\ \beta_k \left[\mathbf{d}_{\boldsymbol{x}_{k-1}} - \Re\left\{\mathbf{d}_{\boldsymbol{x}_{k-1}}^* \odot \boldsymbol{x}_k\right\} \odot \boldsymbol{x}_k\right] \end{array} \right), \end{aligned} \tag{19}$$

where $\mathrm{Trans}_{\boldsymbol{x}_k \leftarrow \boldsymbol{x}_{k-1}}\left(\mathbf{d}_{\boldsymbol{x}_{k-1}}\right)$ is the transportation operation to address the calculation linkage between tangent vectors situated in distinct tangent spaces. This operation relocates the tangent vector $\mathbf{d}_{\boldsymbol{x}_{k-1}}$ from the point $\boldsymbol{x}_{k-1} \in \mathcal{M}$ to an alternate point $\boldsymbol{x}_k \in \mathcal{M}$. β_k is the conjugate parameter, which

is defined as

$$
\begin{aligned}
&\beta_k \\
&= \frac{\Re\left\{ \begin{array}{l} \mathrm{grad}_{\boldsymbol{x}_k}^H \left(\mathrm{grad}_{\boldsymbol{x}_k} - \mathrm{Trans}_{\boldsymbol{x}_k \leftarrow \boldsymbol{x}_{k-1}} \mathrm{grad}_{\boldsymbol{x}_{k-1}}\right) + \\ Tr\left[\mathrm{grad}_{\mathbf{W}_k}^H \left(\begin{array}{l} \mathrm{grad}_{\mathbf{W}_k} - \\ \mathrm{Trans}_{\mathbf{W}_k \leftarrow \mathbf{W}_{k-1}} \mathrm{grad}_{\mathbf{W}_{k-1}} \end{array} \right)\right] \end{array} \right\}}{\Re\left\{ \begin{array}{l} \left[\begin{array}{l} \left(\mathrm{grad}_{\boldsymbol{x}_k} - \mathrm{Trans}_{\boldsymbol{x}_k \leftarrow \boldsymbol{x}_{k-1}} \mathrm{grad}_{\boldsymbol{x}_{k-1}}\right)^H \\ \mathrm{Trans}_{\boldsymbol{x}_k \leftarrow \boldsymbol{x}_{k-1}} \mathbf{d}_{\boldsymbol{x}_{k-1}} \end{array} \right] + \\ Tr\left[\begin{array}{l} \left(\mathrm{grad}_{\mathbf{W}_k} - \mathrm{Trans}_{\mathbf{W}_k \leftarrow \mathbf{W}_{k-1}} \mathrm{grad}_{\mathbf{W}_{k-1}}\right)^H \\ \mathrm{Trans}_{\mathbf{W}_k \leftarrow \mathbf{W}_{k-1}} \mathbf{d}_{\mathbf{W}_{k-1}} \end{array} \right] \end{array} \right\}} \\
&= \frac{\Re\left\{ \begin{array}{l} \mathrm{grad}_{\boldsymbol{x}_k}^H \left[\begin{array}{l} \mathrm{grad}_{\boldsymbol{x}_k} - \\ \left[\begin{array}{l} \mathrm{grad}_{\boldsymbol{x}_{k-1}} - \\ \Re\left\{\mathrm{grad}_{\boldsymbol{x}_{k-1}} \odot \boldsymbol{x}_k^*\right\} \odot \boldsymbol{x}_k \end{array} \right] \end{array} \right] + \\ Tr\left[\mathrm{grad}_{\mathbf{W}_k}^H \left[\begin{array}{l} \mathrm{grad}_{\mathbf{W}_k} - \\ \left[\begin{array}{l} \mathrm{grad}_{\mathbf{W}_{k-1}} - \mathbf{W}_k \\ \Re\left\{Tr(\mathrm{grad}_{\mathbf{W}_{k-1}}^T \mathbf{W}_k^*)\right\} \end{array} \right] \end{array} \right]\right] \end{array} \right\}}{\Re\left\{ \begin{array}{l} \left(\begin{array}{l} \left[\mathrm{grad}_{\boldsymbol{x}_k} - \left(\begin{array}{l} \mathrm{grad}_{\boldsymbol{x}_{k-1}} - \\ \Re\left\{\mathrm{grad}_{\boldsymbol{x}_{k-1}} \odot \boldsymbol{x}_k^*\right\} \odot \boldsymbol{x}_k \end{array} \right)\right]^H \\ \left[\mathbf{d}_{\boldsymbol{x}_{k-1}} - \Re\left\{\mathbf{d}_{\boldsymbol{x}_{k-1}} \odot \boldsymbol{x}_k^*\right\} \odot \boldsymbol{x}_k\right] \end{array} \right) + \\ Tr\left(\begin{array}{l} \left[\mathrm{grad}_{\mathbf{W}_k} - \left(\begin{array}{l} \mathrm{grad}_{\mathbf{W}_{k-1}} - \mathbf{W}_k \\ \Re\left\{Tr(\mathrm{grad}_{\mathbf{W}_{k-1}}^T \mathbf{W}_k^*)\right\} \end{array} \right)\right]^H \\ \left[\mathbf{d}_{\mathbf{W}_{k-1}} - \mathbf{W}_k \Re\left\{Tr(\mathbf{d}_{\mathbf{W}_{k-1}}^T \mathbf{W}_k^*)\right\}\right] \end{array} \right) \end{array} \right\}},
\end{aligned}
\tag{20}
$$

where $\mathrm{grad}_{\boldsymbol{x}_k} = \mathrm{grad}_{\boldsymbol{x}_k} C\left(\boldsymbol{x}_k, \mathbf{W}_k\right)$, $\mathrm{grad}_{\mathbf{W}_k} = \mathrm{grad}_{\mathbf{W}_k} C\left(\boldsymbol{x}_k, \mathbf{W}_k\right)$.

Proof: Please refer to the Appendix B.

Similar to (19), the descent direction of $\mathbf{W}_k$ is

$$
\begin{aligned}
&\mathbf{d}_{\mathbf{W}_k} \\
&= \mathrm{grad}_{\mathbf{W}} C(\boldsymbol{x}_k, \mathbf{W}_k) + \beta_k \mathrm{Trans}_{\mathbf{W}_k \leftarrow \mathbf{W}_{k-1}} \left(\mathbf{d}_{\mathbf{W}_{k-1}}\right) \\
&= \left(\begin{array}{l} \mathrm{grad}_{\mathbf{W}} C(\boldsymbol{x}_k, \mathbf{W}_k) + \\ \beta_k \left[\mathbf{d}_{\mathbf{W}_{k-1}} - \mathbf{W}_k \Re\left\{Tr(\mathbf{d}_{\mathbf{W}_{k-1}}^T \mathbf{W}_k^*)\right\}\right] \end{array} \right).
\end{aligned}
\tag{21}
$$

Combining (19) and (21), the descent direction is

$$
\begin{aligned}
&\mathbf{d}_{(\boldsymbol{x}_k, \mathbf{W}k)} \\
&= [\mathbf{d}_{\boldsymbol{x}_k}; \mathbf{d}_{\mathbf{W}_k}] \\
&= \left[\begin{array}{l} \left(\begin{array}{l} \mathrm{grad}_{\boldsymbol{x}_k} C(\boldsymbol{x}_k, \mathbf{W}_k) + \\ \beta_k \left[\mathbf{d}_{\boldsymbol{x}_{k-1}} - \Re\left\{\mathbf{d}_{\boldsymbol{x}_{k-1}}^* \odot \boldsymbol{x}_k\right\} \odot \boldsymbol{x}_k\right] \end{array} \right); \\ \left(\begin{array}{l} \mathrm{grad}_{\mathbf{W}_k} C(\boldsymbol{x}_k, \mathbf{W}_k) + \\ \beta_k \left[\mathbf{d}_{\mathbf{W}_{k-1}} - \mathbf{W}_k \Re\left\{Tr(\mathbf{d}_{\mathbf{W}_{k-1}}^T \mathbf{W}_k^*)\right\}\right] \end{array} \right) \end{array} \right],
\end{aligned}
\tag{22}
$$

where $\mathbf{d}_{(\boldsymbol{x}_k, \mathbf{W}k)} = (\mathrm{grad}_{\boldsymbol{x}_k} C(\boldsymbol{x}_k, \mathbf{W}_k), \mathrm{grad}_{\mathbf{W}_k} C(\boldsymbol{x}_k, \mathbf{W}_k))$ if k is 1.

The update on the unified tangent space $\mathrm{T}_{(\boldsymbol{x}, \mathbf{W})}\mathcal{M}$ is

$$
\begin{aligned}
&\left(\hat{\boldsymbol{x}}_{k+1}, \hat{\mathbf{W}}_{k+1}\right) \\
&= (\boldsymbol{x}_k, \mathbf{W}_k) + \alpha_k \mathbf{d}_{(\boldsymbol{x}_k, \mathbf{W}k)} \\
&= [\boldsymbol{x}_k + \alpha_k \mathbf{d}_{\boldsymbol{x}_k}, \mathbf{W}_k + \alpha_k \mathbf{d}_{\mathbf{W}_k}] \\
&= \left[\begin{array}{l} \boldsymbol{x}_k + \alpha_k \left[\begin{array}{l} \mathrm{grad}_{\boldsymbol{x}_k} C(\boldsymbol{x}_k, \mathbf{W}_k) + \\ \beta_k \left[\begin{array}{l} \mathbf{d}_{\boldsymbol{x}_{k-1}} - \\ \Re\left\{\mathbf{d}_{\boldsymbol{x}_{k-1}}^* \odot \boldsymbol{x}_k\right\} \odot \boldsymbol{x}_k \end{array} \right] \end{array} \right]; \\ \mathbf{W}_k + \alpha_k \left[\begin{array}{l} \mathrm{grad}_{\mathbf{W}_k} C(\boldsymbol{x}_k, \mathbf{W}_k) + \\ \beta_k \left[\begin{array}{l} \mathbf{d}_{\mathbf{W}_{k-1}} - \\ \mathbf{W}_k \Re\left\{Tr(\mathbf{d}_{\mathbf{W}_{k-1}}^T \mathbf{W}_k^*)\right\} \end{array} \right] \end{array} \right] \end{array} \right].
\end{aligned}
\tag{23}
$$

where α_k is the step size adaptively updated by the Armijo linear search algorithm, as given in the next step. Update on the tangent space $\mathrm{T}_{(\boldsymbol{x}, \mathbf{W})}\mathcal{M}$ may not guarantee that the resulting point remains on the unified manifold space $\mathcal{M}$. Retraction is the process of mapping a point from $\mathrm{T}_{(\boldsymbol{x}, \mathbf{W})}\mathcal{M}$ back onto $\mathcal{M}$ to ensure feasibility. Hence, the next feasible solution is obtained by

$$
\begin{aligned}
&\left(\boldsymbol{x}_{k+1}, \mathbf{W}_{k+1}\right) \\
&= \mathrm{Ret}_{(\boldsymbol{x}, \mathbf{W})} \left(\hat{\boldsymbol{x}}_{k+1}, \hat{\mathbf{W}}_{k+1}\right) \\
&= \left[\mathrm{Ret}_{\boldsymbol{x}} \left(\hat{\boldsymbol{x}}_{k+1}\right), \mathrm{Ret}_{\mathbf{W}} \left(\hat{\mathbf{W}}_{k+1}\right)\right] \\
&= \left[\hat{\boldsymbol{x}}_{k+1} \oslash \left|\hat{\boldsymbol{x}}_{k+1}\right|, \hat{\mathbf{W}}_{k+1} / \left\|\hat{\mathbf{W}}_{k+1}\right\|_F^2\right],
\end{aligned}
\tag{24}
$$

where $\oslash$ denotes the element-wise deviation, and $|\cdot|$ is the element-wise absolute value.

4) Adaptive Step Size Update: Utilizing the Armijo linear search strategy, we dynamically adjust the step size during updates [37], [38]. This adaptive approach aligns the search step size with momentum variations, preserving the property of non-increasing objective function values and enhancing algorithmic convergence speed. The linear search strategy is

$$
-C(\boldsymbol{x}_{k+1}, \mathbf{W}_{k+1}) \leq -C(\boldsymbol{x}_k, \mathbf{W}_k) + \tau_1 \tau_2^n \alpha_k \hat{\mathbf{d}}_{C_k}, \tag{25}
$$

where $\alpha_{k+1} = \tau_2^n \alpha_k$, α_k is the initial step size for the next iteration, τ_2 is a coefficient greater than 0 and less than 1 ($\tau_2 = 0.5$ in our simulation), n is the number of linear searches, and $\hat{\mathbf{d}}_{C_k}$ is given by

$$
\hat{\mathbf{d}}_{C_k} = \mathrm{grad}^H[-C(\boldsymbol{x}_k, \mathbf{W}_k)] \mathbf{d}_{(\boldsymbol{x}_k, \mathbf{W}_k)}. \tag{26}
$$

To expedite convergence even further, the upcoming iteration's step size α can be appropriately adjusted based on variations in n. If $n = 1$ satisfies (25), it means that only one linear search has been performed, which results in satisfying the objective function decrease. This suggests the current step size is small, it should be increased to $\alpha = \xi_1 \alpha_{k+1}$ in the next iteration, where $\xi_1 > 1$. If $n = 2$ satisfies (25), it indicates a reasonably chosen initial step size, necessitating that the subsequent iteration maintains the step size at $\alpha = \alpha_{k+1}$. If $n \geq 3$ satisfies (25), it indicates that more than three linear searches have been performed, resulting in a small step size at the current point, it should be increased to $\alpha = \xi_2 \alpha_{k+1}$ in the next iteration, where $\xi_2 > 1$. In summary, the step size α

update rule for the next iteration is summarized as

$$\alpha=\begin{cases}\xi_1\alpha_{k+1}, n=1,\\ \alpha_{k+1}, n=2,\\ \xi_2\alpha_{k+1}, n\geq 3.\end{cases} \tag{27}$$

Taking into account the preceding discussions, the Armijo line-search algorithm is compactly presented in **Algorithm 1**.

Algorithm 1 : The Armijo line-search algorithm

Input: $\boldsymbol{x}_k, \mathbf{W}_k, \mathbf{d}_{(\boldsymbol{x}_k,\mathbf{W}_k)}, \tau_1\in(0,1), \tau_2\in(0,1), \alpha_k$
1: Set $n=1$;
2: **while** not satisfy (25) and $n\leq 10$, **do:**
3: Uptade $\alpha_k=\tau_2\alpha_k$;
4: Update $(\boldsymbol{x}_{k+1}, \mathbf{W}_{\mathrm{k+1}})$ with (23) and (24);
5: $n=n+1$;
6: **end while**;
7: **switch** n
8: **case** 1: Let $\alpha_{k+1}=2\alpha_k$;
9: **case** 2: Let $\alpha_{k+1}=\alpha_k$;
10: **else** : Let $\alpha_{k+1}=2\alpha_k$;
11: **end switch**;
Output: $\boldsymbol{x}_{k+1}, \mathbf{W}_{\mathrm{k+1}}, \alpha_{k+1}$

5) The overall UMO Method: The UMO method to tackle (4) is compactly presented in **Algorithm 2**. The algorithm mainly includes: 1) update the parallel Euclidean gradient; 2) update the parallel manifold gradient; 3) update the feasible solution; 4) adaptively update the step size with Armijo linear search.

Algorithm 2 : The UMO Method to solve (4)

Input: $\boldsymbol{x}_0, \mathbf{W}_0, \varepsilon$
1: Initialize $k=1, \boldsymbol{x}_k=\boldsymbol{x}_0, \mathbf{W}_k=\mathbf{W}_0$, $\alpha_k=1$, update $\mathbf{d}_{(\boldsymbol{x}_k,\mathbf{W}_k)}$ with (22);
2: **While** $\|\mathrm{grad}\ C(x_k,\mathbf{W}_k)\|\geq\varepsilon$ **do:**
3: Update $\mathbf{K}(\boldsymbol{x}_k,\mathbf{W}_k)$ with (13);
4: Update Euclidean gradient $\nabla C(\boldsymbol{x}_k,\mathbf{W}_k)$ with (15);
5: Update manifold gradient $\mathrm{grad}\ C(\boldsymbol{x}_k,\mathbf{W}_k)$ with (18);
6: Update β_k with (20) and set $\beta_k=\max(0,\beta_k)$;
7: Update descent direction $\mathbf{d}_{(\boldsymbol{x}_k,\mathbf{W}_k)}$ with (22);
8: Update $(\boldsymbol{x}_{k+1}, \mathbf{W}_{\mathrm{k+1}})$ with (23) and (24);
9: Update $\boldsymbol{x}_{k+1}, \mathbf{W}_{\mathrm{k+1}}, \alpha_{k+1}$ with **Algorithm 1**;
10: $k=k+1$;
11: **end While**;
Output: $\boldsymbol{x}_{k+1}, \mathbf{W}_{\mathrm{k+1}}$

C. Convergence analysis

As shown in [39], [40], we assume a sufficient descent direction is achieved by (25), given by Assumption 1 below.

Assumption 1 (Sufficient decrease): There exist $c>0, c'>0$ such that, for all $k>0$,

$$\begin{aligned}&-C(\mathbf{W}_k,x_k)-[-C(\mathbf{W}_{k+1},x_{k+1})]\\ &\geq\min\left(c\left\|\mathrm{grad}C(\mathbf{W}_k,x_k)\right\|,c'\right)\left\|\mathrm{grad}C(\mathbf{W}_k,x_k)\right\|.\end{aligned} \tag{28}$$

According to Assumption 1, we can conclude that $-C(\mathbf{W}_k,\boldsymbol{x}_k)\geq -C(\mathbf{W}_{k+1},\boldsymbol{x}_{k+1})$, indicating that our algorithm can reach convergence.

D. Complexity analysis

Based on **Algorithm 2**, the primary factor influencing the computational complexity of each iteration arises from the computation of $\mathbf{G}_{\text{eff}}$ in (3), $\mathbf{K}(\boldsymbol{x},\mathbf{W})$ in (13), the gradient (i.e., $\nabla_{\boldsymbol{x}}C(\boldsymbol{x},\mathbf{W})$ in (12) and $\nabla_{\mathbf{W}}C(\boldsymbol{x},\mathbf{W})$ in (14)). Here, the complexity for calculating $\mathbf{G}_{\text{eff}}$ is $(N_bN_r+N_bN_rN_t)$; the complexity for calculating $\mathbf{K}(\boldsymbol{x},\mathbf{W})$ is $(N_b^3+N_bN_sN_t+N_b^2N_s)$; the complexity for calculating $\nabla_{\boldsymbol{x}}C(\boldsymbol{x},\mathbf{W})$ is $(N_tN_r^2+N_r(N_b^2+N_bN_t+2N_tN_s))$, and the complexity for calculating $\nabla_{\mathbf{W}}C(\boldsymbol{x},\mathbf{W})$ is $(N_tN_b^2+N_bN_t^2+N_sN_t^2)$. Since N_r is usually much larger than N_t, N_b, and N_s, the complexity order is approximated as $\mathcal{O}\left(N_tN_r^2\right)$ per iteration.

The complexity order of ADMM [17] is approximated as $\mathcal{O}\left(N_r^3\right)$ per iteration. The complexity order of GP [23], [24] is approximated as $\mathcal{O}\left(N_tN_r^2\right)$ per iteration. The complexity order of DSM and GA is approximated as $\mathcal{O}\left(N_r^2\right)$ per iteration [20]. The complexity order of MM [18], [19] is approximated as $\mathcal{O}\left(N_r\right)$ per iteration.

A brief comparison of the complexity orders for all methods is presented in Table I. As observed, the complexity order of our method is equal to that of GP in [23], [24], DSM, and GA in [20], lower than that of ADMM [17], and higher than MM [18], [19].

TABLE I: A comparison of the complexity orders

Method	Complexity order
Proposed method	$\mathcal{O}\left(N_tN_r^2\right)$
ADMM in [17]	$\mathcal{O}\left(N_r^3\right)$
GP in [23], [24]	$\mathcal{O}\left(N_tN_r^2\right)$
DSM and GA in [20]	$\mathcal{O}\left(N_r^2\right)$
MM in [18], [19]	$\mathcal{O}\left(N_r\right)$

IV. Numerical results

In this section, we compare the proposed method with ADMM [17], DSM [20], GA [20], GP [23], [24], and MM [18], [19] algorithms. We consider both scenarios of perfect channel and imperfect channel. To ensure fair comparisons, all methods are initialized with the same values. The RIS phase shifters and beamforming are initialized as random vectors with unit-magnitude complex entries.

A. Spectral efficiency comparison with perfect channel

For the sake of convenient comparisons, the simulation setup follows the same configuration as described in [17], [20]: $\mathbf{G}_d$, $\mathbf{G}_r$, and $\mathbf{M}$ are represented as Rician fading channels, encompassing both non-line-of-sight (NLoS) and line-of-sight (LoS) components. They are given by

$$\mathbf{G}=\sqrt{L(d)}\left(\sqrt{\frac{\kappa}{1+\kappa}}\boldsymbol{a}_r(\phi_r)\boldsymbol{a}_t(\phi_t)^H+\sqrt{\frac{1}{1+\kappa}}\mathbf{G}^{\mathrm{NLoS}}\right), \tag{29}$$

where $L(d)=C_0(d/D_0)^{-a}$ is a path-loss factor. Here, a denotes the path loss exponent, D_0 is the reference distance, and

C_0 is the path loss at D_0. Additionally, κ denotes the Rician factor, while ϕ_r and $\phi_t \in [0, 2\pi)$ respectively denote the azimuth angles of arrival and departure for the LoS component. $\boldsymbol{a}_t(\phi_t)$ and $\boldsymbol{a}_r(\phi_r)$ stand for the transmit antenna vector and receive antenna vector, respectively. To maintain simplicity, we consider a uniform linear array of N antennas, denoted as $\boldsymbol{a}(\phi) = \left(1/\sqrt{N}\right)\left[1, e^{jkd_a \sin\phi}, \ldots, e^{jkd_a(N-1)\sin\phi}\right]^T$, where $k = 2\pi/\lambda$, and λ represents the wavelength while d_a stands for the antenna spacing. $\mathbf{G}^{\mathrm{NLoS}}$ is generated randomly, with its elements satisfying a Gaussian distribution with a unit variance and zero mean.

In the subsequent simulations, the number of users is $N_b = 12$, and the number of antennas is $N_t = 16$. The Rician factor for channels is set to $\kappa = 10$ dB. $N_s = 12, C_0 = -30$ dB, $d = 30$ m, $D_0 = 1$ m, $d_a = \lambda/2$, and $a = 2$. The experimental results are averaged over 50 Monte Carlo simulations.

Figure 2 shows a comparison of spectral efficiency versus power for the proposed method, ADMM [17], DSM [20], GA [20], GP [23], [24], and MM [18], [19] methods. The superiority of spectral efficiency gain with the proposed method becomes more apparent as the power level increases for both $N_r = 16$ and $N_r = 32$. Specifically, when $N_r = 16$ and power is 20 dB, the proposed method achieves a 6.18 bits/Hz/s higher gain than ADMM [17], 2.78 bits/Hz/s higher gain than the methods in [18]–[20], and 0.76 bits/Hz/s higher gain than GP [23], [24]. Similarly, when $N_r = 32$ and power is 20 dB, the proposed method achieves a 10.12 bits/Hz/s higher gain than ADMM [17], 3.94 bits/Hz/s higher gain than the methods in [18]–[20], and 1.65 bits/Hz/s higher gain than GP [23], [24].

Figure 3 shows a comparison of spectral efficiency versus the number of RIS elements for the proposed method, ADMM [17], DSM [20], GA [20], GP [23], [24], and MM [18], [19] methods. Specifically, when $N_r = 200$, the proposed method achieves a 25 bits/Hz/s higher gain than ADMM [17], approximately 5.58 bits/Hz/s higher gain than the methods in [18]–[20], and 4.88 bits/Hz/s higher gain than GP method in [23], [24]. Furthermore, compared to the GP method, the proposed

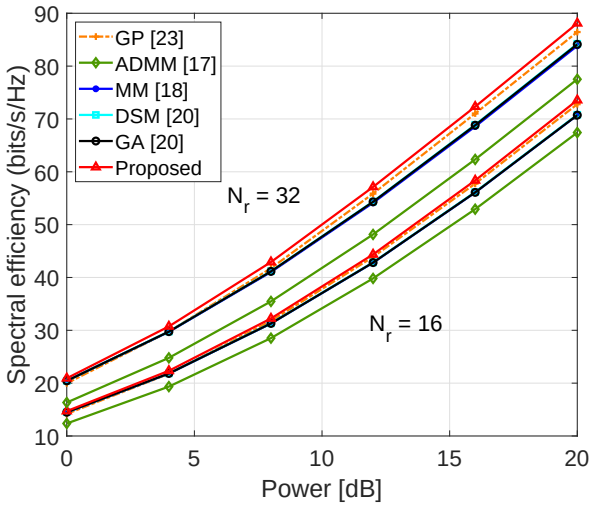


Fig. 2: Spectral efficiency comparison versus power.

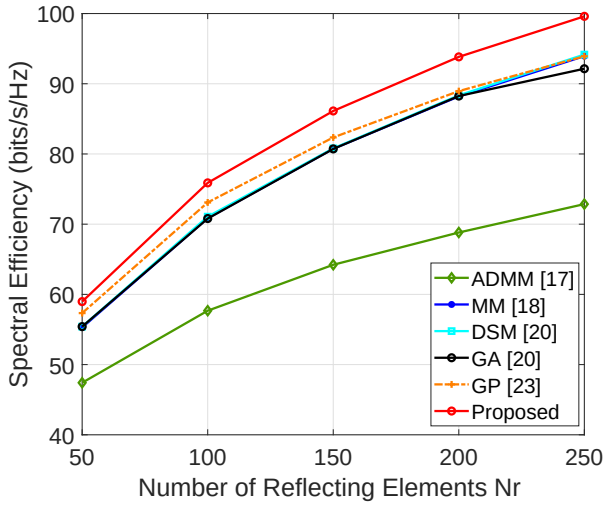


Fig. 3: Spectral efficiency comparison versus the number of RIS elements with 10 dB power.

method demonstrates superior spectral efficiency gains as the number of RIS elements increases. This is because, with larger RIS sizes, the optimization problem involves more variables, becoming more non-convex with numerous local minima. The GP method performs a direct projection from Euclidean space to a complex non-convex space without adapting to the optimization landscape's local features. In contrast, our proposed UMO method adapts to these local features. This adaptability aligns its update steps with the structure of the unified manifold space (UMS), allowing UMO to converge more effectively towards better solutions.

Figs. 2 and 3 demonstrate that the proposed method outperforms existing methods. In contrast to relaxing the spectral efficiency maximization into the SPGM problem with a large relaxation gap [17]–[20], we directly solve the spectral efficiency maximization problem. Different from GP, which performs a direct projection from Euclidean space to a complex non-convex space without adapting to the local features of the complex non-convex optimization landscape [23], [24], the proposed UMO method is adaptive to the local features of the complex non-convex optimization landscape.

B. Spectral efficiency comparison with imperfect channel

The imperfect channel is the same as the one in [27]: Referring to (31), the requirement is to estimate the small-scale fading variables $\mathbf{G}_d$, $\mathbf{G}_r$, and $\mathbf{M}$ per frame. Let g represent an element within these variables, with $\hat{g}$ being the associated estimated value. The estimation error $(g - \hat{g})$ is assumed to satisfy complex Gaussian distribution with a zero-mean, and all elements share a common normalized mean square error (MSE)

$$\rho = \frac{E\left[|g - \hat{g}|^2\right]}{E\left[|\hat{g}|^2\right]}. \tag{30}$$

In the following simulations, the MSE factor is $\rho = 0.5$. The numbers of antennas are set as $N_t = 16$ and $N_b = 12$.

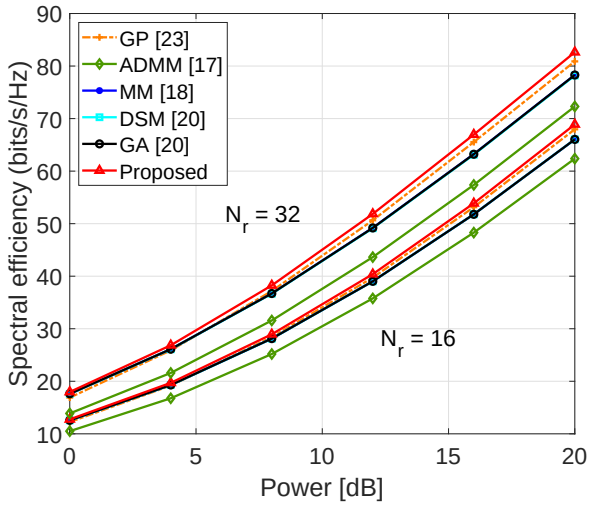


Fig. 4: Spectral efficiency comparison versus power in the case of imperfect channel.

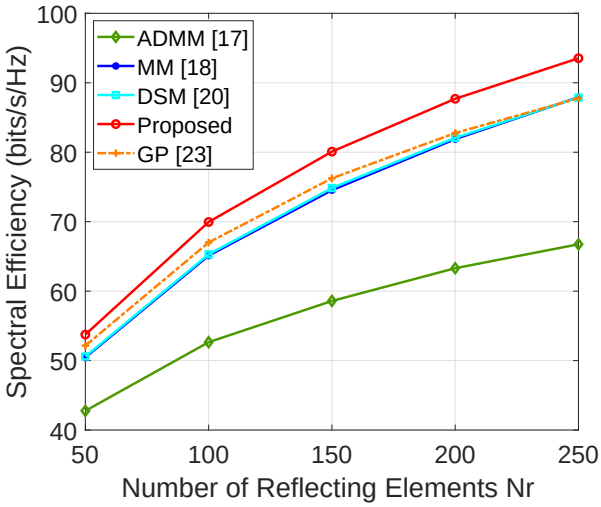


Fig. 5: Spectral efficiency comparison versus the number of reflecting elements with imperfect channel.

The Rician factors for channels are set as $\kappa = 10$ dB. $N_s = 12, C_0 = -30$ dB, $d = 30$ m, $D_0 = 1$ m, $d_a = \lambda/2$, and $a = 2$. The experimental results take the average value over 50 times of Monte Carlo simulations.

Figure 4 shows the comparison of spectral efficiency versus power in the case of imperfect channels for the proposed method, ADMM [17], DSM [20], GA [20], GP [23], [24], and MM [18], [19] methods. Notably, the superior spectral efficiency gain of the proposed method becomes more apparent as the power level increases for both $N_r = 16$ and $N_r = 32$. Specifically, at a power level of 20 dB with $N_r = 16$, the proposed method achieves a 6.32 bits/Hz/s higher gain than ADMM [17], about 2.77 bits/Hz/s higher than methods in [18]–[20], and 0.93 bits/Hz/s higher gain than GP [23], [24]. Similarly, for $N_r = 32$ at a power level of 20 dB, the proposed method outperforms ADMM [17] by 10.21 bits/Hz/s, other methods in [18]–[20] by approximately 4 bits/Hz/s, and GP method in [23], [24] by 1.87 bits/Hz/s.

Figure 5 presents a spectral efficiency comparison versus reflecting elements N_r with imperfect channels for the proposed method, ADMM [17], DSM [20], GA [20], GP [23], [24], and MM [18], [19] methods. As N_r increases, the effectiveness of the proposed method in enhancing spectral efficiency becomes increasingly evident. Specifically, when $N_r = 200$, the proposed method achieves a 24.39 bits/Hz/s higher gain than ADMM [17], approximately 5.62 bits/Hz/s higher than methods in [18]–[20], and 4.93 bits/Hz/s higher gain than GP [23], [24].

Figs. 4 and 5 demonstrate that the proposed method can achieve better spectral efficiency gains than existing methods in the case of imperfect channels.

C. Comparison of computational time

Figure 6 shows the comparison of execution time versus reflecting elements N_r with (a) perfect channels and (b) imperfect channels. The experimental results represent the average value over 50 Monte Carlo simulations. As illustrated, the execution time required by our method is much less than that of DSM [20], GA [20] and GP [23], [24] methods, approximately equal to the ADMM method [17], and more than the MM method [18], [19]. Typically, for the scenario of perfect channels with $N_r = 250$, the proposed method exhibits an execution time of 0.496 seconds. This is approximately 2 orders of magnitude faster than GA [20], 1 order of magnitude faster than DSM [20], 0.5 order of magnitude faster than GP [23], roughly equivalent to ADMM [17], and approximately 1 order of magnitude slower than MM [18], [19].

In summary, the results from Figures 2 to 6 indicate that the proposed method outperforms existing methods [17], [20], [23] in solving (4). This superiority is evident not only in terms of improved spectral efficiency performance but also in favorable computational costs. While the MM method [18], [19] has lower complexity than the proposed method, the proposed method achieves a more significant gain in spectral efficiency compared to the MM method.

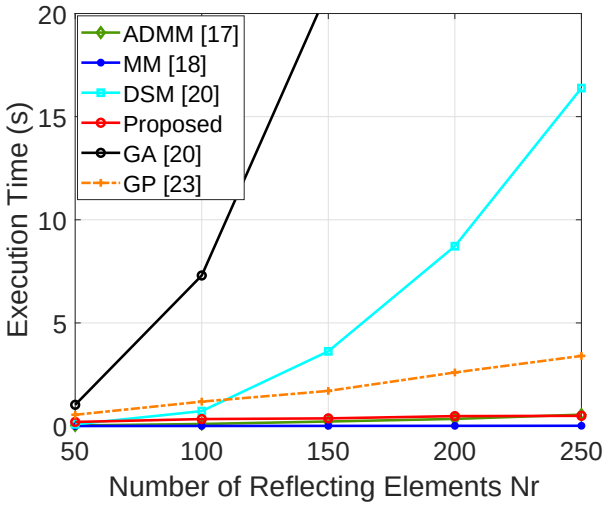
D. Convergence behavior

The simulation configurations for both the perfect channel and imperfect channel are set to be the same as in subsections A and B, respectively.

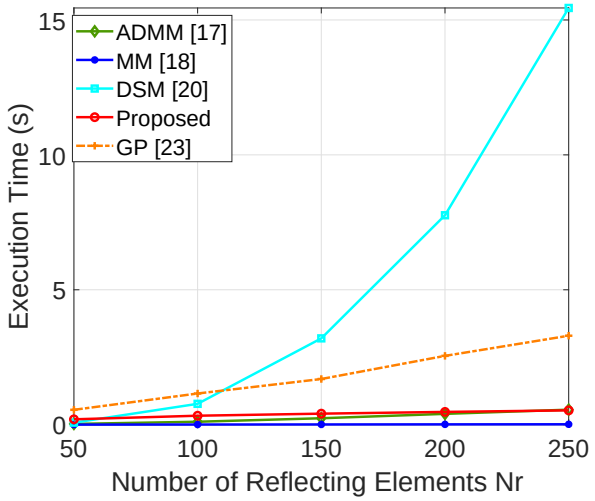
Figure 7 shows the cost value versus the number of iterations for (a) N_r = 16 and (b) N_r = 32. It is evident that the proposed method exhibits rapid convergence for both perfect and imperfect channels. Specifically, with $N_r = 16$, the proposed method achieves convergence within 100 iterations for the perfect channel and within 200 iterations for the imperfect channel. Additionally, when MSE $\rho = 0.1$, the spectral efficiency difference between the perfect and imperfect channels diminishes as the number of RIS elements increases.

V. Conclusion

Reconfigurable intelligent surfaces (RIS) aided MIMO beamforming design for spectral efficiency maximization is

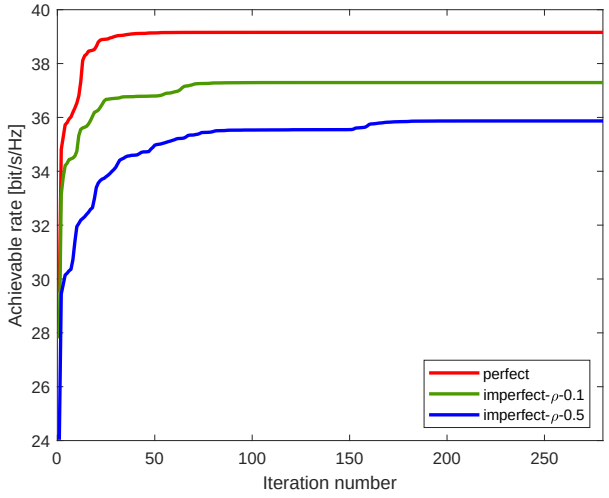


(a) Perfect channel

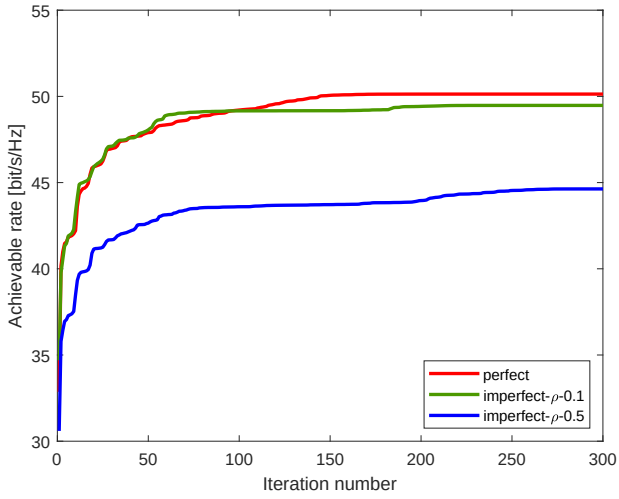


(b) Imperfect channel

Fig. 6: Execution time comparison versus the number of reflecting elements (a) Perfect channel; (b) Imperfect channel.



(a) N_r = 16



(b) N_r = 32

Fig. 7: The achievable rate versus iterations with (a)N_r = 16, (b)N_r = 32.

studied in this work. A specific challenge in this context involves codesigning the nonconvex phase shifts for RIS and a precoding matrix with a complex sphere constraint, where these coupled variables are used to formulate the nonconvex objective of maximizing spectral efficiency. Most current approaches addressed the problem with relaxing the spectral efficiency maximization into sum-path-gain-maximization (SPGM) problem with large relaxation gap. We proposed an efficient Unified Manifold Optimization (UMO) framework to directly solve the problem. This is achieved through utilizing the inherent constant modulus characteristic of the RIS and the complex sphere characteristic of the precoding matrix constraint. Specifically, we constructed a unified manifold space (UMS) that can simultaneously satisfy the RIS and the precoding matrix constraints, enabling the problem to rephrase as an unconstrained Riemannian problem over the UMS. Based on the UMS, we derived a parallel conjugate gradient algorithm to simultaneously optimize the precoding matrix and phase shifts of RIS. Simulation outcomes indicate that the proposed method excels when compared to current approaches in achieving spectral efficiency enhancement. Furthermore, our algorithm has lower computational cost than several existing techniques.

APPENDIX

A. Appendix A: proof of (15)

To derive the gradient of $C(\boldsymbol{x}, \mathbf{W})$ (8) w.r.t. $\boldsymbol{x}^*$ and $\mathbf{W}^*$, we need to derive the total differential $dC(\boldsymbol{x}, \mathbf{W})$ at $\boldsymbol{x}$, which is given by

$$\begin{aligned} & dC(\boldsymbol{x}, \mathbf{W}) \\ &= \mathrm{Tr}\left\{\mathbf{K}(\boldsymbol{x}, \mathbf{W}) d\left(\mathbf{G}_{\mathrm{eff}} \mathbf{W}\mathbf{W}^H \mathbf{G}_{\mathrm{eff}}^H\right)\right\} \\ &= \mathrm{Tr}\left\{\mathbf{K}(\boldsymbol{x}, \mathbf{W}) \begin{pmatrix} d(\mathbf{G}_{\mathrm{eff}}) \mathbf{W}\mathbf{W}^H \mathbf{G}_{\mathrm{eff}}^H \\ +\mathbf{G}_{\mathrm{eff}} \mathbf{W}\mathbf{W}^H d(\mathbf{G}_{\mathrm{eff}}^H) \end{pmatrix}\right\}, \end{aligned} \tag{31}$$

where $\mathbf{K}(\boldsymbol{x},\mathbf{W}) = (\mathbf{I}+\mathbf{G}_{\text{eff}}\mathbf{W}\mathbf{W}^H\mathbf{G}_{\text{eff}}^H)^{-1}$. Using the property $\text{Tr}\left(\mathbf{A}^T\mathbf{B}\right) = \text{vec}^T(\mathbf{A})\,\text{vec}(\mathbf{B})$, (31) is simplified as

$$\begin{aligned} &dC(\boldsymbol{x},\mathbf{W}) \\ &= \text{vec}^T\left((\mathbf{M}\mathbf{W}\mathbf{W}^H\mathbf{G}_{\text{eff}}^H\mathbf{K}(\boldsymbol{x},\mathbf{W})\mathbf{G}_r)^T\right)\text{vec}_d(d\boldsymbol{x}) \\ &+ \text{vec}^T\left((\mathbf{M}^*\mathbf{W}^*\mathbf{W}^T\mathbf{G}_{\text{eff}}^T\mathbf{K}(\boldsymbol{x},\mathbf{W})\mathbf{G}_r^*)^T\right)\text{vec}_d(d\boldsymbol{x}^*). \end{aligned} \tag{32}$$

Let $\mathbf{L}_d$ be the matrix used to place the diagonal elements of the square matrix $\mathbf{A}$ on $\text{vec}(\mathbf{A})$, i.e., $\text{vec}_d(\mathbf{A}) = \mathbf{L}_d\,\text{vec}(\mathbf{A})$. Then we can rewrite $dC(\boldsymbol{x},\mathbf{W})$ as

$$\begin{aligned} &dC(\boldsymbol{x},\mathbf{W}) \\ &= \text{vec}^T\left((\mathbf{M}\mathbf{W}\mathbf{W}^H\mathbf{G}_{\text{eff}}^H\mathbf{K}(\boldsymbol{x},\mathbf{W})\mathbf{G}_r)^T\right)\mathbf{L}_d\,\text{vec}(d\boldsymbol{x}) \\ &+ \text{vec}^T\left((\mathbf{M}^*\mathbf{W}^*\mathbf{W}^T\mathbf{G}_{\text{eff}}^T\mathbf{K}(\boldsymbol{x},\mathbf{W})\mathbf{G}_r^*)^T\right)\mathbf{L}_d\,\text{vec}(d\boldsymbol{x}^*). \end{aligned} \tag{33}$$

Based on (33), the gradient of $C(\boldsymbol{x},\mathbf{W})$ at $\boldsymbol{x}^*$ is

$$\begin{aligned} &\nabla_{\boldsymbol{x}^*}C(\boldsymbol{x},\mathbf{W}) \\ &= \text{vec}_d\left(\mathbf{G}_r^H\mathbf{K}(\boldsymbol{x},\mathbf{W})\mathbf{G}_{\text{eff}}\mathbf{W}\mathbf{W}^H\mathbf{M}^H\right) \\ &= \text{diag}\left(\mathbf{G}_r^H\mathbf{K}(\boldsymbol{x},\mathbf{W})\mathbf{G}_{\text{eff}}\mathbf{W}\mathbf{W}^H\mathbf{M}^H\right) \end{aligned}. \tag{34}$$

Similar to (31), the complex differential of $C(\boldsymbol{x},\mathbf{W})$ w.r.t. $\mathbf{W}$ and $\mathbf{W}^*$ is given by

$$\begin{aligned} &dC(\boldsymbol{x},\mathbf{W}) \\ &= \text{Tr}\left\{\mathbf{K}(\boldsymbol{x},\mathbf{W})d\left(\mathbf{G}_{\text{eff}}\mathbf{W}\mathbf{W}^H\mathbf{G}_{\text{eff}}^H\right)\right\} \\ &= \text{Tr}\left\{\mathbf{K}(\boldsymbol{x},\mathbf{W})\begin{pmatrix}\mathbf{G}_{\text{eff}}d(\mathbf{W})\mathbf{W}^H\mathbf{G}_{\text{eff}}^H + \\ \mathbf{G}_{\text{eff}}\mathbf{W}d(\mathbf{W}^H)\mathbf{G}_{\text{eff}}^H\end{pmatrix}\right\} \\ &= \begin{pmatrix}\text{vec}^T\left(\left(\mathbf{W}^H\mathbf{G}_{\text{eff}}^H\mathbf{K}(\boldsymbol{x},\mathbf{W})\mathbf{G}_{\text{eff}}\right)^T\right)\text{vec}(d\mathbf{W}) \\ +\text{vec}^T\left(\left(\mathbf{W}^T\mathbf{G}_{\text{eff}}^T\mathbf{K}(\boldsymbol{x},\mathbf{W})\mathbf{G}_{\text{eff}}^*\right)\right)\text{vec}(d\mathbf{W}^*)\end{pmatrix}. \end{aligned} \tag{35}$$

According to (35), the gradient of $C(\boldsymbol{x},\mathbf{W})$ at $\mathbf{W}^*$ is

$$\nabla_{\mathbf{W}^*}C(\boldsymbol{x},\mathbf{W}) = \mathbf{G}_{\text{eff}}^H\mathbf{K}(\boldsymbol{x},\mathbf{W})\mathbf{G}_{\text{eff}}\mathbf{W}. \tag{36}$$

Combining (34) and (36), we can obtain the parallel Euclidean gradient $\nabla C(\boldsymbol{x},\mathbf{W})$ in (14).

B. Appendix B: proof of (20)

For complex vector $\boldsymbol{x}_k$, the Hestenes and Stiefel (H-S) conjugate parameter is defined as [38], [41]

$$\beta_{\boldsymbol{x}_k} = \frac{\Re\left\{\text{grad}_{\boldsymbol{x}_k}^H\left(\text{grad}_{\boldsymbol{x}_k} - \text{Trans}_{\boldsymbol{x}_k\leftarrow\boldsymbol{x}_{k-1}}\text{grad}_{\boldsymbol{x}_{k-1}}\right)\right\}}{\Re\left\{\begin{bmatrix}\left(\text{grad}_{\boldsymbol{x}_k} - \text{Trans}_{\boldsymbol{x}_k\leftarrow\mathbf{x}_{k-1}}\text{grad}_{\boldsymbol{x}_{k-1}}\right)^H \\ \text{Trans}_{\boldsymbol{x}_k\leftarrow\boldsymbol{x}_{k-1}}\mathbf{d}_{\boldsymbol{x}_{k-1}}\end{bmatrix}\right\}}. \tag{37}$$

For complex matrix $\mathbf{W}_k$, H-S conjugate parameter is

$$\beta_{\mathbf{W}_k} = \frac{\Re\left\{Tr\left[\text{grad}_{\mathbf{W}_k}^H\begin{pmatrix}\text{grad}_{\mathbf{W}_k} - \\ \text{Trans}_{\mathbf{W}_k\leftarrow\mathbf{W}_{k-1}}\text{grad}_{\mathbf{W}_{k-1}}\end{pmatrix}\right]\right\}}{\Re\left\{Tr\left[\begin{matrix}\left(\text{grad}_{\mathbf{W}_k} - \text{Trans}_{\mathbf{W}_k\leftarrow\mathbf{W}_{k-1}}\text{grad}_{\mathbf{W}_{k-1}}\right)^H \\ \text{Trans}_{\mathbf{W}_k\leftarrow\mathbf{W}_{k-1}}\mathbf{d}_{\mathbf{W}_{k-1}}\end{matrix}\right]\right\}}. \tag{38}$$

Combining (37) and (38), H-S conjugate parameter for $(\boldsymbol{x}_k,\mathbf{W}_k)$ can be obtained by

$$\beta_k = \frac{\Re\left\{\begin{matrix}\text{grad}_{\boldsymbol{x}_k}^H\left(\text{grad}_{\boldsymbol{x}_k} - \text{Trans}_{\boldsymbol{x}_k\leftarrow\mathbf{x}_{k-1}}\text{grad}_{\boldsymbol{x}_{k-1}}\right) + \\ Tr\left[\text{grad}_{\mathbf{W}_k}^H\begin{pmatrix}\text{grad}_{\mathbf{W}_k} - \\ \text{Trans}_{\mathbf{W}_k\leftarrow\mathbf{W}_{k-1}}\text{grad}_{\mathbf{W}_{k-1}}\end{pmatrix}\right]\end{matrix}\right\}}{\Re\left\{\begin{matrix}\begin{bmatrix}\left(\text{grad}_{\boldsymbol{x}_k} - \text{Trans}_{\boldsymbol{x}_k\leftarrow\mathbf{x}_{k-1}}\text{grad}_{\boldsymbol{x}_{k-1}}\right)^H \\ \text{Trans}_{\boldsymbol{x}_k\leftarrow\boldsymbol{x}_{k-1}}\mathbf{d}_{\boldsymbol{x}_{k-1}}\end{bmatrix} + \\ Tr\begin{bmatrix}\left(\text{grad}_{\mathbf{W}_k} - \text{Trans}_{\mathbf{W}_k\leftarrow\mathbf{W}_{k-1}}\text{grad}_{\mathbf{W}_{k-1}}\right)^H \\ \text{Trans}_{\mathbf{W}_k\leftarrow\mathbf{W}_{k-1}}\mathbf{d}_{\mathbf{W}_{k-1}}\end{bmatrix}\end{matrix}\right\}}. \tag{39}$$

REFERENCES

[1] B. C. Pandey, S. K. Mohammed, P. Raviteja, Y. Hong, and E. Viterbo, "Low complexity precoding and detection in multi-user massive mimo otfs downlink," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 5, pp. 4389–4405, 2021.

[2] S. J. Maeng, Y. Yapıcı, s. Güvenç, A. Bhuyan, and H. Dai, "Precoder design for physical-layer security and authentication in massive mimo uav communications," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 3, pp. 2949–2964, 2022.

[3] R. Zhang, H. Zhang, Y. Li, Y. Ruan, O. A. Dobre, N. Song, T. Yang, and Y. Lu, "Energy efficient hybrid precoding for adaptive partially-connected mmwave massive mimo: A decomposition-based approach," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 12, pp. 15 967–15 980, 2023.

[4] Z. Wang, J. Zhang, H. Du, W. E. I. Sha, B. Ai, D. Niyato, and M. Debbah, "Extremely large-scale MIMO: Fundamentals, challenges, solutions, and future directions," *IEEE Wireless Communications*, pp. 1–9, 2023.

[5] P. Zhang, J. Zhang, H. Xiao, X. Zhang, D. W. K. Ng, and B. Ai, "Joint distributed precoding and beamforming for ris-aided cell-free massive mimo systems," *IEEE Transactions on Vehicular Technology*, pp. 1–6, 2023.

[6] M. Hua, Q. Wu, W. Chen, O. A. Dobre, and A. L. Swindlehurst, "Secure intelligent reflecting surface-aided integrated sensing and communication," *IEEE Transactions on Wireless Communications*, vol. 23, no. 1, pp. 575–591, 2024.

[7] Y. Zhang, J. Zhang, M. D. Renzo, H. Xiao, and B. Ai, "Performance analysis of ris-aided systems with practical phase shift and amplitude response," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 5, pp. 4501–4511, 2021.

[8] H. Xiao, X. Hu, T.-X. Zheng, and K.-K. Wong, "Star-ris assisted covert communications in noma systems," *IEEE Transactions on Vehicular Technology*, pp. 1–6, 2023.

[9] C. Huang, A. Zappone, G. C. Alexandropoulos, M. Debbah, and C. Yuen, "Reconfigurable intelligent surfaces for energy efficiency in wireless communication," *IEEE Transactions on Wireless Communications*, vol. 18, no. 8, pp. 4157–4170, 2019.

[10] M. Di Renzo, A. Zappone, M. Debbah, M.-S. Alouini, C. Yuen, J. de Rosny, and S. Tretyakov, "Smart radio environments empowered by reconfigurable intelligent surfaces: How it works, state of research, and the road ahead," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 11, pp. 2450–2525, 2020.

[11] K. Zhi, C. Pan, H. Ren, K. Wang, M. Elkashlan, M. D. Renzo, R. Schober, H. V. Poor, J. Wang, and L. Hanzo, "Two-timescale design for reconfigurable intelligent surface-aided massive MIMO systems with imperfect CSI," *IEEE Transactions on Information Theory*, vol. 69, no. 5, pp. 3001–3033, 2023.

[12] E. Shtaiwi, H. Zhang, A. Abdelhadi, A. Lee Swindlehurst, Z. Han, and H. Vincent Poor, "Sum-rate maximization for RIS-assisted integrated sensing and communication systems with manifold optimization," *IEEE Transactions on Communications*, pp. 1–1, 2023.

[13] S. Zhang and R. Zhang, "Capacity characterization for intelligent reflecting surface aided MIMO communication," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 8, pp. 1823–1838, 2020.

[14] K. Zhi, C. Pan, H. Ren, K. K. Chai, and M. Elkashlan, "Active RIS versus passive RIS: Which is superior with the same power budget?" *IEEE Communications Letters*, vol. 26, no. 5, pp. 1150–1154, 2022.

[15] R. P. Sankar, S. P. Chepuri, and Y. C. Eldar, "Beamforming in integrated sensing and communication systems with reconfigurable intelligent surfaces," *IEEE Transactions on Wireless Communications*, pp. 1–1, 2023.

[16] Z. Shao, X. Yuan, W. Zhang, and M. Di Renzo, "Joint localization and information transfer for reconfigurable intelligent surface aided full-duplex systems," *IEEE Transactions on Wireless Communications*, pp. 1–1, 2023.

[17] B. Ning, Z. Chen, W. Chen, and J. Fang, "Beamforming optimization for intelligent reflecting surface assisted MIMO: A sum-path-gain maximization approach," *IEEE Wireless Communications Letters*, vol. 9, no. 7, pp. 1105–1109, 2020.

[18] C. Pan, H. Ren, K. Wang, W. Xu, M. Elkashlan, A. Nallanathan, and L. Hanzo, "Multicell MIMO communications relying on intelligent reflecting surfaces," *IEEE Transactions on Wireless Communications*, vol. 19, no. 8, pp. 5218–5233, 2020.

[19] P. Zeng, D. Qiao, H. Qian, and Q. Wu, "Joint beamforming design for IRS aided multiuser MIMO with imperfect CSI," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 10, pp. 10 729–10 743, 2022.

[20] A. Sirojuddin, D. D. Putra, and W.-J. Huang, "Low-complexity sum-capacity maximization for intelligent reflecting surface-aided MIMO systems," *IEEE Wireless Communications Letters*, vol. 11, no. 7, pp. 1354–1358, 2022.

[21] J.-C. Chen, "Joint transceiver and intelligent reflecting surface design for mmwave massive MIMO systems," *IEEE Systems Journal*, vol. 17, no. 1, pp. 792–803, 2023.

[22] M. A. ElMossallamy, K. G. Seddik, W. Chen, L. Wang, G. Y. Li, and Z. Han, "Ris optimization on the complex circle manifold for interference mitigation in interference channels," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 6, pp. 6184–6189, 2021.

[23] C. G. Tsinos, A. Arora, S. Chatzinotas, and B. Ottersten, "Joint transmit waveform and receive filter design for dual-function radar-communication systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 15, no. 6, pp. 1378–1392, 2021.

[24] J. Tranter, N. D. Sidiropoulos, X. Fu, and A. Swami, "Fast unit-modulus least squares with applications in beamforming," *IEEE Transactions on Signal Processing*, vol. 65, no. 11, pp. 2875–2887, 2017.

[25] X. Song, J. Xu, F. Liu, T. X. Han, and Y. C. Eldar, "Intelligent reflecting surface enabled sensing: Cramér-Rao bound optimization," *IEEE Transactions on Signal Processing*, vol. 71, pp. 2011–2026, 2023.

[26] K. Zhong, J. Hu, C. Pan, M. Deng, and J. Fang, "Joint waveform and beamforming design for RIS-aided ISAC systems," *IEEE Signal Processing Letters*, vol. 30, pp. 165–169, 2023.

[27] H. Guo, Y.-C. Liang, J. Chen, and E. G. Larsson, "Weighted sum-rate maximization for reconfigurable intelligent surface aided wireless networks," *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3064–3076, 2020.

[28] C. Pan, G. Zhou, K. Zhi, S. Hong, T. Wu, Y. Pan, H. Ren, M. D. Renzo, A. Lee Swindlehurst, R. Zhang, and A. Y. Zhang, "An overview of signal processing techniques for RIS/IRS-aided wireless systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 5, pp. 883–917, 2022.

[29] X. Liu, Y. Yuan, T. Zhang, G. Cui, and W. P. Tay, "Integrated transmit waveform and ris phase shift design for lpi detection and communication," *IEEE Transactions on Wireless Communications*, pp. 1–1, 2023.

[30] X. Qian, X. Hu, C. Liu, M. Peng, and C. Zhong, "Sensing-based beamforming design for joint performance enhancement of ris-aided isac systems," *IEEE Transactions on Communications*, vol. 71, no. 11, pp. 6529–6545, 2023.

[31] K. Zhong, J. Hu, Z. Zhao, X. Yu, G. Cui, B. Liao, and H. Hu, "MIMO radar unimodular waveform design with learned complex circle manifold network," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 60, no. 2, pp. 1798–1807, 2024.

[32] C. Jiang, C. Zhang, C. Huang, J. Ge, J. He, and C. Yuen, "Secure beamforming design for ris-assisted integrated sensing and communication systems," *IEEE Wireless Communications Letters*, vol. 13, no. 2, pp. 520–524, 2024.

[33] N. T. Nguyen, N. Shlezinger, Y. C. Eldar, and M. Juntti, "Multiuser MIMO wideband joint communications and sensing system with subcarrier allocation," *IEEE Transactions on Signal Processing*, pp. 1–16, 2023.

[34] K. Zhong, J. Hu, J. Liu, D. An, C. Pan, K. C. Teh, X. Yu, and H. Li, "P2C2M: Parallel product complex circle manifold for RIS-aided ISAC waveform design," *IEEE Transactions on Cognitive Communications and Networking*, pp. 1–1, 2024.

[35] P. Wang, J. Fang, L. Dai, and H. Li, "Joint transceiver and large intelligent surface design for massive MIMO mmWave systems," *IEEE Transactions on Wireless Communications*, vol. 20, no. 2, pp. 1052–1064, 2021.

[36] K. Alhujaili, V. Monga, and M. Rangaswamy, "Transmit MIMO radar beampattern design via optimization on the complex circle manifold," *IEEE Transactions on Signal Processing*, vol. 67, no. 13, pp. 3561–3575, 2019.

[37] J. Li, G. Liao, Y. Huang, Z. Zhang, and A. Nehorai, "Riemannian geometric optimization methods for joint design of transmit sequence and receive filter on MIMO radar," *IEEE Transactions on Signal Processing*, vol. 68, pp. 5602–5616, 2020.

[38] H. Sato, "Riemannian conjugate gradient methods: General framework and specific algorithms with convergence analyses," *SIAM Journal on Optimization*, vol. 32, no. 4, pp. 2690–2717, 2022.

[39] H. Sakai, H. Sato, and H. Iiduka, "Global convergence of Hager–Zhang type riemannian conjugate gradient method," *Applied Mathematics and Computation*, vol. 441, p. 127685, 2023.

[40] N. Boumal, P.-A. Absil, and C. Cartis, "Global rates of convergence for nonconvex optimization on manifolds," *IMA Journal of Numerical Analysis*, vol. 39, no. 1, pp. 1–33, 2019.

[41] W. W. Hager and H. Zhang, "A survey of nonlinear conjugate gradient methods," *Pacific journal of Optimization*, vol. 2, no. 1, pp. 35–58, 2006.