

Low-Complexity Iterative Precoding Design for Near-field Multiuser Systems With Spatial Non-Stationarity

Mengyu Liu, Cunhua Pan, *Senior Member, IEEE*, Kangda Zhi, Hong Ren, *Member, IEEE*, Cheng-Xiang Wang, *Fellow, IEEE*, Jiangzhou Wang, *Fellow, IEEE*, and Yonina C. Eldar, *Fellow, IEEE*

Abstract—Extremely large antenna arrays (ELAA) are regarded as a promising technology for supporting sixth-generation (6G) networks. However, the large number of antennas significantly increases the computational complexity in precoding design, even for linearly regularized zero-forcing (RZF) precoding. To address this issue, a series of low-complexity iterative precoding are investigated. The main idea of these methods is to avoid matrix inversion of RZF precoding. Specifically, RZF precoding is equivalent to a system of linear equations that can be solved by fast iterative algorithms, such as random Kaczmarz (RK) algorithm. Yet, the performance of RK-based precoding algorithm is limited by the energy distributions of multiple users, which restricts its application in ELAA-assisted systems. To accelerate the RK-based precoding, we introduce the greedy random Kaczmarz (GRK)-based precoding by using the greedy criterion-based selection strategy. To further reduce the complexity of the GRK-based precoding, we propose a visibility region (VR)-based orthogonal GRK (VR-OGRK) precoding that leverages near-field spatial non-stationarity, which is characterized by the concept of VR. Next, by utilizing the information from multiple hyperplanes in each iteration, we extend the GRK-based precoding to the aggregation hyperplane Kaczmarz (AHK)-based precoding algorithm, which further enhances the convergence rate. Building upon the AHK algorithm, we propose a VR-based orthogonal AHK (VR-OAHK) precoding to further reduce the computational complexity. Furthermore, the proposed iterative precoding algorithms are proven to converge to RZF globally at an exponential rate. Simulation results show that the proposed algorithms achieve faster convergence and lower computational complexity than benchmark algorithms, and yield very similar performance to the RZF precoding.

Index Terms—Near-field communication, extremely large-scale antenna array, spatial non-stationarity, Kaczmarz algorithm.

I. INTRODUCTION

SIXTH-generation (6G) wireless communication networks are expected to offer much higher data rates and nearly ubiquitous coverage compared to fifth-generation (5G) wireless communication networks [1]–[4]. To meet these growing demands, several emerging technologies have been investigated, including millimeter-wave communication [5], [6], terahertz communication [7], [8], reconfigurable intelligent surface (RIS) [9]–[11], and extremely large antenna arrays (ELAA)

[12], [13]. Among these technologies, ELAA has recently garnered rapidly growing interest [14], [15]. By assembling a large number of antennas, ELAA achieves a high beam gain to compensate for the path loss associated with high signal frequency. Furthermore, with its large physical dimension, ELAA is expected to be integrated into large structures, such as shopping malls, smart factories, and high-speed rail stations, simultaneously supporting high-speed data transmission for a large number of devices [16].

The spectral efficiency of ELAA-assisted communications in hotspot scenarios with dense users could be limited if severe multiuser interference is not well tackled. In conventional massive MIMO systems, linear precoding schemes, such as zero-forcing (ZF) precoding and regularized zero-forcing precoding (RZF), are widely utilized [17], [18]. Due to the matrix inversion and the calculation of Gram matrix, the computational complexity of such linear precoding schemes is on the order of $\mathcal{O}(N_t K^2 + K^3)$, where N_t and K denote the number of antennas and users, respectively. Therefore, for ELAA-supported hotspot scenarios with very large number of N_t and K , even linear precoding could result in a prohibitive computational burden. To reduce the computational complexity, several low-complexity precoding algorithms have been proposed [19]–[24]. The main objective of these algorithms is to approximate matrix inversion in RZF precoding, achieving a balance between performance and computational complexity.

The low-complexity algorithms can be classified into two categories: truncated polynomial expansion (TPE) algorithms and iterative algorithms. In the former, the matrix inversion is approximated by summing a limited series of matrix multiplications. For example, in [19] and [20], the Neumann Series expansion was utilized to approximate the matrix inversion in RZF precoding. However, even with this algorithm, the series of matrix multiplications still incur relatively high computational complexity. In contrast, low-complexity iterative precoding algorithms are promising due to their faster convergence rate than TPE. Specifically, the iterative algorithms first transforms the RZF precoding design problem into solving a system of linear equations. Then, a series of iterative methods for solving systems of linear equations is applied to achieve low-complexity precoding designs, including the random Kaczmarz (RK) algorithm [21], the Jacobi iteration [22], the Richardson iteration [23], and the Gauss Seidel iteration [24]. The RK algorithm is widely used to solve large systems of linear equations due to its simplicity of application and fast

M. Liu, C. Pan, H. Ren, C. -X. Wang, J. Wang are with National Mobile Communications Research Laboratory, Southeast University, Nanjing 211189, China. (e-mail: mengyuliu, cpan, hren, chxwang, j.z.wang@seu.edu.cn). K. Zhi is with the School of Electrical Engineering and Computer Science, Technical University of Berlin, Berlin 10623 (e-mail: k.zhi@tu-berlin.de). Y. C. Eldar is with the Faculty of Mathematics and Computer Science, Weizmann Institute of Science, 7610001 Rehovot, Israel (e-mail: yonina.eldar@weizmann.ac.il).
Corresponding authors: Cunhua Pan and Hong Ren.

convergence, making it particularly suitable for ELAA-assisted systems. The main idea of the RK algorithm is to project the solution for the current iteration onto the hyperplane defined by the selected equation [25]. Based on [25], the authors of [21] proposed the RK algorithm-based precoding algorithm to effectively reduce the computational complexity where, in each iteration, one of the systems of equations is selected based on a pre-defined probability distribution depending on the energy of the equation.

The performance of the RK-based precoding algorithm in [21] is limited in the following two cases: (1) the channel vector energies of the users are similar, and (2) when users are located at the edge of the cell, the low-energy users will be selected with small probability [26]–[28]. In addition, due to the large number of antennas and users in the ELAA-assisted multiuser system, it is essential to further improve convergence rate of existing RK algorithms and further reduce the computational complexity in the precoding design. Hence, we focus on designing low-complexity and fast-converging RK-based precoding algorithms, making it feasible and promising for ELAA-assisted multiuser systems.

Besides, compared with existing RK-based precoding algorithms, another difference that should be considered is the near-field spatial non-stationarity introduced by ELAA. The Rayleigh distance, defined as $\frac{2D^2}{\lambda}$, indicates the boundary between the near-field and far-field regions, where D and λ represent the array aperture and carrier wavelength, respectively. The large array aperture of ELAA will significantly enlarge the Rayleigh distance, extending it from a few meters to several hundred meters. Consequently, for ELAA-assisted communication systems, near-field spherical wave channel models should be considered instead of far-field planar wave channel models [29]–[31]. Due to the propagation property of near-field spherical wave, different parts of ELAA may have different views of the propagation environment, which is called the spatial non-stationarity [32]. Specifically, the signal power received by the user may be primarily contributed by a portion of ELAA, termed as visibility region (VR), while the contribution of signal strength from other parts of the ELAA could be marginal due to attenuation and obstacles. In [33], an RK-based uplink detection algorithm was investigated with VR. The authors of [34] analyzed the interference between subarrays by considering spatial non-stationarity and proposed an RK-based precoding design algorithm. However, existing works mainly focused on the impact of VR on channel modeling, overlooking the potential of leveraging VR information to reduce computational complexity in precoding design.

To overcome the limitations of the existing RK-based precoding and further reduce the computational complexity, we explore a low-complexity near-field precoding design for multiuser ELAA-based systems. The main contributions of this paper are summarized as follows:

- To overcome existing limitations and accelerate convergence of the RK-based precoding, we introduce the greedy random Kaczmarz (GRK)-based precoding, which selects hyperplane with larger residual instead of that with larger energy in each iteration. Next, to further reduce the computational complexity of computing the residuals

in each iteration, we propose a VR-based orthogonal greedy random Kaczmarz (VR-OGRK)-based precoding by utilizing the spatial non-stationarity introduced by ELAA. Specifically, non-overlapping VRs ensure channel vector orthogonality between users, which significantly reduces the number of residuals computations per iteration. Furthermore, we show that the VR-OGRK algorithm converges globally at an exponential rate in terms of the normalized mean squared error (NMSE).

- Subsequently, by introducing the concept of an aggregation hyperplane, we extend GRK-based precoding to aggregation hyperplane Kaczmarz (AHK)-based precoding. In each iteration, the AHK algorithm utilizes information from multiple hyperplanes, rather than relying on a single hyperplane, thereby effectively accelerates the convergence rate. Then, we derive that when the AHK algorithm is applied to an orthogonal set, which contains the hyperplanes that are orthogonal to each other, it benefits from both a fast convergence rate and the potential for parallel computation. To further reduce the complexity, we propose a VR-based orthogonal AHK (VR-OAHK) algorithm. Specifically, we partition the users into orthogonal and non-orthogonal sets based on an undirected graph derived from VRs of users. In each iteration, the AHK algorithm is applied sequentially to the orthogonal set and non-orthogonal set. Besides, it is proved that the VR-OAHK algorithm converges globally at an exponential rate.
- Simulation results are provided to demonstrate the effectiveness of the proposed low-complexity algorithms. It is shown that the performances of the algorithms are close to that of RZF precoding, while the computational complexity is significantly reduced compared to existing RK-based precoding design.

The remainder of this paper is organized as follows. In Section II, we introduce the system model with spatial non-stationarity, and formulate the precoding design problem. In Section III, we briefly introduce an existing RK-based precoding algorithm. In Section IV, by exploiting the features of VR information, we propose the VR-OGRK-based precoding and demonstrate its globally exponential convergence. We then propose the VR-OAHK-based precoding in Section V and prove its globally exponential convergence. Section VI presents the computational complexity analysis for the proposed algorithms. Finally, Sections VII and VIII present simulation results and conclusions, respectively.

Notations: In this paper, scalars, vectors, and matrices are represented by lowercase, bold lowercase, and bold uppercase letters, respectively. The identity matrix and the all-zero vector are denoted by \mathbf{I} and $\mathbf{0}$, respectively. A complex matrix of dimensions $M \times N$ is denoted as $\mathbb{C}^{M \times N}$. The operation of expectation is represented by $\mathbb{E}(\cdot)$. The function $\text{diag}(\cdot)$ denotes the diagonalization operation. The absolute value and the real part of a complex number a are denoted by $|a|$ and $\text{Re}(a)$, respectively. The Euclidean norm of a vector \mathbf{a} is represented as $\|\mathbf{a}\|_2$. For a matrix \mathbf{A} , $\text{Tr}(\mathbf{A})$ refers to its trace, and $\|\mathbf{A}\|_F$ represents its Frobenius norm. The Hadamard

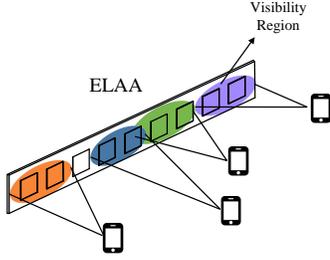


Fig. 1. ELAA-based near-field multiuser system with spatial non-stationarities.

product of two matrices \mathbf{A} and \mathbf{B} is expressed as $\mathbf{A} \odot \mathbf{B}$. A vector following a normal distribution with zero mean and unit covariance matrix is denoted as $\mathcal{CN}(\mathbf{0}, \mathbf{I})$. The conjugate, transpose, and Hermitian operations are represented by $(\cdot)^*$, $(\cdot)^T$, and $(\cdot)^H$, respectively.

II. SYSTEM MODEL

We consider a downlink ELAA-based near-field multiuser system, as illustrated in Fig. 1. In this system, the ELAA with an N_t -antenna uniform linear array (ULA) simultaneously serves K ($K < N_t$) single-antenna users. The antenna spacing of the ELAA is denoted by $d = \frac{\lambda}{2} = \frac{c}{2f}$, where λ , f , and c are the carrier wavelength, frequency, and the speed of light, respectively. For tractability, we set the center of the ELAA as the origin with the n -th antenna located at $(nd, 0)$, where $n = -\frac{N_t-1}{2}, \dots, \frac{N_t-1}{2}$.

A. Near-Field Channel Model with Spatial Non-Stationarity

Due to the large aperture of the antenna array, we assume that all users and scatterers are distributed within the near-field region of the ELAA. Consequently, the conventional far-field planar wave approximation is no longer valid, and a channel model based on spherical waves should be used to accurately capture the wireless propagation characteristics in the near field. The near-field channel between user k and the base station (BS) is modeled as [29]

$$\bar{\mathbf{h}}_k = \sqrt{\frac{N_t}{L_k}} (\beta_{k,1} \mathbf{a}(\theta_{k,1}, \mathbf{d}_{k,1}) + \sum_{l=2}^{L_k} \beta_{k,l} \mathbf{a}(\theta_{k,l}, \mathbf{d}_{k,l})), \quad (1)$$

where $\theta_{k,l}$, $\mathbf{d}_{k,l} = [\mathbf{d}_{k,l}^{(-\frac{N_t-1}{2})}, \dots, \mathbf{d}_{k,l}^{(0)}, \dots, \mathbf{d}_{k,l}^{(\frac{N_t-1}{2})}]^T \in \mathbb{C}^{N_t \times 1}$, and L_k denotes the angle of arrival (AoA) of the l -th path, the vector containing distances between the l -th scatterer (user) and all antennas at BS, and the number of paths within the channel of user k , respectively. For ease of representation, the first path ($l = 1$) represents the line-of-sight (LoS) path. The path of $2 \leq l \leq L_k$ is the non-LoS path contributed by the l -th scatterer of user k , whose location is denoted by $(\mathbf{d}_{k,l}^{(0)} \cos \theta_{k,l}, \mathbf{d}_{k,l}^{(0)} \sin \theta_{k,l})$, where $k = 1, \dots, K$. Based on the law of cosines, the distance between the l -th scatterer and the n -th antenna is calculated as $\mathbf{d}_{k,l}^{(n)} = \sqrt{(\mathbf{d}_{k,l}^{(0)})^2 + n^2 d^2 - 2nd \mathbf{d}_{k,l}^{(0)} \cos \theta_{k,l}}$. The near-field array response vector $\mathbf{a}(\theta_{k,l}, \mathbf{d}_{k,l}) \in \mathbb{C}^{N_t \times 1}$ is expressed as

$$\mathbf{a}(\theta_{k,l}, \mathbf{d}_{k,l}) = \frac{1}{\sqrt{N_t}} [e^{-j\frac{2\pi}{\lambda}(\mathbf{d}_{k,l}^{(-\frac{N_t-1}{2})} - \mathbf{d}_{k,l}^{(0)})}, \dots, e^{-j\frac{2\pi}{\lambda}(\mathbf{d}_{k,l}^{(\frac{N_t-1}{2})} - \mathbf{d}_{k,l}^{(0)})}]^T. \quad (2)$$

In addition to the near-field spherical wave effect, the ELAA introduces spatial non-stationarities, making users receive signals only from a small portion of the whole antenna array [16]. This phenomenon is characterized by VR, as illustrated in Fig. 1. Specifically, the spatial non-stationarity of the ELAA is mainly attributed to the following two reasons: 1. the large physical size of the entire array causes the channel amplitude to vary across the array and therefore the majority of the signal power is contributed by proximity subarrays; 2. depending on the location and shape of obstacles between the ELAA and the user, significant portions of the antenna array may be affected by blockage, forming different VRs for different users [35], [36]. Considering the spatial non-stationarity effect in the near field, the channel between ELAA and user k is rewritten as

$$\mathbf{h}_k = \bar{\mathbf{h}}_k \odot \mathbf{u}_k, \quad (3)$$

where $\mathbf{u}_k \in \mathbb{C}^{N_t \times 1}$ denotes the indicator vector of array VR, which is defined as follows:

$$[\mathbf{u}_k]_n = \begin{cases} 1, & n \in \Gamma_k, \\ 0, & n \notin \Gamma_k, \end{cases} \quad (4)$$

where Γ_k represents the set of the indices of antennas that are visible to user k .

Based on the experimental measurements in [35], the whole array of ELAA can be evenly divided into several subarrays, each exhibiting spatial stationarity. Thus, in this paper, we assume that the whole array is divided into S subarrays. Specifically, each subarray possesses $\frac{N_t}{S}$ antennas, which is assumed to be an integer. Therefore, the VR indicator vector (4) can be rewritten with respect to subarrays as

$$[\mathbf{u}_k]_{\frac{(s-1)N_t}{S} + 1: \frac{sN_t}{S}} = \begin{cases} \mathbf{1}_{\frac{N_t}{S}}, & \text{if } s \in \mathcal{M}_k, \\ \mathbf{0}_{\frac{N_t}{S}}, & \text{else,} \end{cases} \quad (5)$$

where \mathcal{M}_k denotes the set of the indices of subarrays that are visible to user k .

B. Signal Transmission Model

The signal received at user k is represented as

$$y_k = \sqrt{\rho} \mathbf{h}_k^H \mathbf{F} \mathbf{s} + n_k = \sqrt{\rho} \mathbf{h}_k^H \sum_{k=1}^K \mathbf{f}_k s_k + n_k, \quad (6)$$

where $\mathbf{F} = [\mathbf{f}_1, \dots, \mathbf{f}_k, \dots, \mathbf{f}_K] \in \mathbb{C}^{N_t \times K}$ represents the digital precoding matrix. $\mathbf{s} = [s_1, \dots, s_K]^T \in \mathbb{C}^{K \times 1}$ denotes the transmitted symbol satisfying $\mathbb{E}[\mathbf{s}\mathbf{s}^H] = \mathbf{I}_K$. ρ represents the average transmitted signal power and n_k represents thermal noise at the user, which follows a complex Gaussian distribution $\mathcal{CN}(0, \sigma^2)$ with σ^2 denoting the noise power. The spectral efficiency (bit/s/Hz) of user k is then calculated as

$$R_k = \log_2 \left(1 + \frac{\rho |\mathbf{h}_k^H \mathbf{f}_k|^2}{\rho \sum_{i=1, i \neq k}^K |\mathbf{h}_k^H \mathbf{f}_i|^2 + \sigma^2} \right). \quad (7)$$

C. Problem Formulation

We consider the problem of designing the precoding matrix \mathbf{F} to maximize the sum spectral efficiency (7), which is formulated as

$$\max_{\mathbf{F}} \sum_{k=1}^K R_k \quad (8a)$$

$$\text{s.t.} \quad \|\mathbf{F}\|_F^2 \leq 1, \quad (8b)$$

where constraint (8b) denotes the power budget of the BS.

Linear precoding designs, such as ZF and RZF, have been widely utilized to address (8), since they are effective with low complexity in conventional massive MIMO systems. The main idea of ZF is to project the precoding vector of user k onto the null space of the remaining users' channels, ensuring that the transmitted signal to the target user is free from interference from other users. RZF offers a compromise between interference suppression and noise enhancement. Specifically, the precoding matrix is computed as

$$\mathbf{F}^{\text{RZF}} = \beta \mathbf{H}(\mathbf{H}^H \mathbf{H} + \xi \mathbf{I}_K)^{-1}, \quad (9)$$

where β , $\xi = \frac{1}{\text{SNR}} = \frac{\sigma^2}{\rho}$ denote the power normalized factor for meeting power constraint (8b) and the regularization factor, respectively. $\mathbf{H} = [\mathbf{h}_1, \dots, \mathbf{h}_K] \in \mathbb{C}^{N_t \times K}$ denotes the channel matrix of all users. Note that as $\xi \rightarrow \infty$, RZF precoding reduces to maximum ratio combining (MRC) scheme. Conversely, when $\xi = 0$, RZF precoding simplifies to ZF precoding.

However, due to matrix inversion, the computational complexity of the RZF precoding is $\mathcal{O}(N_t K^2 + K^3)$, which increases with the number of antennas and users. Thus, considering ELAA-assisted hotspot areas with dense users, the computational complexity associated with the RZF precoding could be prohibitive, which limits the practical implementation of ELAA. To address this challenge, a series of iterative-based precoding methods are investigated [21]–[24]. Among them, the RK-based precoding is widely used due to its simplicity of application and fast convergence. Thus, in this paper, we first briefly introduce the RK-based precoding algorithm and analyze the limitations of the RK-based precoding algorithm in later sections.

III. LOW-COMPLEXITY RANDOM KACZMARZ ALGORITHM-BASED PRECODING DESIGN

In this section, we present RK-based precoding algorithm [21], [34] as a benchmark. Subsequently, we analyze the limitations of existing RK-based precoding algorithm.

A. Principle of Random Kaczmarz (RK) Algorithm

The Kaczmarz algorithm is an iterative method suitable for solving large systems of linear equations [25], [33]. Specifically, given a system of linear equations $\mathbf{A}\mathbf{x} = \mathbf{b}$, where $\mathbf{A} = [\mathbf{a}_1^H; \dots; \mathbf{a}_M^H] \in \mathbb{C}^{M \times N}$, $\mathbf{x} \in \mathbb{C}^{N \times 1}$, and $\mathbf{b} \in \mathbb{C}^{M \times 1}$. At the t -th iteration, the i -th equation $\mathbf{a}_i^H \mathbf{x}^t = b_i$ is selected and the solution for the next iteration $\mathbf{x}^{(t+1)}$ is updated as [25]

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \frac{b_i - \mathbf{a}_i^H \mathbf{x}^{(t)}}{\|\mathbf{a}_i^H\|_2^2} \mathbf{a}_i. \quad (10)$$

From (10), the main idea of the RK algorithm is to project \mathbf{x}^t onto the hyperplane defined by the selected equation $\mathbf{a}_i^H \mathbf{x}^t = b_i$. At each iteration, the RK algorithm selects the hyperplane based on the pre-determined probability distribution. The pre-determined probability distribution, often referred to as the

selection strategy, is typically defined as $\mathbf{p}_i = \frac{\|\mathbf{a}_i^H\|_2^2}{\|\mathbf{A}\|_F^2}$ (energy criterion) or $\mathbf{p}_i = \frac{1}{M}$ (uniform criterion), $\forall i \in [1, M]$, where \mathbf{p}_i denotes the probability of selecting the i -th row of the systems of linear equations.

B. Random Kaczmarz (RK)-based precoding design

By applying the RZF precoding (9), the received signal of all users is represented as follows:

$$\begin{aligned} \mathbf{y} &= \mathbf{H}^H \mathbf{F}^{\text{RZF}} \mathbf{s} + \mathbf{n} = \beta \mathbf{H}^H \mathbf{H} (\mathbf{H}^H \mathbf{H} + \xi \mathbf{I}_K)^{-1} \mathbf{s} + \mathbf{n} \\ &= \beta \mathbf{H}^H \mathbf{H} \mathbf{v} + \mathbf{n}, \end{aligned} \quad (11)$$

where $\mathbf{v} = (\mathbf{H}^H \mathbf{H} + \xi \mathbf{I}_K)^{-1} \mathbf{s} \in \mathbb{C}^{K \times 1}$ denotes the auxiliary vector. Subsequently, the acquisition of auxiliary vector $\mathbf{v} = (\mathbf{H}^H \mathbf{H} + \xi \mathbf{I}_K)^{-1} \mathbf{s}$ is readily transformed into an equivalent least square problem, as described by the following theorem.

Theorem 1: The auxiliary vector $\mathbf{v} = (\mathbf{H}^H \mathbf{H} + \xi \mathbf{I}_K)^{-1} \mathbf{s}$ can be obtained from an equivalent least square problem

$$\mathbf{v} = \arg \min_{\mathbf{v}} \|\mathbf{G}\mathbf{v} - \mathbf{g}\|_2^2, \quad (12)$$

where $\mathbf{G} = [\mathbf{H}; \sqrt{\xi} \mathbf{I}_K] \in \mathbb{C}^{(N_t+K) \times K}$, $\mathbf{g} = [\mathbf{0}_{N_t}; \frac{\mathbf{s}}{\sqrt{\xi}}] \in \mathbb{C}^{(N_t+K) \times 1}$.

Proof: See Appendix A. ■

According to (11) and (12), if we get \mathbf{v} , the precoded signal can be easily obtained by $\mathbf{H}\mathbf{v} = \mathbf{F}\mathbf{s}$. Thus, RZF precoding (9) is equivalent to finding \mathbf{v} in the least square problem of (12). If \mathbf{g} lies in the range space spanned by \mathbf{G} , Problem (12) can be further reformulated as a system of linear equations, i.e., $\mathbf{G}\mathbf{v} = \mathbf{g}$, which can be effectively solved by the RK algorithm as we discussed in Section III. A. However, this system of linear equations is inconsistent unless $\mathbf{s} = \mathbf{0}$. This inconsistency makes it challenging to obtain an accurate solution for \mathbf{v} . In other words, the system of linear equations $\mathbf{G}\mathbf{v} = \mathbf{g}$ may not have a solution.

To tackle this issue, the inconsistency in the systems of linear equations needs to be removed [37]. Specifically, we first decompose \mathbf{g} as follows

$$\mathbf{g} = \mathbf{w} + \boldsymbol{\psi}, \quad (13)$$

with $\mathbf{w} \in \mathcal{R}(\mathbf{G})$, $\boldsymbol{\psi} \in \mathcal{R}(\mathbf{G})^\perp = \mathcal{K}(\mathbf{G}^H)$, where $\mathcal{R}(\cdot)$, $\mathcal{R}(\cdot)^\perp$ and $\mathcal{K}(\cdot)$ represents the range space, the orthogonal complementary space of the range space, and the kernel space. Thus, if we can accurately estimate \mathbf{w} from \mathbf{g} , the solution of the system of linear equations $\mathbf{G}\mathbf{v} = \mathbf{w}$ can be approximated as the solution of Problem (12). To accurately estimate \mathbf{w} , we establish a system of linear equations as follows

$$\mathbf{G}^H \mathbf{w} = \mathbf{G}^H (\mathbf{g} - \boldsymbol{\psi}) = \mathbf{G}^H \mathbf{g} = \mathbf{s}. \quad (14)$$

By using the RK algorithm to solve the system of linear equations $\mathbf{G}^H \mathbf{w} = \mathbf{s}$, we obtain the estimated value \mathbf{w} . Note that, \mathbf{w} lies in the space spanned by the columns of \mathbf{G} , as indicated by (10). Next, the consistent over-determined system of linear equations is formulated as

$$\mathbf{G}\mathbf{v} = \mathbf{w}. \quad (15)$$

Finally, we obtain the value of \mathbf{v} by solving this equation. Based on the above analysis, the solution of the least square

Algorithm 1 RK-based precoding algorithm

- 1: **Input:** \mathbf{H} , N_t , K , ξ , the tolerance error ϵ .
 - 2: Initialize $\mathbf{V} = \mathbf{0}_{K \times K}$, $t = 0$;
 - 3: Initialize $\mathbf{m}_k = \mathbf{0}$, $\mathbf{q}_k = \mathbf{0}$, $\forall k \in [1, K]$;
 - 4: Calculate and store $\|\mathbf{h}_i\|_2^2 + \xi$;
 - 5: **while** NMSE $\geq \epsilon$ **do**
 - 6: Run the RK algorithm K times:
 - 7: Pick the i -th row \mathbf{H}^H based on the pre-determined probability distribution;
 - 8: Calculate $[\mathbf{r}_k^{(t)}]_i = [\mathbf{e}_k]_i - \mathbf{h}_i^H \mathbf{m}_k^{(t)} - \xi [\mathbf{q}_k]_i^{(t)}$;
 - 9: Calculate $[\gamma_k^{(t)}]_i = [\mathbf{r}_k^{(t)}]_i / (\|\mathbf{h}_i^H\|_2^2 + \xi)$;
 - 10: Update $\mathbf{m}_k^{(t+1)} = \mathbf{m}_k^{(t)} + [\gamma_k^{(t)}]_i \mathbf{h}_i$;
 - 11: Update $[\mathbf{q}_k]_i^{(t+1)} = [\mathbf{q}_k]_i^{(t)} + [\gamma_k^{(t)}]_i$;
 - 12: $t = t + 1$;
 - 13: **end while**
 - 14: $\mathbf{V} = [\mathbf{q}_1, \dots, \mathbf{q}_K]$, Calculate \mathbf{F} as (17);
 - 15: **Output:** \mathbf{F} , $T = t$.
-

problem (12) can be obtained in two steps. Specifically, in Step 1, we solving the system of linear equation (14) by using the RK algorithm to get the estimated value \mathbf{w} . In Step 2, after obtaining \mathbf{w} , the system of linear equations (15) is also solved by using RK algorithm. Furthermore, we can conclude the following remarks.

Remark 1: We first define the estimated value $\mathbf{w} \in \mathbb{C}^{(N_t+K) \times 1}$ as $[\mathbf{m}; \sqrt{\xi} \mathbf{q}]$, where vector \mathbf{q} is of length K . Due to the special structure of \mathbf{G} , we have

$$\mathbf{G}\mathbf{v} = [\mathbf{H}; \sqrt{\xi} \mathbf{I}_K] \mathbf{v} = [\mathbf{H}\mathbf{v}; \sqrt{\xi} \mathbf{v}]. \quad (16)$$

Thus, the estimated solution of \mathbf{v} is readily determined by extracting the scaled last K elements of \mathbf{w} , i.e., \mathbf{q} , rather than solving the system of linear equations (15).

Remark 2: In the downlink precoding design, the objective is to compute \mathbf{F}^{RZF} . However, due to the relationship $\mathbf{H}\mathbf{v} = \mathbf{F}^{\text{RZF}}\mathbf{s}$, we can only obtain the product $\mathbf{F}^{\text{RZF}}\mathbf{s}$, given solution \mathbf{v} . Therefore, to derive the precoding \mathbf{F}^{RZF} , we propose to use the parallel RK algorithm as introduced in [21], [33]. Specifically, we run the RK algorithm in parallel K times by setting $\mathbf{s}_k = \mathbf{e}_k \in \mathbb{C}^{K \times 1}$, where \mathbf{e}_k denotes the k -th canonical basis. Then, the precoding matrix can then be computed as

$$\mathbf{F}^{\text{RZF}} = \mathbf{H} \times [\mathbf{v}_1, \dots, \mathbf{v}_K] = \mathbf{H}\mathbf{V}. \quad (17)$$

Based on the above discussions, the RK-based precoding algorithm is detailed in Algorithm 1, which reduces the computational complexity of RZF from $\mathcal{O}(N_t K^2 + K^3)$ to $\mathcal{O}(TN_t)$. In Algorithm 1, we apply the RK algorithm to (14), i.e., $\mathbf{G}^H \mathbf{w} = \mathbf{s}$. Then, \mathbf{v} is updated based on Remark 1. Specifically, Steps 8 and 9 are to calculate the update weights, which is defined as $\frac{b_i - \mathbf{a}_i^H \mathbf{x}^{(t)}}{\|\mathbf{a}_i\|_2^2}$ in (10). In Steps 10 and 11, the value of solution $\mathbf{w} = [\mathbf{m}; \sqrt{\xi} \mathbf{q}]$ is updated according to (10).

C. Limitations of the RK-based Precoding Algorithm

As shown in Algorithm 1, the pre-determined probability distribution of the RK algorithm determines which hyperplane is selected at each iteration. Thus, the convergence rate is

mainly depending on the design of the pre-determined probability distribution. The existing RK-based precoding algorithms [21], [34] usually utilize the energy criterion-based selection strategy, which has a faster convergence rate than uniform criterion-based selection strategy [25]. Specifically, as discussed in Section III. B, the RK algorithm is applied to solve the system of linear equations $\mathbf{G}^H \mathbf{w} = \mathbf{s}$. Recall that $\mathbf{G} = [\mathbf{H}; \sqrt{\xi} \mathbf{I}_K]$, the physical significance of the RK-based precoding algorithm is that the hyperplane defined in terms of the user channel vector is selected at each iteration based on the selection strategy. Thus, existing RK-based precoding algorithms select the high-energy user with a higher probability in each iteration.

However, the performance of existing RK-based precoding algorithms utilizing energy criterion-based selection strategy is limited in the following two cases: (1) when users are located at the edge of the cell, and (2) the channel vector energies of the users are similar. For the first case, the RK-based precoding algorithm always selects users with higher energy and users with lower-energy are difficult to be selected. For the second case, the energy criterion will degenerate into a uniform criterion. In both of the above scenarios, the convergence speed of the RK-based precoding algorithm is drastically reduced. Both of the above cases are common in the ELAA-assisted hotspot scenarios in future 6G networks. Furthermore, existing RK-based precoding algorithms do not consider near-field spatial non-stationarity introduced by ELAA. According to (3) and (5), the channel vectors of users have a large number of zero elements. By fully exploiting the near-field non-stationarity, the complexity of RK-based precoding can be further reduced. Thus, to overcome these limitations and further reduce the computational complexity, we propose low-complexity iterative precoding algorithms.

IV. VR-BASED ORTHOGONAL GREEDY RANDOM KACZMARZ ALGORITHM

In this section, we first introduce the greedy random Kaczmarz (GRK)-based precoding algorithm by designing the greedy criterion-based selection strategy. Subsequently, to further reduce the computational complexity and exploit the spatial non-stationarity, we propose the VR-based Orthogonal GRK (VR-OGRK) algorithm and provide the convergence analysis of the proposed algorithm.

A. Greedy Random Kaczmarz (GRK)-based precoding

As discussed in Section III. C, the energy criterion-based selection strategy is not applicable to the RK-based precoding algorithm. Therefore, we introduce the greedy random Kaczmarz (GRK)-based precoding algorithm by using the greedy criterion-based selection strategy. Specifically, Different from existing RK-based precoding, the GRK-based precoding algorithm employs a selection strategy that prioritizes the i -th row with larger residual, i.e., with larger $r_i = b_i - \mathbf{a}_i^H \mathbf{x}^{(t)}$ [38], [39]. The greedy selection strategy of GRK-based precoding is defined as

$$\mathbf{p}_i = \frac{|r_i|^2}{\|\mathbf{r}\|_2^2}, \quad (18)$$

where $\mathbf{r} = [r_1; \dots; r_i; \dots; r_M]$ denotes the residual vector.

Algorithm 2 VR-OGRK-based precoding algorithm

- 1: **Input:** \mathbf{H} , \mathbf{H}_s , N_t , K , ξ , \mathcal{X}_k , \mathcal{M}_k , \mathcal{Q}_s , error threshold ϵ .
 - 2: Repeat Algorithm 1 Steps 2-4;
 - 3: Initialize the residuals $\mathbf{r}_k = \mathbf{e}_k$, $\forall k \in [1, K]$
 - 4: Calculate the probability vector $[\mathbf{p}_k]_i = \frac{|\mathbf{r}_k|_i|^2 / \|\mathbf{r}_k\|_2^2}{\|\mathbf{r}_k\|_2^2}$, $\forall i, k \in [1, K]$;
 - 5: Construct $\mathbf{h}_k = [\mathbf{h}_k]_{\Gamma_k}$;
 - 6: **while** NMSE $\geq \epsilon$ **do**
 - 7: Run VR-OGRK algorithm K times:
 - 8: $[\mathbf{r}_k^{(t)}]_{\mathcal{X}_i(t-1)} = [\mathbf{e}_k]_{\mathcal{X}_i(t-1)} - \tilde{\mathbf{h}}_{\mathcal{X}_i(t-1)}^{\mathbf{H}} \mathbf{m}_k^{(t)} - \xi [\mathbf{q}_k^{(t)}]_{\mathcal{X}_i(t-1)}$;
 - 9: Update the probability vector as Step 4;
 - 10: Pick the $i^{(t)}$ -th row $\mathbf{H}^{\mathbf{H}}$ according to \mathbf{p}_k ;
 - 11: Repeat Algorithm 1 Steps 8-12;
 - 12: **end while**
 - 13: $\mathbf{V} = [\mathbf{q}_1, \dots, \mathbf{q}_K]$, Calculate \mathbf{F} as (20);
 - 14: **Output:** \mathbf{F} , $T = t$.
-

The GRK-based precoding effectively overcomes the limitations of existing RK-based precoding and accelerate the convergence. However, the GRK-based precoding algorithm introduces additional complexity in calculating all residuals \mathbf{r} at each iteration. To further reduce the complexity, we propose the VR-based orthogonal GRK (VR-OGRK)-based precoding algorithm by using the spatial non-stationarity.

B. VR-based Orthogonal GRK (VR-OGRK)-based precoding

To effectively exploit the spatial non-stationarity brought by the ELAA-assisted system, we introduce the following lemma.

Lemma 1: If the VRs of user i and user j do not overlap, the channels between the two users are mutually orthogonal, i.e. $\mathbf{h}_i^{\mathbf{H}} \mathbf{h}_j = 0$, if $\mathcal{M}_i \cap \mathcal{M}_j = \emptyset$.

Proof: This can be readily proved by calculating the inner product of the two users' channel vectors. ■

Based on Lemma 1, we establish a non-orthogonal set \mathcal{X}_k corresponding to user k , where the elements of this set represent the indices of users whose VRs overlap with that of user k . For the system of linear equations $\mathbf{G}^{\mathbf{H}} \mathbf{w} = \mathbf{s}$, the k -th row of $\mathbf{G}^{\mathbf{H}}$ can be expressed as $[\mathbf{h}_k, \sqrt{\xi} \mathbf{e}_k]$. Thus, if VRs of user i and user j do not overlap, the i -th row and the j -th row of $\mathbf{G}^{\mathbf{H}}$ are also mutually orthogonal. Theorem 2 elaborates how to apply the orthogonality in Kaczmarz algorithm.

Theorem 2: Consider the linear equations $\mathbf{A} \mathbf{x} = \mathbf{b}$, solved by the Kaczmarz algorithm. At the t -th iteration, if the i -th row of \mathbf{A} is selected, the residuals corresponding to the rows that are orthogonal to the i -th row will remain unchanged, i.e.,

$$r_j^{(t+1)} = r_j^{(t)}, \text{ if } \mathbf{a}_j^{\mathbf{H}} \mathbf{a}_i = 0, \quad (19)$$

where $r_j^{(t)} = b_j - \mathbf{a}_j^{\mathbf{H}} \mathbf{x}^{(t)}$ denotes the residual corresponding to the j -th row at the i -th iteration.

Proof: See Appendix B. ■

According to Lemma 1 and Theorem 2, when selecting the k -th row of $\mathbf{G}^{\mathbf{H}}$, only the residuals associated with the rows in the non-orthogonal set \mathcal{X}_k are affected.

After building the set \mathcal{X}_k based on VR information, we can only update the residuals corresponding to the rows whose

indices are in \mathcal{X}_k , rather than all rows in each iteration. This method effectively reduces the complexity of the GRK-based precoding algorithm. Furthermore, the operation $\mathbf{a}_i^{\mathbf{H}} \mathbf{x}$ incurs a computational cost of $\mathcal{O}(N_t)$, which brings the primary computation burden at each iteration. Due to the presence of spatial non-stationarity, we leverage the VR information of users to further reduce the computational cost in this operation. Specifically, we construct the effective channel $\tilde{\mathbf{h}}_k = [\mathbf{h}_k]_{\Gamma_k} \in \mathbb{C}^{|\Gamma_k| \times 1}$. Thus, the complexity of the original operation is further reduced to $\mathcal{O}(|\Gamma|)$, assuming that $|\Gamma_k| = |\Gamma|$ for all users, which simplifies the analysis.

In Step 14 of Algorithm 1, the computational cost of $\mathbf{F} = \mathbf{H} \mathbf{V}$ scales on the order of $\mathcal{O}(N_t K^2)$. However, existing literature usually ignored the complexity introduced by this term, which leads to significant computational burden in precoding design. Therefore, we will again leverage the VR information to further reduce the complexity in this step.

We first define $\mathbf{H}_s \in \mathbb{C}^{\frac{N_t}{S} \times K}$ and \mathcal{Q}_s as the channel matrix between the s -th subarray and users and the set of indices of users whose VRs include the s -th subarray, respectively. Then, based on the VR information \mathcal{Q}_s , we construct the equivalent matrix $[\mathbf{V}]_{\mathcal{Q}_s} \in \mathbb{C}^{|\mathcal{Q}_s| \times K}$ and $[\mathbf{H}_s]_{\mathcal{Q}_s} \in \mathbb{C}^{\frac{N_t}{S} \times |\mathcal{Q}_s|}$. The precoding matrix is further calculated as

$$\mathbf{F} = [[\mathbf{H}_1]_{\mathcal{Q}_1} [\mathbf{V}]_{\mathcal{Q}_1}; \dots; [\mathbf{H}_S]_{\mathcal{Q}_S} [\mathbf{V}]_{\mathcal{Q}_S}]. \quad (20)$$

For analytical tractability, we assume that $|\mathcal{Q}_s| = |\mathcal{Q}|$ for all subarrays. Then, it is observed that the complexity of calculating \mathbf{F} based on (20) is reduced from $\mathcal{O}(N_t K^2)$ to $\mathcal{O}(N_t |\mathcal{Q}| K)$, which is beneficial for ELAA systems.

Based on the above discussions, the overall VR-OGRK-based precoding is detailed in Algorithm 2. In each iteration, a row is randomly selected based on the selection strategy determined by the relative value of residual, as described in Step 4. The residuals corresponding to the rows whose indices are in \mathcal{X}_k are updated, as shown in Step 8.

C. Convergence Analysis

For the subsequent convergence analysis, we first present the following lemma:

Lemma 2: In the VR-OGRK algorithm, given a system of linear equations $\mathbf{A} \mathbf{x} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{C}^{M \times N}$, $\mathbf{x} \in \mathbb{C}^{N \times 1}$, the vector $(\mathbf{x}^{(t+1)} - \mathbf{x}^{(t)})$ is perpendicular to the vector $(\mathbf{x}^{(t+1)} - \mathbf{x}^*)$, i.e.

$$(\mathbf{x}^{(t+1)} - \mathbf{x}^*)^{\mathbf{H}} (\mathbf{x}^{(t+1)} - \mathbf{x}^{(t)}) = 0, \quad (21)$$

where \mathbf{x}^* denotes the optimal solution of the system.

Proof: According to (10), we have

$$(\mathbf{x}^{(t+1)} - \mathbf{x}^{(t)}) = \frac{b_i - \mathbf{a}_i^{\mathbf{H}} \mathbf{x}^{(t)}}{\|\mathbf{a}_i^{\mathbf{H}}\|_2^2} \mathbf{a}_i. \quad (22)$$

From (22), the vector $(\mathbf{x}^{(t+1)} - \mathbf{x}^{(t)})$ lies in the subspace spanned by \mathbf{a}_i . Thus, to demonstrate the relationship in (21), we need to prove $\mathbf{a}_i^{\mathbf{H}} (\mathbf{x}^{(t+1)} - \mathbf{x}^*) = 0$, as demonstrated below:

$$\begin{aligned} \mathbf{a}_i^{\mathbf{H}} (\mathbf{x}^{(t+1)} - \mathbf{x}^*) &= \mathbf{a}_i^{\mathbf{H}} (\mathbf{x}^{(t)} - \mathbf{x}^*) + \frac{b_i - \mathbf{a}_i^{\mathbf{H}} \mathbf{x}^{(t)}}{\|\mathbf{a}_i^{\mathbf{H}}\|_2^2} \mathbf{a}_i \\ &= \mathbf{a}_i^{\mathbf{H}} \mathbf{x}^{(t)} - \mathbf{a}_i^{\mathbf{H}} \mathbf{x}^* + b_i - \mathbf{a}_i^{\mathbf{H}} \mathbf{x}^{(t)} = 0, \end{aligned} \quad (23)$$

which completes the proof of Lemma 2. ■

Based on Lemma 2, we obtain the following relationship by using the Pythagorean theorem

$$\|\mathbf{x}^{(t+1)} - \mathbf{x}^*\|_2^2 = \|\mathbf{x}^{(t)} - \mathbf{x}^*\|_2^2 - \|\mathbf{x}^{(t+1)} - \mathbf{x}^{(t)}\|_2^2. \quad (24)$$

Based on (24), we derive the following theorem to prove the convergence of our proposed algorithm.

Theorem 3: The VR-OGRK algorithm converges as

$$\|\mathbf{x}^{(t+1)} - \mathbf{x}^*\|_2^2 \leq \eta^t \|\mathbf{x}^{(0)} - \mathbf{x}^*\|_2^2, \quad (25)$$

with global convergence rate of

$$\eta = 1 - \frac{\kappa^2(\mathbf{A})}{\|\mathbf{A}\|_F} < 1, \quad (26)$$

where $\kappa(\mathbf{A}) = \min_{\mathbf{x}-\mathbf{x}^* \in \mathcal{R}(\mathbf{A}^H)} \frac{\|\mathbf{A}(\mathbf{x}-\mathbf{x}^*)\|_2}{\|\mathbf{x}-\mathbf{x}^*\|_2}$.

Proof: First, according to (24), we have

$$\begin{aligned} \mathbb{E}[\|\mathbf{x}^{(t+1)} - \mathbf{x}^*\|_2^2] &= \|\mathbf{x}^{(t)} - \mathbf{x}^*\|_2^2 - \mathbb{E}[\|\mathbf{x}^{(t+1)} - \mathbf{x}^{(t)}\|_2^2] \\ &= \|\mathbf{x}^{(t)} - \mathbf{x}^*\|_2^2 - \sum_{i=1}^M \frac{|\mathbf{r}_i|^2}{\|\mathbf{r}^{(t)}\|_2^2} \frac{|b_i - \mathbf{a}_i^H \mathbf{x}^{(t)}|^2}{\|\mathbf{a}_i\|_2^2} \\ &= \|\mathbf{x}^{(t)} - \mathbf{x}^*\|_2^2 - \frac{1}{\|\mathbf{r}^{(t)}\|_2^2} \sum_{i=1}^M \frac{|\mathbf{r}_i|^4}{\|\mathbf{a}_i\|_2^2} \\ &\leq \|\mathbf{x}^{(t)} - \mathbf{x}^*\|_2^2 - \frac{1}{\|\mathbf{r}^{(t)}\|_2^2} \frac{\sum_{i=1}^M |\mathbf{r}_i|^4}{\sum_{i=1}^M \|\mathbf{a}_i\|_2^2} \\ &= \|\mathbf{x}^{(t)} - \mathbf{x}^*\|_2^2 - \frac{\|\mathbf{r}^{(t)}\|_2^2}{\sum_{i=1}^M \|\mathbf{a}_i\|_2^2} \\ &= \|\mathbf{x}^{(t)} - \mathbf{x}^*\|_2^2 - \frac{\|\mathbf{A}(\mathbf{x}^{(t)} - \mathbf{x}^*)\|_2^2}{\|\mathbf{A}\|_F^2} \\ &\leq (1 - \frac{\kappa^2(\mathbf{A})}{\|\mathbf{A}\|_F^2}) \|\mathbf{x}^{(t)} - \mathbf{x}^*\|_2^2, \end{aligned} \quad (27)$$

where $\kappa(\mathbf{A}) = \min_{\mathbf{x}-\mathbf{x}^* \in \mathcal{R}(\mathbf{A}^H)} \frac{\|\mathbf{A}(\mathbf{x}-\mathbf{x}^*)\|_2}{\|\mathbf{x}-\mathbf{x}^*\|_2}$ denotes the minimum singular value. Define $\eta = \frac{\kappa^2(\mathbf{A})}{\|\mathbf{A}\|_F^2} \leq 1$, which represents the global convergence rate.

Hence, the proof of Theorem 3 is completed. \blacksquare

V. VR-BASED ORTHOGONAL AGGREGATION HYPERPLANE KACZMARZ ALGORITHM

We extend the GRK-based precoding to the aggregation hyperplane Kaczmarz (AHK)-based precoding by exploiting the information from multiple hyperplanes in each iteration. To further reduce computational complexity, we exploit VR information to propose the VR-based orthogonal aggregation hyperplane Kaczmarz (VR-OAHK)-based precoding and analyze its convergence.

A. Aggregation Hyperplane Kaczmarz-based Precoding

As discussed in Section IV, at the t -th iteration of the VR-OGRK algorithm, if the i -th row of matrix \mathbf{A} is selected, the solution of the $(t+1)$ -th iteration is updated as follows

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \frac{b_i - \mathbf{a}_i^H \mathbf{x}^{(t)}}{\|\mathbf{a}_i^H\|_2^2} \mathbf{a}_i = \mathbf{x}^{(t)} + \frac{r_i^{(t)}}{\|\mathbf{a}_i^H\|_2^2} \mathbf{a}_i. \quad (28)$$

Thus, at each iteration, the solution of the next iteration $\mathbf{x}^{(t+1)}$ is updated by projecting the current solution $\mathbf{x}^{(t)}$ into the

selected hyperplane, which is defined by $\mathbf{a}_i^H \mathbf{x} = b_i$. This means that we only use the information from one hyperplane at each iteration.

A natural question arises: *is it possible to utilize information from multiple hyperplanes instead of a single hyperplane at each iteration?* If the information from multiple hyperplanes are jointly utilized, the convergence rate can be further enhanced. Inspired by the surrogate constraint proposed in [40], [41], which is introduced as a heuristic method to solve integer programming problems and graph optimization problems, we next propose an aggregation hyperplane Kaczmarz (AHK) algorithm to jointly use multiple hyperplane. Specifically, AHK algorithm constructs an aggregation hyperplane by linearly combining multiple hyperplanes with predefined weights. The next iteration solution $\mathbf{x}^{(t+1)}$ is then updated by projecting $\mathbf{x}^{(t)}$ onto the aggregation hyperplane, which incorporates information from multiple hyperplanes.

The update criteria of the AHK methods is expressed as

$$\begin{aligned} \mathbf{x}^{(t+1)} &= \mathbf{x}^{(t)} + \frac{(\boldsymbol{\varphi}^{(t)})^H (\mathbf{b} - \mathbf{A} \mathbf{x}^{(t)})}{\|\mathbf{A}^H \boldsymbol{\varphi}^{(t)}\|_2^2} \mathbf{A}^H \boldsymbol{\varphi}^{(t)} \\ &= \mathbf{x}^{(t)} + \frac{(\boldsymbol{\varphi}^{(t)})^H \mathbf{r}^{(t)}}{\|\mathbf{A}^H \boldsymbol{\varphi}^{(t)}\|_2^2} \mathbf{A}^H \boldsymbol{\varphi}^{(t)}, \end{aligned} \quad (29)$$

where $\boldsymbol{\varphi}$ denotes the aggregation coefficient vector. Note that, when $\boldsymbol{\varphi} = \mathbf{e}_i$, where i represents the index of the row which is selected at the t -th iteration, (29) reduces to the update of traditional Kaczmarz algorithm.

In (29), the solution of the next iteration $\mathbf{x}^{(t+1)}$ is updated by projecting the current solution $\mathbf{x}^{(t)}$ onto the aggregation hyperplane which is defined by

$$(\boldsymbol{\varphi}^{(t)})^H \mathbf{A} \mathbf{x} = (\boldsymbol{\varphi}^{(t)})^H \mathbf{b}. \quad (30)$$

The aggregation hyperplane retains more information of the matrix \mathbf{A} and vector \mathbf{b} compared to GRK algorithm. The AHK algorithm linearly combines multiple hyperplanes into an aggregation hyperplane with predefined weights. Thus, at each iteration, more information from the matrix \mathbf{A} and vector \mathbf{b} is utilized to accelerate convergence.

Recall that, in GRK algorithm, the residuals of the current solution on all hyperplanes, i.e., $\mathbf{r}^{(t)}$, are obtained at each iteration, as shown in Step 8 of Algorithm 2. Thus, the aggregation coefficient vector $\boldsymbol{\varphi}^{(t)}$ is defined as

$$[\boldsymbol{\varphi}^{(t)}]_i = \frac{b_i - \mathbf{a}_i^H \mathbf{x}^{(t)}}{\|\mathbf{a}_i^H\|_2^2}, \forall i \in [1, M]. \quad (31)$$

B. VR-based Orthogonal AHK (VR-OAHK)-based Precoding

The AHK-based precoding requires the generation of an aggregation hyperplane, introducing additional complexity of $\mathcal{O}(N_t K)$ at each iteration. While this algorithm can substantially accelerate convergence, it may not result in a significant reduction in overall complexity. To address this issue, we propose VR-based orthogonal aggregation hyperplane Kaczmarz (VR-OAHK) algorithm. Specifically, by fully exploiting the VR features and further accelerating the convergence, we first divide K users into two sets: the orthogonal set \mathcal{F} and the non-orthogonal set \mathcal{V} . The orthogonal set \mathcal{F} consists of users whose VRs do not overlap, i.e. $|\mathcal{M}_i \cap \mathcal{M}_j| > 0, \forall i, j \in \mathcal{F}$,

Algorithm 3 VR-OAHK-based precoding algorithm

- 1: **Input:** \mathbf{H} , \mathbf{H}_s , N_t , K , ξ , \mathcal{F} , \mathcal{V} , Γ_k , \mathcal{Q}_s , threshold ϵ .
 - 2: Repeat Algorithm 1 Steps 2-4;
 - 3: Initialize $\mathbf{r}_k^{\text{or}} = \mathbf{0}_K$, $\mathbf{r}_k^{\text{nor}} = \mathbf{0}_K$, $\boldsymbol{\varphi}_k^{\text{or}} = \mathbf{0}_K$, $\boldsymbol{\varphi}_k^{\text{nor}} = \mathbf{0}_K$, $\forall k \in [1, K]$;
 - 4: Construct $\mathbf{h}_k = [\mathbf{h}_k]_{\Gamma_k}$;
 - 5: **while** NMSE $\geq \epsilon$ **do**
 - 6: Run the VR-OAHK algorithm K times:
 - 7: For orthogonal user group \mathcal{F} :
 - 8: $[\mathbf{r}_k^{\text{or}(t)}]_{\mathcal{F}} = [\mathbf{e}_k]_{\mathcal{F}} - \mathbf{h}_{\mathcal{F}}^H \mathbf{m}_k^{(t)} - \xi [\mathbf{q}_k^{(t)}]_{\mathcal{F}}$;
 - 9: $[\boldsymbol{\varphi}_k^{\text{or}(t)}]_i = \frac{[\mathbf{r}_k^{\text{or}(t)}]_i}{\|\mathbf{h}_i\|_2 + \xi}$, $\forall i \in \mathcal{F}$;
 - 10: $\mathbf{m}_k^{(t+\frac{1}{2})} = \mathbf{m}_k^{(t)} + \sum_{i \in \mathcal{F}} [\boldsymbol{\varphi}_k^{\text{or}(t)}]_i \mathbf{h}_i$;
 - 11: $\mathbf{q}_k^{(t+\frac{1}{2})} = \mathbf{q}_k^{(t)} + \boldsymbol{\varphi}_k^{\text{or}(t)}$;
 - 12: For non-orthogonal user group \mathcal{V} :
 - 13: $[\mathbf{r}_k^{\text{nor}(t)}]_{\mathcal{V}} = [\mathbf{e}_k]_{\mathcal{V}} - \tilde{\mathbf{h}}_{\mathcal{V}}^H \mathbf{m}_k^{(t+\frac{1}{2})} - \xi [\mathbf{q}_k^{(t+\frac{1}{2})}]_{\mathcal{V}}$;
 - 14: $[\boldsymbol{\varphi}_k^{\text{nor}(t)}]_i = \frac{[\mathbf{r}_k^{\text{nor}(t)}]_i}{\|\mathbf{h}_i\|_2 + \xi}$, $\forall i \in \mathcal{V}$;
 - 15: $\gamma_k^{\text{nor}(t)} = \frac{(\boldsymbol{\varphi}_k^{\text{nor}(t)})^H \mathbf{r}_k^{\text{nor}(t)}}{\|\mathbf{H} \boldsymbol{\varphi}_k^{\text{nor}(t)}\|_2^2 + \xi \|\boldsymbol{\varphi}_k^{\text{nor}(t)}\|_2^2}$;
 - 16: $\mathbf{m}_k^{(t+1)} = \mathbf{m}_k^{(t+\frac{1}{2})} + \gamma_k^{\text{nor}(t)} \mathbf{H} \boldsymbol{\varphi}_k^{\text{nor}(t)}$;
 - 17: $\mathbf{q}_k^{(t+1)} = \mathbf{q}_k^{(t+\frac{1}{2})} + \gamma_k^{\text{nor}(t)} \boldsymbol{\varphi}_k^{\text{nor}(t)}$;
 - 18: $t = t + 1$;
 - 19: **end while**
 - 20: $\mathbf{V} = [\mathbf{q}_1, \dots, \mathbf{q}_K]$, Calculate \mathbf{F} as (20);
 - 21: **Output:** \mathbf{F} , $T = t$.
-

whereas the opposite holds true for the non-orthogonal set \mathcal{V} . In the following, we prove some promising properties of the proposed AHK algorithm on the orthogonal set, such as faster convergence and parallelization features. In each iteration, we first apply AHK algorithm using information from the hyperplanes corresponding to \mathcal{F} . Subsequently, we apply AHK algorithm using information from the hyperplanes corresponding to \mathcal{V} .

To utilize the potential of exploiting spatial non-stationarity in simplifying algorithms, we derive the following theorem.

Theorem 4: Consider using the AHK algorithm to solve a system of linear equations $\mathbf{A}\mathbf{x} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{C}^{M \times N}$, $\mathbf{x} \in \mathbb{C}^{N \times 1}$. We assume that there exists a set of mutually orthogonal hyperplanes in the linear system, denoted as \mathcal{F} . At the t -th iteration, if the aggregation hyperplane is generated by the hyperplanes in \mathcal{F} , the updated solution $\mathbf{x}^{(t+1)}$ will lie in the solution space spanned by the hyperplanes in \mathcal{F} , i.e.,

$$\mathbf{A}_{[\mathcal{F},:]}\mathbf{x}^{(t+1)} = \mathbf{b}_{\mathcal{F}}. \quad (32)$$

Proof: See Appendix C. \blacksquare

According to Theorem 4, if the orthogonal set \mathcal{F} can be established, only one iteration is required to converge to the solution space spanned by the hyperplanes in \mathcal{F} , which further accelerates convergence. Furthermore, the orthogonality among different users in the set \mathcal{F} enables parallelization of the algorithm, as detailed in the following proposition.

Proposition 1: A system of linear equations $\mathbf{A}\mathbf{x} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{C}^{M \times N}$, $\mathbf{x} \in \mathbb{C}^{N \times 1}$, is considered to be solved by the AHK algorithm. We assume that there exists a set of mutually orthogonal hyperplanes in the linear system, denoted as \mathcal{F} . If

the aggregation hyperplane is generated by the hyperplanes in \mathcal{F} , the solution of the next iteration is updated as

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \sum_{i \in \mathcal{F}} [\boldsymbol{\varphi}^{(t)}]_i \mathbf{a}_i. \quad (33)$$

Proof: See Appendix D. \blacksquare

According to (33), the vector $[\boldsymbol{\varphi}^{(t)}]_i \mathbf{a}_i$, $\forall i \in \mathcal{F}$, can be independently calculated for each hyperplane and then summed to update the next iteration solution $\mathbf{x}^{(t+1)}$. When updating the solution based on the orthogonal set \mathcal{F} , the computation of the relevant parameters can be divided into multiple subtasks. Each subtask is then executed in parallel on the multicore CPU at the BS and therefore greatly reduces the running time.

Next, we focus on how to accurately construct the orthogonal set \mathcal{F} . Specifically, we propose a graph theory-based orthogonal set generation algorithm. We first construct an undirected graph $G = (V, E)$, where $V = \{v_1, \dots, v_K\}$ and $E = \{(v_i, v_j) \mid v_i, v_j \in V\}$ denote the set of vertex corresponding with K users and the set of edges, respectively. If $(v_i, v_j) \in E$, then there exists an edge between the vertices v_i and v_j . By utilizing the VR information of different users, then we establish the edge between two users based on their VR overlap situations. Specifically, if $|\mathcal{M}_i \cap \mathcal{M}_j| > 0$, then we establish an edge $(v_i, v_j) \in E$ in the graph.

After constructing the VR-based undirected graph, the orthogonal set \mathcal{F} is regarded as the independent set in the undirected graph, which ensures that there are no edges between any two vertices in this set. To enlarge the impact of the orthogonal set on improving algorithm performance, our goal is to find as many as possible vertices in the VR-based graph $G = (V, E)$ to form a maximum independent set. Thus, we formulate following maximum independent set problem

$$\max |\mathcal{F}| \quad (34a)$$

$$\text{s.t. } \mathcal{F} \subseteq V, \quad (34b)$$

$$(v_i, v_j) \notin E, \forall v_i, v_j \in \mathcal{F}. \quad (34c)$$

Problem (34) is a classic problem in graph theory, which can be effectively addressed with a low-complexity algorithm with linear complexity $\mathcal{O}(K)$ [42]. After constructing the orthogonal set \mathcal{F} , the remaining users can naturally be grouped into \mathcal{V} . Thus, in the VR-OAHK algorithm, the update criteria at each iteration is

$$\mathbf{x}^{(t+\frac{1}{2})} = \mathbf{x}^{(t)} + \frac{(\boldsymbol{\varphi}^{\text{or}(t)})^H \mathbf{r}^{\text{or}(t)}}{\|\mathbf{A}^H \boldsymbol{\varphi}^{\text{or}(t)}\|_2^2} \mathbf{A}^H \boldsymbol{\varphi}^{\text{or}(t)}, \quad (35)$$

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t+\frac{1}{2})} + \frac{(\boldsymbol{\varphi}^{\text{nor}(t)})^H \mathbf{r}^{\text{nor}(t)}}{\|\mathbf{A}^H \boldsymbol{\varphi}^{\text{nor}(t)}\|_2^2} \mathbf{A}^H \boldsymbol{\varphi}^{\text{nor}(t)}, \quad (36)$$

where the aggregation coefficient and residual of the orthogonal set and non-orthogonal set are respectively defined as

$$[\mathbf{r}^{\text{or}(t)}]_{\mathcal{F}} = \mathbf{b}_{\mathcal{F}} - \mathbf{A}_{[\mathcal{F},:]}\mathbf{x}^{(t)}, \quad (37)$$

$$[\boldsymbol{\varphi}^{\text{or}(t)}]_{\mathcal{F}} = \frac{[\mathbf{r}^{\text{or}(t)}]_i}{\|\mathbf{a}_i^H\|_2^2}, \forall i \in \mathcal{F}, \quad (38)$$

$$[\mathbf{r}^{\text{nor}(t)}]_{\mathcal{V}} = \mathbf{b}_{\mathcal{V}} - \mathbf{A}_{[\mathcal{V},:]}\mathbf{x}^{(t+\frac{1}{2})}, \quad (39)$$

$$[\boldsymbol{\varphi}^{\text{nor}(t)}]_{\mathcal{V}} = \frac{[\mathbf{r}^{\text{nor}(t)}]_i}{\|\mathbf{a}_i^H\|_2^2}, \forall i \in \mathcal{V}. \quad (40)$$

TABLE I: Computational Complexity Comparison

Scheme	Computational Complexity [FLOPS]
RZF	$8K^3 + 9K^2 + 12N_t K^2 - 3K$
URK	$8N_t K^2 + 4N_t K + T^{\text{URK}}(16N_t - 4)$
SWOR-ERK	$8N_t K^2 + 4N_t K + K - 1 + T^{\text{ERK}}(16N_t + K + 8)$
GK	$8N_t K^2 + 4N_t K - K + T^{\text{GK}}(8N_t(K + 1) + K - 5)$
VR-OGRK	$8N_t K(\mathcal{Q} + \frac{1}{2}) - K + T^{\text{OGRK}}(8 \mathcal{M} (\mathcal{X} + 1) + K - 5)$
VR-OAHK	$8N_t K(\mathcal{Q} + \frac{1}{2}) + T^{\text{OAHK}}(8 \mathcal{M} (2\mathcal{V} + \mathcal{F} + 2) + 24 \mathcal{V} + 14N_t + 5)$

Based on the above discussions, the detailed procedure is outlined in Algorithm 3.

C. Convergence Analysis

Lemma 3: In the VR-OAHK algorithm, given a system of linear equations $\mathbf{A}\mathbf{x} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{C}^{M \times N}$, $\mathbf{x} \in \mathbb{C}^{N \times 1}$, in each iteration, we have the following relationship:

$$(\mathbf{x}^{(t+\frac{1}{2})} - \mathbf{x}^*)^H (\mathbf{x}^{(t+1)} - \mathbf{x}^{(t+\frac{1}{2})}) = 0, \quad (41)$$

$$(\mathbf{x}^{(t)} - \mathbf{x}^*)^H (\mathbf{x}^{(t+\frac{1}{2})} - \mathbf{x}^{(t)}) = 0, \quad (42)$$

where \mathbf{x}^* denotes the optimal solution of the system.

Proof: According to (35) and (36), we have

$$\mathbf{x}^{(t+1)} - \mathbf{x}^{(t+\frac{1}{2})} = \frac{(\boldsymbol{\varphi}^{\text{nor}(t)})^H \mathbf{r}^{\text{nor}(t)}}{\|\mathbf{A}^H \boldsymbol{\varphi}^{\text{nor}(t)}\|_2^2} \mathbf{A}^H \boldsymbol{\varphi}^{\text{nor}(t)}, \quad (43)$$

$$\mathbf{x}^{(t+\frac{1}{2})} - \mathbf{x}^{(t)} = \frac{(\boldsymbol{\varphi}^{\text{or}(t)})^H \mathbf{r}^{\text{or}(t)}}{\|\mathbf{A}^H \boldsymbol{\varphi}^{\text{or}(t)}\|_2^2} \mathbf{A}^H \boldsymbol{\varphi}^{\text{or}(t)}. \quad (44)$$

Thus, to demonstrate (41), we need to show that $\mathbf{A}^H \boldsymbol{\varphi}^{\text{nor}(t)} (\mathbf{x}^{(t+\frac{1}{2})} - \mathbf{x}^{(*)}) = 0$, which is proved as

$$\begin{aligned} & (\mathbf{A}^H \boldsymbol{\varphi}^{\text{nor}(t)})^H (\mathbf{x}^{(t+\frac{1}{2})} - \mathbf{x}^{(*)}) \\ &= (\boldsymbol{\varphi}^{\text{nor}(t)})^H \mathbf{A} (\mathbf{x}^{(t)} - \mathbf{x}^* + \frac{(\boldsymbol{\varphi}^{\text{nor}(t)})^H \mathbf{r}^{(t)}}{\|\mathbf{A}^H \boldsymbol{\varphi}^{\text{nor}(t)}\|_2^2} \mathbf{A}^H \boldsymbol{\varphi}^{\text{nor}(t)}) \\ &= (\boldsymbol{\varphi}^{\text{nor}(t)})^H \mathbf{A} \mathbf{x}^{(t)} - (\boldsymbol{\varphi}^{\text{nor}(t)})^H \mathbf{A} \mathbf{x}^* + (\boldsymbol{\varphi}^{\text{nor}(t)})^H \mathbf{r}^{(t)} \\ &= (\boldsymbol{\varphi}^{\text{nor}(t)})^H (\mathbf{A} \mathbf{x}^{(t)} - \mathbf{b}) + (\boldsymbol{\varphi}^{\text{nor}(t)})^H \mathbf{r}^{(t)} = 0. \end{aligned} \quad (45)$$

Following similar steps with (45), (44) can be proved, completing the proof. ■

Based on Lemma 3, we obtain the following relationship by using the Pythagorean theorem

$$\|\mathbf{x}^{(t+1)} - \mathbf{x}^*\|_2^2 = \|\mathbf{x}^{(t+\frac{1}{2})} - \mathbf{x}^*\|_2^2 - \|\mathbf{x}^{(t+1)} - \mathbf{x}^{(t+\frac{1}{2})}\|_2^2, \quad (46)$$

$$\|\mathbf{x}^{(t+\frac{1}{2})} - \mathbf{x}^*\|_2^2 = \|\mathbf{x}^{(t)} - \mathbf{x}^*\|_2^2 - \|\mathbf{x}^{(t+\frac{1}{2})} - \mathbf{x}^{(t)}\|_2^2. \quad (47)$$

Theorem 5: The VR-OAHK algorithm converges as

$$\|\mathbf{x}^{(t+1)} - \mathbf{x}^*\|_2^2 \leq \alpha^{2t} \|\mathbf{x}^{(0)} - \mathbf{x}^*\|_2^2, \quad (47)$$

with global convergence rate

$$\alpha = (1 - \frac{\boldsymbol{\vartheta} \mathbf{P} \boldsymbol{\vartheta}^H}{\boldsymbol{\vartheta} \boldsymbol{\vartheta}^H}) < 1, \quad (48)$$

where $\boldsymbol{\vartheta} = \mathbf{A}^H \boldsymbol{\Phi}^T \mathbf{A} \mathbf{e}^{(t+\frac{1}{2})}$, $\mathbf{P} = \frac{\mathbf{e}^{(t+\frac{1}{2})} (\mathbf{e}^{(t+\frac{1}{2})})^H}{(\mathbf{e}^{(t+\frac{1}{2})})^H \mathbf{e}^{(t+\frac{1}{2})}}$, $\mathbf{e}^{(t+\frac{1}{2})} = \mathbf{x}^{(t+\frac{1}{2})} - \mathbf{x}^*$.

Proof: To simplify the analysis, we define $\mathbf{x}^{(t)} - \mathbf{x}^* = \mathbf{e}^{(t)}$. Then, according to (46), we have the following relationship

$$\begin{aligned} \|\mathbf{e}^{(t+1)}\|_2^2 &= \|\mathbf{e}^{(t+\frac{1}{2})}\|_2^2 - \|\mathbf{x}^{(t+1)} - \mathbf{x}^{(t+\frac{1}{2})}\|_2^2 \\ &= \|\mathbf{e}^{(t+\frac{1}{2})}\|_2^2 - \frac{(\mathbf{e}^{(t+\frac{1}{2})})^H \mathbf{A}^H \boldsymbol{\varphi}^{\text{nor}(t)} (\boldsymbol{\varphi}^{\text{nor}(t)})^H \mathbf{A} \mathbf{e}^{(t+\frac{1}{2})}}{\|\mathbf{A}^H \boldsymbol{\varphi}^{\text{nor}(t)}\|_2^2}. \end{aligned} \quad (49)$$

Define $\boldsymbol{\Phi} = \text{diag}(\frac{1}{\|\mathbf{a}_1^H\|_2^2}, \dots, \frac{1}{\|\mathbf{a}_M^H\|_2^2}) \in \mathbb{R}^{M \times M}$. Then, (40) can be rewritten as

$$\boldsymbol{\varphi}^{\text{nor}(t)} = -\boldsymbol{\Phi} \mathbf{A} \mathbf{e}^{(t+\frac{1}{2})}. \quad (50)$$

By substituting (50) into (49), $\|\mathbf{e}^{(t+1)}\|_2^2$ is reformulated as (51), shown at the bottom of this page, where the parameters are defined as

$$\boldsymbol{\vartheta} = \mathbf{A}^H \boldsymbol{\Phi}^T \mathbf{A} \mathbf{e}^{(t+\frac{1}{2})}, \quad (52)$$

$$\mathbf{P} = \frac{\mathbf{e}^{(t+\frac{1}{2})} (\mathbf{e}^{(t+\frac{1}{2})})^H}{(\mathbf{e}^{(t+\frac{1}{2})})^H \mathbf{e}^{(t+\frac{1}{2})}}, \quad (53)$$

where \mathbf{P} is a rank-one Hermitian matrix, the eigenvalue of \mathbf{P} is either 0 or 1. Then, according to Rayleigh quotient theorem, we can obtain the following relationship

$$0 < \frac{\boldsymbol{\vartheta} \mathbf{P} \boldsymbol{\vartheta}^H}{\boldsymbol{\vartheta} \boldsymbol{\vartheta}^H} \leq 1, \quad (54)$$

and the convergence rate $\alpha = (1 - \frac{\boldsymbol{\vartheta} \mathbf{P} \boldsymbol{\vartheta}^H}{\boldsymbol{\vartheta} \boldsymbol{\vartheta}^H}) \in [0, 1)$. The convergence analysis of orthogonal set is similar to the non-stationary set. Therefore, we have

$$\begin{aligned} \|\mathbf{x}^{(t+1)} - \mathbf{x}^*\|_2^2 &\leq \alpha \|\mathbf{x}^{(t+\frac{1}{2})} - \mathbf{x}^*\|_2^2 \\ &\leq \alpha^2 \|\mathbf{x}^{(t)} - \mathbf{x}^*\|_2^2 \leq \alpha^{2t} \|\mathbf{x}^{(0)} - \mathbf{x}^*\|_2^2, \end{aligned} \quad (55)$$

completing the proof. ■

$$\begin{aligned} \|\mathbf{e}^{(t+1)}\|_2^2 &= \|\mathbf{e}^{(t+\frac{1}{2})}\|_2^2 - \frac{(\mathbf{e}^{(t+\frac{1}{2})})^H \mathbf{A}^H \boldsymbol{\Phi} \mathbf{A} \mathbf{e}^{(t+\frac{1}{2})} \mathbf{A} \mathbf{e}^{(t+\frac{1}{2})} (\mathbf{e}^{(t+\frac{1}{2})})^H \mathbf{A}^H \boldsymbol{\Phi} \mathbf{A} \mathbf{e}^{(t+\frac{1}{2})} \mathbf{A} \mathbf{e}^{(t+\frac{1}{2})}}{(\mathbf{e}^{(t+\frac{1}{2})})^H \mathbf{A}^H \boldsymbol{\Phi}^T \mathbf{A} \mathbf{A}^H \boldsymbol{\Phi}^T \mathbf{A} \mathbf{e}^{(t+\frac{1}{2})}} \\ &= (1 - \frac{((\mathbf{e}^{(t+\frac{1}{2})})^H \mathbf{A}^H \boldsymbol{\Phi} \mathbf{A} \mathbf{e}^{(t+\frac{1}{2})} \mathbf{A} \mathbf{e}^{(t+\frac{1}{2})})^2}{(\mathbf{e}^{(t+\frac{1}{2})})^H (\mathbf{A}^H \boldsymbol{\Phi}^T \mathbf{A})^2 \mathbf{e}^{(t+\frac{1}{2})} (\mathbf{e}^{(t+\frac{1}{2})})^H \mathbf{e}^{(t+\frac{1}{2})}}) \|\mathbf{e}^{(t+\frac{1}{2})}\|_2^2 \\ &= (1 - \frac{\boldsymbol{\vartheta} \mathbf{P} \boldsymbol{\vartheta}^H}{\boldsymbol{\vartheta} \boldsymbol{\vartheta}^H}) \|\mathbf{e}^{(t+\frac{1}{2})}\|_2^2, \end{aligned} \quad (51)$$

VI. COMPLEXITY ANALYSIS

The detailed computational complexity, expressed in floating point operations (FLOPS) [43], is presented in Table I. In this table, T^{URK} , T^{ERK} , T^{GK} , T^{OGRK} , T^{OAHK} represent the iteration numbers for the uniform criterion-based RK algorithm (URK) [21], the energy criterion-based RK algorithm with sampling without replacement technology (SWOR-ERK) [34], the GK algorithm [44], the VR-OGRK algorithm, and the VR-OAHK algorithm, respectively. For the sake of analysis, we assume here, $|\Gamma_k| = |\Gamma|$, $|\mathcal{X}_k| = |\mathcal{X}|, \forall k$, and but the true value of $|\Gamma_k|$, $|\mathcal{X}_k|$ will be used in the simulation section. As shown in Table I, for the proposed VR-OGRK algorithm in Algorithm 2, the computational complexity of calculating the precoding matrix \mathbf{F} reduces from $\mathcal{O}(8N_t K^2)$ to $\mathcal{O}(8N_t K |\mathcal{Q}|)$ and the computational complexity of Step 8 reduces from $\mathcal{O}(8N_t K)$ to $\mathcal{O}(8|\Gamma||\mathcal{X}|)$ at each iteration. Therefore, our algorithm can work effectively in ELAA-supported hotspot scenarios. Furthermore, the main advantage of the two proposed algorithms is the significant reduction in the number of iterations compared to the traditional Kaczmarz-type algorithms, which will be shown in the simulation section.

VII. SIMULATION RESULTS

In this section, numerical results are presented to demonstrate the effectiveness of the proposed VR-OGRK algorithm and the VR-OAHK algorithm compared to the traditional Kaczmarz-type algorithm.

A. Simulation Configuration

Without loss of generality, the simulation parameters are assumed to be $N_t = 2000$, $f = 100$ GHz, $S = 20$, $\epsilon = 10^{-6}$, $L_k = 5$. There are $K = 30$ users randomly distributed within a range of 1 m to 50 m from the ELAA. $\text{SNR}_k = \frac{\rho \|\mathbf{h}_k\|_2^2}{\sigma^2}$ is defined as the received SNR for user k . For simplicity, the SNRs of all users are assumed to be equal due to power control, i.e., $\text{SNR} = 0$ dB. From [45] and [46], the VR set \mathcal{M}_k can be modeled as a Bernoulli distribution. Thus, the visibility of the s -th subarray for the k -th user is modeled as

$$\begin{cases} s \in \mathcal{M}_k, \text{ with probability } p, \\ s \notin \mathcal{M}_k, \text{ with probability } 1 - p, \end{cases} \quad (56)$$

where p denotes the VR generation probability. From [45] and [46], we assume $p = 0.35$. Furthermore, the impact of this parameter on the proposed algorithm will be discussed in the simulations.

B. Algorithm Performance Comparison

Fig. 2 compares the convergence behaviors of different algorithms in terms of NMSE ($\text{NMSE} = \frac{\|\mathbf{F}^{\text{RZF}} - \mathbf{F}\|_F}{\|\mathbf{F}^{\text{RZF}}\|_F}$)¹. As shown in Fig. 2, the proposed VR-OGRK algorithm exhibits a faster convergence rate than both the URK and SWOR-ERK algorithms, while its convergence behavior is similar to that of the GK algorithm. Furthermore, to achieve the NMSE of 10^{-6} , the proposed VR-OGRK and VR-OAHK algorithms

¹The NMSE value approaches zero, indicating that the sum rate realized by this scheme is nearly equivalent to that of the RZF scheme.

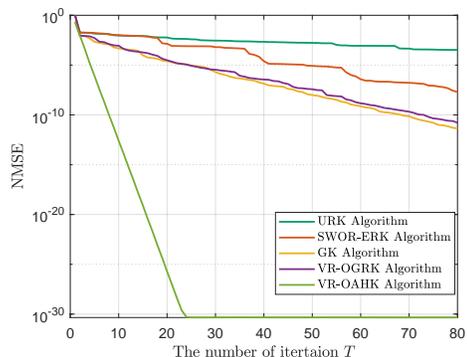


Fig. 2. Convergence behaviors of different algorithms.

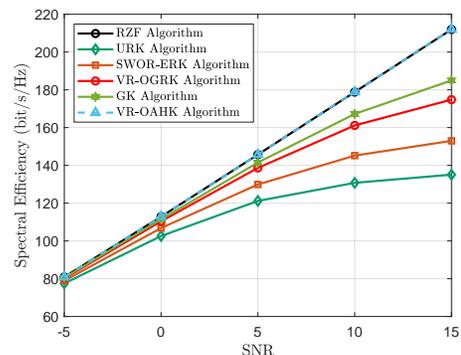


Fig. 3. Spectral efficiency versus SNR with $T = 15$.

require only 27 and 5 iterations, respectively, representing a significant reduction compared to the SWOR-ERK and URK algorithms. For the VR-OGRK algorithm, the improvement of the convergence speed is attributed to its selection strategy, which ensures that the hyperplane with a large residual error is chosen at each iteration. Building on this, the proposed VR-OAHK algorithm further leverages information from multiple hyperplanes rather than a single hyperplane at each iteration, resulting in a faster convergence rate.

In Fig. 3, we investigate the spectral efficiency versus SNR. It is important to note that we assume the iteration number of $T = T^{\text{URK}} = T^{\text{ERK}} = T^{\text{GK}} = T^{\text{OGRK}} = T^{\text{OAHK}} = 15$. As observed from Fig. 3, the spectral efficiency of the proposed VR-OAHK algorithm is nearly equal to that of the RZF algorithm after a small number of iterations. While the spectral efficiency of the proposed VR-OGRK algorithm is slightly lower than that of the VR-OAHK algorithm, it remains significantly higher than that of the SWOR-ERK and URK algorithms. This is because the two proposed algorithms have faster convergence rate by leveraging the VR features. Furthermore, as SNR increases, traditional Kaczmarz-type algorithms do not yield satisfactory performance. This is because communication scenarios with high SNR require higher accuracy in matrix inversion, and thus the traditional Kaczmarz algorithm requires an extra number of iterations to approximate the matrix inversion. Thus, the slower convergence of the traditional Kaczmarz-type algorithm leads to reduced spectral efficiency at a fixed number of iterations. Moreover, when the number of iterations is unrestricted, all algorithms can achieve similar performance to RZF. However, the slow convergence rate of the traditional Kaczmarz algorithm results in additional

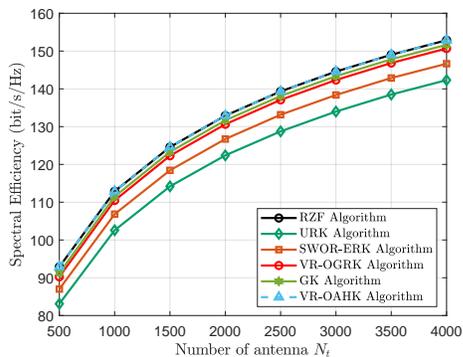


Fig. 4. Spectral efficiency versus N_t with $T = 15$.

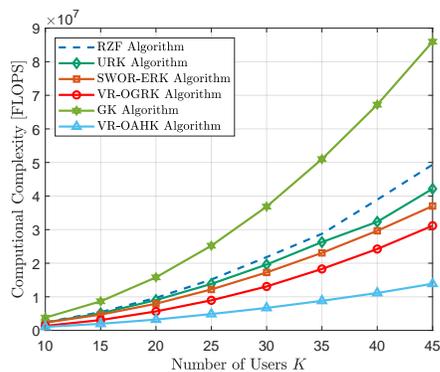


Fig. 5. Computational complexity versus K .

computational overhead, which will be discussed later.

Fig. 4 illustrates the relationship between the spectral efficiency and the number of antenna N_t with a fixed iteration number of $T = 15$ across different algorithms. As illustrated in Fig. 4, the proposed VR-OGRK algorithm and VR-OAHK algorithm can achieve the spectral efficiency comparable to the RZF algorithm. Furthermore, as the number of antennas N_t increases, both proposed algorithms continue to maintain similar performance to the RZF algorithm. This is because both the proposed algorithms converge faster than the SWOR-ERK and URK algorithm. Consequently, with a fixed number of iterations, the two proposed algorithms exhibit a lower NMSE, resulting in higher spectral efficiency. Therefore, even with a larger number of antennas, the two proposed algorithms can achieve the spectral efficiencies close to that of RZF.

In Fig. 5, we compare the computational complexity of different algorithms in terms of K . As observed in Fig. 5, the proposed VR-OGRK algorithm and the proposed VR-OAHK algorithm have a lower computational complexity than the traditional Kaczmarz-type algorithms. The complexity of the GK algorithm exceeds that of the RZF scheme, mainly due to the additional residual computation during each iteration. It can be seen that the computational complexity of the proposed VR-OGRK algorithm increases at a slower rate as K rises. This is because the complexity with respect to the number of users in the proposed VR-OGRK algorithm scales as $\mathcal{O}(8N_tK|Q|)$, while for the URK and SWOR-ERK algorithms, it is on the order of $\mathcal{O}(8N_tK^2)$. Since the growth of $|Q|$ is less than that of K , the proposed VR-OGRK algorithm maintains a low computational complexity even with

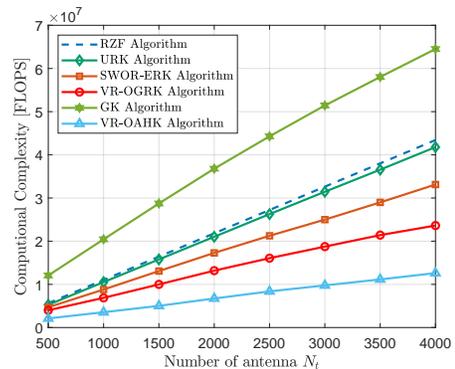


Fig. 6. Computational complexity versus N_t .

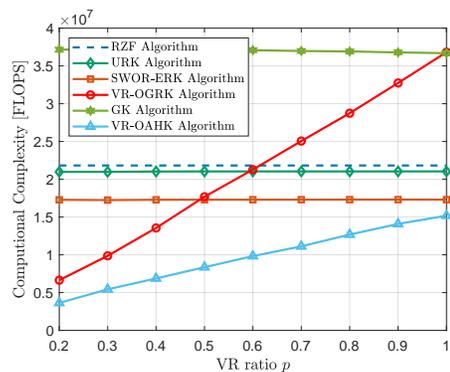


Fig. 7. Computational complexity versus p .

a large total number of users. Additionally, the proposed VR-OAHK algorithm further reduces the computational cost by decreasing the required number of iterations using information from multiple hyperplanes in each iteration.

Fig. 6 examines the computational complexity of different algorithms in relation to N_t . It can be observed that the proposed VR-OGRK and VR-OAHK algorithms substantially reduce the computational complexity compared to the other algorithms. Moreover, as N_t increases, the proposed algorithms maintain low computational complexities. This is because the computational complexity of N_t in the VR-OGRK algorithm is $\mathcal{O}(T(8|\Gamma|(|\mathcal{X}| + 1)))$, whereas for the URK and SWOR-ERK algorithms, it is $\mathcal{O}(T(16N_t))$. Due to the spatial non-stationarity effect, the increase in $|\Gamma|$ is less than that in N_t . Furthermore, by utilizing information from multiple hyperplanes in each iteration, the proposed VR-OAHK algorithm achieves a lower computational complexity than the VR-OGRK algorithm. Thus, as shown in Fig. 3-6, the two proposed algorithms can achieve satisfactory performance with low complexity in ELAA systems.

In Fig. 7, we explore the relationship between the computational complexity and the VR generation probability p . As the value of p increases, the computational complexity of both the proposed VR-OGRK algorithm and the proposed VR-OAHK algorithm also increases. A larger p indicates that the spatial non-stationary effect of the communication system is less pronounced. Therefore, the computational complexity reduction achieved by the proposed VR-OGRK algorithm by utilizing the spatial non-stationary effect will be weakened, and will eventually degenerate to the performance of standard

GK algorithm. Fortunately, owing to the fast convergence rate, the proposed VR-OAHK algorithm substantially reduces the computational complexity even at high values of p compared to the SWOR-ERK and URK algorithm. This is because the proposed VR-OAHK algorithm utilizes information from multiple hyperplanes, in contrast to the VR-OGRK algorithm and traditional Kaczmarz-type algorithms, which rely on a single hyperplane. Therefore, the proposed VR-OAHK algorithm possesses promising potentials to enable future ELAA systems even in case of weak spatial non-stationary effects.

VIII. CONCLUSION

In this paper, we proposed a low-complexity iterative precoding design for near-field multiuser systems. Firstly, we reformulated the precoding design problem as a least square problem, which can be addressed by utilizing the iterative Kaczmarz algorithm. Next, to address the limitations of existing algorithms and leverage the properties of spatial non-stationarity, we proposed a VR-OGRK algorithm to enhance convergence rate and significantly reduce computational complexity. We also proposed a novel VR-OAHK algorithm based on graph theory and the idea of aggregation hyperplane to further reduce computational complexity. Furthermore, the two proposed algorithms were shown to exhibit global exponential convergence. Simulation results demonstrated the effectiveness of the two proposed low-complexity algorithms.

APPENDIX A

Consider the following optimization problem

$$\mathbf{v} = \arg \min_{\mathbf{v}} f(\mathbf{v}) = \|\mathbf{G}\mathbf{v} - \mathbf{g}\|_2^2, \quad (57)$$

where $\mathbf{G} = [\mathbf{H}; \sqrt{\xi}\mathbf{I}_K] \in \mathbb{C}^{(N_t+K) \times K}$, $\mathbf{g} = [\mathbf{0}_{N_t}; \frac{\mathbf{s}}{\sqrt{\xi}}] \in \mathbb{C}^{(N_t+K)}$. Since $f(\mathbf{v})$ is convex with \mathbf{v} , by checking the optimality condition, the optimal solution of \mathbf{v} can be expressed as

$$\begin{aligned} \frac{\partial f(\mathbf{v})}{\partial \mathbf{v}} &= 2\mathbf{G}^H\mathbf{G}\mathbf{v} - 2\mathbf{G}^H\mathbf{g} = \mathbf{0} \\ \Rightarrow \mathbf{v} &= (\mathbf{H}^H\mathbf{H} + \xi\mathbf{I}_K)^{-1}\mathbf{s}. \end{aligned} \quad (58)$$

Hence, the proof of Theorem 1 is completed.

APPENDIX B

At the t -th iteration, we assume that the i -th row of \mathbf{A} , i.e., \mathbf{a}_i^H is selected, and then the solution for the next iteration \mathbf{x}^{t+1} is updated as follows

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \frac{b_i - \mathbf{a}_i^H \mathbf{x}^{(t)}}{\|\mathbf{a}_i^H\|_2^2} \mathbf{a}_i. \quad (59)$$

Next, if the j -th row of \mathbf{A} , i.e., \mathbf{a}_j^H which is orthogonal to \mathbf{a}_i^H , i.e., $\mathbf{a}_j^H \mathbf{a}_i = 0$, the residual for \mathbf{a}_j^H at the $t+1$ iteration is computed as follows

$$\begin{aligned} r_j^{(t+1)} &= b_j - \mathbf{a}_j^H \mathbf{x}^{(t+1)} \\ &\stackrel{(b)}{=} b_j - \mathbf{a}_j^H (\mathbf{x}^{(t)} + \frac{b_i - \mathbf{a}_i^H \mathbf{x}^{(t)}}{\|\mathbf{a}_i^H\|_2^2} \mathbf{a}_i) \\ &\stackrel{(c)}{=} b_j - \mathbf{a}_j^H \mathbf{x}^{(t)} = r_j^{(t)}, \end{aligned} \quad (60)$$

where (b) denotes the update criteria (59), (c) uses the orthogonal property between \mathbf{a}_i and \mathbf{a}_j . Clearly, after selecting the i -th row of \mathbf{A} , only the residuals corresponding to the rows that are not orthogonal to the i -th row are affected. Hence, the proof of Theorem 2 is completed.

APPENDIX C

We assume that there exists a set of mutually orthogonal hyperplanes in the linear system, denoted as \mathcal{F} , i.e., $\mathbf{a}_i^H \mathbf{a}_j = 0, \forall i, j \in \mathcal{F}, i \neq j$. At the t -th iteration, the aggregation hyperplane is generated by the hyperplanes in \mathcal{F} , i.e.,

$$[\varphi^{(t)}]_i = \begin{cases} \frac{b_i - \mathbf{a}_i^H \mathbf{x}^{(t)}}{\|\mathbf{a}_i^H\|_2^2}, & i \in \mathcal{F}, \\ 0, & i \notin \mathcal{F}. \end{cases} \quad (61)$$

Thus, the next iteration solution $\mathbf{x}^{(t+1)}$ is calculated as

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \frac{(\varphi^{(t)})^H \mathbf{r}^{(t)}}{\|\mathbf{A}^H \varphi^{(t)}\|_2^2} \mathbf{A}^H \varphi^{(t)}. \quad (62)$$

To demonstrate the relationship $\mathbf{A}_{[\mathcal{F},:]}\mathbf{x}^{(t+1)} = \mathbf{b}_{\mathcal{F}}$, we need to prove that $\mathbf{a}_j^H \mathbf{x}^{(t+1)} = b_j, \forall j \in \mathcal{F}$, as demonstrated below:

$$\begin{aligned} \mathbf{a}_j^H \mathbf{x}^{(t+1)} &= \mathbf{a}_j^H \mathbf{x}^{(t)} + \mathbf{a}_j^H \frac{(\varphi^{(t)})^H \mathbf{r}^{(t)}}{\|\mathbf{A}^H \varphi^{(t)}\|_2^2} \mathbf{A}^H \varphi^{(t)} \\ &= \mathbf{a}_j^H \mathbf{x}^{(t)} + \frac{(\varphi^{(t)})^H \mathbf{r}^{(t)}}{\|\mathbf{A}^H \varphi^{(t)}\|_2^2} \mathbf{a}_j^H \mathbf{A}^H \varphi^{(t)} \\ &= \mathbf{a}_j^H \mathbf{x}^{(t)} + \frac{\sum_{i \in \mathcal{F}} [\varphi^{(t)}]_i r_i^{(t)}}{\|\sum_{i \in \mathcal{F}} [\varphi^{(t)}]_i \mathbf{a}_i\|_2^2} \mathbf{a}_j^H \sum_{i \in \mathcal{F}} [\varphi^{(t)}]_i \mathbf{a}_i. \end{aligned} \quad (63)$$

By using the property of orthogonality, we can establish the following relationship

$$\mathbf{a}_j^H \mathbf{x}^{(t+1)} = \mathbf{a}_j^H \mathbf{x}^{(t)} + \frac{\sum_{i \in \mathcal{F}} [\varphi^{(t)}]_i r_i^{(t)}}{\sum_{i \in \mathcal{F}} \|\varphi^{(t)}\|_2^2} [\varphi^{(t)}]_j \|\mathbf{a}_j\|_2^2. \quad (64)$$

Next, by substituting (31) into (64), we can arrive at the following result

$$\begin{aligned} \mathbf{a}_j^H \mathbf{x}^{(t+1)} &= \mathbf{a}_j^H \mathbf{x}^{(t)} + \frac{\sum_{i \in \mathcal{F}} [\varphi^{(t)}]_i r_i^{(t)}}{\sum_{i \in \mathcal{F}} [\varphi^{(t)}]_i r_i^{(t)}} [\varphi^{(t)}]_j \|\mathbf{a}_j\|_2^2 \\ &= \mathbf{a}_j^H \mathbf{x}^{(t)} + r_j^{(t)} \\ &= \mathbf{a}_j^H \mathbf{x}^{(t)} + (b_j - \mathbf{a}_j^H \mathbf{x}^{(t)}) = b_j. \end{aligned} \quad (65)$$

Hence, the proof of Theorem 4 is completed.

APPENDIX D

We assume that there exists a set of mutually orthogonal hyperplanes in the linear system, denoted as \mathcal{F} , i.e., $\mathbf{a}_i^H \mathbf{a}_j = 0, \forall i, j \in \mathcal{F}, i \neq j$. At the t -th iteration, the aggregation hyperplane is generated by the hyperplanes in \mathcal{F} , i.e.,

$$[\varphi^{(t)}]_i = \begin{cases} \frac{b_i - \mathbf{a}_i^H \mathbf{x}^{(t)}}{\|\mathbf{a}_i^H\|_2^2}, & i \in \mathcal{F}, \\ 0, & i \notin \mathcal{F}. \end{cases} \quad (66)$$

Thus, the next iteration solution $\mathbf{x}^{(t+1)}$ is calculated as

$$\begin{aligned}
\mathbf{x}^{(t+1)} &= \mathbf{x}^{(t)} + \frac{(\boldsymbol{\varphi}^{(t)})^H \mathbf{r}^{(t)}}{\|\mathbf{A}^H \boldsymbol{\varphi}^{(t)}\|_2^2} \mathbf{A}^H \boldsymbol{\varphi}^{(t)} \\
&= \mathbf{x}^{(t)} + \frac{\sum_{i \in \mathcal{F}} |\boldsymbol{\varphi}^{(t)}|_i r_i^{(t)}}{\|\sum_{i \in \mathcal{F}} |\boldsymbol{\varphi}^{(t)}|_i \mathbf{a}_i\|_2^2} \sum_{i \in \mathcal{F}} |\boldsymbol{\varphi}^{(t)}|_i \mathbf{a}_i \\
&= \mathbf{x}^{(t)} + \frac{\sum_{i \in \mathcal{F}} |\boldsymbol{\varphi}^{(t)}|_i r_i^{(t)}}{\sum_{i \in \mathcal{F}} |\boldsymbol{\varphi}^{(t)}|_i r_i^{(t)}} \sum_{i \in \mathcal{F}} |\boldsymbol{\varphi}^{(t)}|_i \mathbf{a}_i \\
&= \mathbf{x}^{(t)} + \sum_{i \in \mathcal{F}} |\boldsymbol{\varphi}^{(t)}|_i \mathbf{a}_i. \tag{67}
\end{aligned}$$

Hence, the proof of Proposition 1 is completed.

REFERENCES

- [1] X. You *et al.*, "Towards 6G wireless communication networks: Vision, enabling technologies, and new paradigm shifts," *Sci. China Inf. Sci.*, vol. 64, pp. 1–74, Nov. 2021.
- [2] Z. Zhang *et al.*, "6G wireless networks: Vision, requirements, architecture, and key technologies," *IEEE Veh. Technol. Mag.*, vol. 14, no. 3, pp. 28–41, Sep. 2019.
- [3] C.-X. Wang *et al.*, "On the road to 6G: Visions, requirements, key technologies, and testbeds," *IEEE Commun. Surveys Tuts.*, vol. 25, no. 2, pp. 905–974, 2nd Quart 2023.
- [4] —, "6G wireless channel measurements and models: Trends and challenges," *IEEE Veh. Technol. Mag.*, vol. 15, no. 4, pp. 22–32, Dec. 2020.
- [5] S. Rangan, T. S. Rappaport, and E. Erkip, "Millimeter-wave cellular wireless networks: Potentials and challenges," *Proc. IEEE*, vol. 102, no. 3, pp. 366–385, Mar. 2014.
- [6] R. W. Heath *et al.*, "An overview of signal processing techniques for millimeter wave MIMO systems," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 3, pp. 436–453, Apr. 2016.
- [7] H.-J. Song and T. Nagatsuma, "Present and future of Terahertz communications," *IEEE Trans. THz Sci. Technol.*, vol. 1, no. 1, pp. 256–263, Sep. 2011.
- [8] I. F. Akyildiz, C. Han, and S. Nie, "Combating the distance problem in the millimeter wave and Terahertz frequency bands," *IEEE Commun. Mag.*, vol. 56, no. 6, pp. 102–108, Jun. 2018.
- [9] C. Pan *et al.*, "Reconfigurable intelligent surfaces for 6G systems: Principles, applications, and research directions," *IEEE Commun. Mag.*, vol. 59, no. 6, pp. 14–20, Jun. 2021.
- [10] Y. Liu, X. Liu, X. Mu, T. Hou, J. Xu, M. Di Renzo, and N. Al-Dhahir, "Reconfigurable intelligent surfaces: Principles and opportunities," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1546–1577, 3rd Quart. 2021.
- [11] Q. Wu, S. Zhang, B. Zheng, C. You, and R. Zhang, "Intelligent reflecting surface-aided wireless communications: A tutorial," *IEEE Trans. Commun.*, vol. 69, no. 5, pp. 3313–3351, May. 2021.
- [12] Z. Wang *et al.*, "Extremely large-scale MIMO: Fundamentals, challenges, solutions, and future directions," *IEEE Wireless Commun.*, vol. 31, no. 3, pp. 117–124, Jun. 2024.
- [13] E. Björnson and others, "Massive MIMO is a reality—what is next?: Five promising research directions for antenna arrays," *Digit. Signal Process.*, vol. 94, pp. 3–20, Nov. 2019.
- [14] F. Tariq *et al.*, "A speculative study on 6G," *IEEE Wireless Commun.*, vol. 27, no. 4, pp. 118–125, Aug. 2020.
- [15] W. Saad, M. Bennis, and M. Chen, "A vision of 6G wireless systems: Applications, trends, technologies, and open research problems," *IEEE Netw.*, vol. 34, no. 3, pp. 134–142, May. 2020.
- [16] E. D. Carvalho, A. Ali, A. Amiri, M. Angelichinoski, and R. W. Heath, "Non-stationarities in extra-large-scale massive MIMO," *IEEE Wireless Commun.*, vol. 27, no. 4, pp. 74–80, Aug. 2020.
- [17] J. Hoydis, S. ten Brink, and M. Debbah, "Massive MIMO in the UL/DL of cellular networks: How many antennas do we need?" *IEEE J. Sel. Areas Commun.*, vol. 31, no. 2, pp. 160–171, Feb. 2013.
- [18] N. Fatema *et al.*, "Massive MIMO linear precoding: A survey," *IEEE syst. J.*, vol. 12, no. 4, pp. 3920–3931, Dec. 2018.
- [19] D. Zhu, B. Li, and P. Liang, "On the matrix inversion approximation based on neumann series in massive MIMO systems," in *Proc IEEE ICC*, Jun. 2015, pp. 1763–1769.
- [20] B. Nagy, M. Elsabrouty, and S. Elramly, "Fast converging weighted neumann series precoding for massive MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 2, pp. 154–157, Apr. 2018.
- [21] M. N. Boroujerdi *et al.*, "Low-complexity statistically robust precoder/detector computation for massive MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 17, no. 10, pp. 6516–6530, Oct. 2018.
- [22] J. Minango and C. de Almeida, "A low-complexity linear precoding algorithm based on Jacobi method for massive MIMO systems," in *Proc. IEEE 87th Veh. Technol. Conf. (VTC Spring)*, Jun. 2018, pp. 1–5.
- [23] Z. Lu, J. Ning, Y. Zhang, T. Xie, and W. Shen, "Richardson method based linear precoding with low complexity for massive MIMO systems," in *Proc. IEEE 81st Veh. Technol. Conf. (VTC Spring)*, May. 2015, pp. 1–4.
- [24] Z. Wang *et al.*, "Rapidly converging low-complexity iterative transmit precoders for massive MIMO downlink," *IEEE Trans. Commun.*, vol. 71, no. 12, pp. 7228–7243, Dec. 2023.
- [25] T. Strohmer and R. Vershynin, "A randomized kaczmarz algorithm with exponential convergence," *J. Fourier Anal. Appl.*, vol. 15, no. 2, pp. 262–278, Apr. 2009.
- [26] V. Croisfelt *et al.*, "Accelerated randomized methods for receiver design in extra-large scale MIMO arrays," *IEEE Trans. Veh. Technol.*, vol. 70, no. 7, pp. 6788–6799, Jul. 2021.
- [27] T. Strohmer and R. Vershynin, "Comments on the randomized kaczmarz method," *J. Fourier Anal. Appl.*, vol. 15, no. 4, pp. 437–440, 2009.
- [28] Y. Censor, G. T. Herman, and M. Jiang, "A note on the behavior of the randomized kaczmarz algorithm of strohmer and vershynin," *J. Fourier Anal. Appl.*, vol. 15, pp. 431–436, 2009.
- [29] Y. Liu, Z. Wang, J. Xu, C. Ouyang, X. Mu, and R. Schober, "Near-field communications: A tutorial review," *IEEE Open J. Commun. Soc.*, vol. 4, pp. 1999–2049, Aug. 2023.
- [30] H. Lu *et al.*, "A tutorial on near-field XL-MIMO communications towards 6G," *IEEE Commun. Surveys Tuts.*, pp. 1–1, 2024.
- [31] C.-X. Wang *et al.*, "A complete study of space-time-frequency statistical properties of the 6G pervasive channel model," *IEEE Trans. Commun.*, vol. 71, no. 12, pp. 7273–7287, Dec. 2023.
- [32] K. Zhi *et al.*, "Performance analysis and low-complexity design for XL-MIMO with near-field spatial non-stationarities," *IEEE J. Sel. Areas Commun.*, vol. 42, no. 6, pp. 1656–1672, Jun. 2024.
- [33] V. C. Rodrigues, A. Amiri, T. Abrão, E. de Carvalho, and P. Popovski, "Low-complexity distributed XL-MIMO for multiuser detection," in *Proc IEEE ICC Workshops*, Jun. 2020, pp. 1–6.
- [34] B. Xu, Z. Wang, H. Xiao, J. Zhang, B. Ai, and D. W. Kwan Ng, "Low-complexity precoding for extremely large-scale MIMO over non-stationary channels," in *Proc IEEE ICC*, May. 2023, pp. 6516–6522.
- [35] Z. Yuan *et al.*, "Spatial non-stationary near-field channel modeling and validation for massive MIMO systems," *IEEE Trans. Antennas Propag.*, vol. 71, no. 1, pp. 921–933, Jan. 2023.
- [36] C.-X. Wang *et al.*, "Pervasive wireless channel modeling theory and applications to 6G gbsms for all frequency bands and all scenarios," *IEEE Trans. Veh. Technol.*, vol. 71, no. 9, pp. 9159–9173, Sep. 2022.
- [37] A. Zouzias and N. M. Freris, "Randomized extended kaczmarz for solving least squares," *SIAM J. Matrix Anal. Appl.*, vol. 34, no. 2, pp. 773–793, 2013.
- [38] M. Griebel and P. Oswald, "Greedy and randomized versions of the multiplicative Schwarz method," *Linear Algebra Appl.*, vol. 437, no. 7, pp. 1596–1610, Oct. 2012.
- [39] Z. Bai and W. Wu, "On greedy randomized kaczmarz method for solving large sparse linear systems," *SIAM J. Sci. Comput.*, vol. 40, no. 1, pp. A592–A606, 2018.
- [40] F. Glover, "Surrogate constraints," *Oper. Res.*, vol. 16, no. 4, pp. 741–749, 1968.
- [41] —, "Tutorial on surrogate constraint approaches for optimization in graphs," *J. Heuristics*, vol. 9, pp. 175–227, 2003.
- [42] L. Chang, W. Li, and W. Zhang, "Computing a near-maximum independent set in linear time by reducing-peeling," in *Proc. ACM International Conf. Management of Data*, 2017, p. 1181–1196.
- [43] G. H. Golub and C. F. Van Loan, *Matrix computations*. The John Hopkins University Press, 2013.
- [44] J. Nutini *et al.*, "Convergence rates for greedy kaczmarz algorithms, and faster randomized kaczmarz rules using the orthogonality graph," *Proc. UAI*, pp. 1–14, 2016.
- [45] H. Iimori *et al.*, "Joint activity and channel estimation for extra-large MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 21, no. 9, pp. 7253–7270, Sep. 2022.
- [46] Y. Han *et al.*, "Localization and channel reconstruction for extra large RIS-assisted massive MIMO systems," *IEEE J. Sel. Topics Signal Process.*, vol. 16, no. 5, pp. 1011–1025, Aug. 2022.