Master-Assisted Distributed Uplink Operation for User-Centric Cell-Free Massive MIMO Networks

Andreas Angelou, *Graduate Student Member, IEEE,* Pourya Behmandpoor, *Member, IEEE,* Marc Moonen, *Fellow, IEEE*

***Abstract*—Cell-free massive multiple-input-multiple-output (CFmMIMO) is considered a promising technology for sixth-generation (6G) wireless communication networks. The main idea is to distribute a large number of access points (APs) in a geographical region to serve the user equipments (UEs). In the uplink, CFmMIMO networks typically adopt one of two types of operations: centralized or distributed. In centralized operation, channel estimation and data detection are performed at the central processing unit (CPU), whereas in distributed operation, channel estimation occurs at the APs and final data detection at the CPU. In this paper, we propose a novel uplink operation, termed *Master-Assisted Distributed Uplink Operation* (MADUO), where each UE is assigned a master AP (MAP) and a set of additional serving APs (ASAPs). The MAP receives soft data estimates from the ASAPs and detects the data using its own local signals and the received data estimates from the ASAPs. Moreover, we introduce a statistical cooperation framework for MADUO, which controls the trade-off between performance and fronthaul overhead. Numerical experiments demonstrate that MADUO offers near-centralized performance and a favorable trade-off between fronthaul overhead and computational complexity.**

***Index Terms*—Cell-free massive MIMO, distributed processing, receive combining, uplink, spectral efficiency.**

I. Introduction

In cellular networks, access points (APs) are deployed over a geographical region, and each user equipment (UE) is served by only one AP. Each AP serves all UEs within its coverage area, which is called a cell. These cells define disjoint regions in which the UEs are subject to interference from both other UEs and the AP within the same cell (intra-cell interference) and neighboring cells (inter-cell interference).

A plethora of technologies have been developed based on cellular network topology. For example, in multiuser multiple-input-multiple-output (MU-MIMO), each AP is equipped with multiple co-located antennas and serves multiple UEs sharing time-frequency resources [2], [3]. Massive MIMO (M-MIMO), a successor to MU-MIMO, equips each AP with significantly more antennas (typically exceeding the number of UEs within the cell) [3], [4]. M-MIMO can offer remarkable performance gains with a large number of antennas, and allows for simple signal processing at the APs thanks to the effects of channel hardening and favorable propagation. However, this technology faces challenges with inter-cell interference, which is inherent to cellular networks and stems from the independent operation of each AP. This challenge is prominent for UEs located at cell edges as they experience low signal-to-interference-plus-noise ratio (SINR) conditions due to inter-cell interference, together with a high path loss to their serving AP.

A technology that addresses the inherent challenge of inter-cell interference in cellular networks is called Network MIMO [5], otherwise known as distributed MIMO [6]. The core concept of Network MIMO is that each UE is served by multiple cooperative APs. This cooperation among APs enables the exploitation and mitigation of inter-cell interference, leading to significant performance gains [5]. Despite its theoretical promise, Network MIMO has underperformed in practice, due to poor channel estimation and inadequate fast synchronization across APs [7].

Cell-free massive MIMO (CFmMIMO) was introduced in 2015 [8] to address the bottleneck of inter-cell interference in cellular networks by eliminating the notion of cells [9]. In a canonical CFmMIMO network, the APs are deployed over a geographical region and serve all UEs in the network cooperatively, thereby eliminating conventional cell boundaries. As in M-MIMO, the total number of antennas in the network is much larger than the number of UEs, so that interference can be canceled successfully [10] and high-quality service can be provided to all UEs [9], effectively blurring the lines between fortunate and unfortunate UEs. Owing to this attribute, CFmMIMO has drawn considerable attention and is considered a promising technology for the sixth-generation (6G) of wireless communication networks [7], [11]–[14].

Despite the favorable attributes of canonical CFmMIMO networks, scalability issues arise from the requirement that all APs jointly serve all UEs. These scalability issues have motivated the conception of the user-centric topology, where each UE is served by a subset of APs with the most favorable channel conditions. The procedure that forms these subsets is called dynamic cooperation clustering (DCC) [15], [16]. DCC reduces fronthaul overhead and computational complexity with marginal performance loss compared to that of a canonical CFmMIMO network, rendering the user-centric CFmMIMO network a scalable and thus practical topology.

In the uplink, CFmMIMO networks typically adopt one of

A conference version of this paper can be found in [1].

This research work was carried out at the ESAT Laboratory of KU Leuven, in the frame of FWO Research Project nr. G0C0623N "User-centric distributed signal processing algorithms for next generation cell-free massive MIMO based wireless communication networks". The scientific responsibility is assumed by its authors.

Andreas Angelou and Marc Moonen are with the Department of Electrical Engineering (ESAT), STADIUS Center for Dynamical Systems, Signal Processing and Data Analytics, KU Leuven, 3000 Leuven, Belgium (e-mail: andreas.angelou@esat.kuleuven.be; marc.moonen@esat.kuleuven.be). Pourya Behmandpoor is with the Department of Electronics and Informatics, Vrije Universiteit Brussel, 1050 Brussels, Belgium (e-mail: pourya.behmandpoor@vub.be).

two types of operation: centralized or distributed. In centralized operation, APs send their received pilot and data signals to the central processing unit (CPU), which performs channel estimation and data detection using global channel state information (CSI). In distributed operation, the APs locally perform channel estimation and compute soft data estimates, which are then sent to the CPU for final data detection using large-scale fading decoding (LSFD), a decoding method that uses only statistical CSI [11].

Numerous receive combining schemes have been proposed in the literature for both centralized and distributed operation. In centralized operation, the *Centralized Minimum-Mean-Squared-Error* (C-MMSE) combiner is optimal in the mean-square sense [11], but the CPU uses global CSI to cancel interference from all UEs in the network. Using information from all UEs makes this combiner non-scalable as the number of UEs grows. A scalable version of C-MMSE is the *Partial MMSE* (P-MMSE) [15], where the CPU cancels interference only from the UEs in the neighborhood of the UE of interest. In distributed operation, the APs estimate the data signal and then send it to the CPU. The optimal local receive combiner is called *Local MMSE* (L-MMSE) [11], where each AP uses global CSI to cancel interference, and the scalable version is called *Local Partial MMSE* (LP-MMSE) [17], where each AP uses only local CSI to mitigate interference. Other combiners, such as the zero-forcing [18] and maximum ratio (MR) combiner [9], have been proposed in the literature. However, results in [11] demonstrate that MMSE-based processing is essential to achieve good performance in CFmMIMO networks.

In [19], an optimal bilinear combiner has been proposed to reduce computational complexity by replacing the matrix inversion in MMSE-based combiners with a statistics-dependent matrix, at the cost of some performance loss. In [20], three Team MMSE (T-MMSE) combiners have been proposed, which achieved near-optimal performance with reduced CSI sharing. In [21], an improved P-MMSE (IP-MMSE) combiner has been proposed, which allows for analytic derivation of the achievable spectral efficiency (SE) and provides better performance compared to P-MMSE. In [22], a hybrid uplink scheme has been proposed, which compresses the received signals at the APs before sending them to the CPU for centralized detection with P-MMSE, reducing fronthaul overhead. In [23], heuristics for the computation of the LSFD coefficients at the APs have been proposed to reduce fronthaul overhead in the distributed operation. In [24], an over-the-air framework has been proposed, allowing the APs to compute and send only the necessary information to the CPU for final data detection, thereby reducing fronthaul overhead, with marginal performance loss.

A. Contributions

Despite the significant interest in combining schemes in CFmMIMO, there remains a need for efficient schemes that provide favorable trade-offs between SE performance, fronthaul overhead, and computational complexity. To address this gap, we propose a novel distributed uplink operation, called *Master-Assisted Distributed Uplink Operation* (MADUO). In MADUO, each UE is served by a master AP (MAP) and a set of additional serving APs (ASAPs). The ASAPs perform local signal estimation for their served UEs and forward the signal estimates, along with compressed local CSI, to the corresponding MAPs for final data detection. Moreover, we introduce a statistical cooperation framework comprising three cooperation levels, in which the ASAPs send progressively less information to the CPU, thereby reducing fronthaul overhead at the expense of SE performance. We highlight that MADUO is different from the distributed operation, as further explained in Section III.

The contributions of this paper are summarized as follows:

- We develop a novel distributed uplink operation, called MADUO, that effectively balances SE performance, fronthaul overhead, and computational complexity;
- We analyze the SE performance and derive closed-form expressions for the achievable SE in MADUO;
- We quantify the fronthaul overhead and computational complexity for MADUO, and compare them with the complexity of centralized and distributed operation;
- We introduce a statistical cooperation framework for MADUO comprising three cooperation levels, in which the ASAPs send progressively less information to the MAP, and evaluate the trade-off between performance and fronthaul overhead;
- We present numerical experiments comparing MADUO with centralized and distributed operation in terms of SE performance, fronthaul overhead, and computational complexity.

B. Paper Outline

The paper is structured as follows. Section II presents the system model for uplink CFmMIMO, focusing on pilot transmission and channel estimation, data transmission, and receive combining. Section III presents MADUO, the main contribution of this paper, and discusses receive combining, fronthaul overhead, and computational complexity of the proposed operation. Section IV introduces a statistical cooperation framework for MADUO. Section V describes the simulation setup and presents the numerical results. Finally, Section VI presents the conclusions.[1]

C. Notation

Lowercase letters x denote complex scalars, bold lowercase letters $\mathbf{x}$ denote complex column vectors, and bold uppercase letters $\mathbf{X}$ denote complex matrices. The complex conjugate of a scalar x is denoted by x^*, and the ℓ_2-norm of a vector $\mathbf{x}$ is denoted by $\|\mathbf{x}\|$. A random vector $\mathbf{x} \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \mathbf{R})$ is a zero-mean circularly symmetric complex Gaussian vector with covariance matrix $\mathbf{R}$. The Hermitian (conjugate transpose) of a matrix is denoted by $(\cdot)^H$, the regular transpose is denoted by $(\cdot)^T$, the inverse of a matrix is denoted by $(\cdot)^{-1}$, and the pseudoinverse by $(\cdot)^\dagger$. The expectation operator is denoted by $\mathbb{E}\{\cdot\}$, the trace of a matrix is denoted by $\mathrm{tr}(\cdot)$, and $\mathbf{I}_N$

[1]The numerical experiments are conducted using MATLAB. The code is available at...

denotes the $N \times N$ identity matrix. The all-zeros $N \times N$ matrix is denoted by $\mathbf{0}_{N\times N}$ and the all-zeros N-dimensional vector by $\mathbf{0}_N$. The notation $[\tau]$ denotes the set $\{1,\ldots,\tau\}$ for $\tau \in \mathbb{Z}_+$. A diagonal matrix with elements $x_1,\ldots,x_n$ on the diagonal is denoted by $\mathtt{diag}\{x_i\}_{i=1}^n$, and a block-diagonal matrix with rectangular blocks $\mathbf{X}_1,\ldots,\mathbf{X}_n$ along the diagonal is denoted by $\mathtt{blkdiag}\{\mathbf{X}_i\}_{i=1}^n$ or $\mathtt{blkdiag}(\mathbf{X}_1,\ldots,\mathbf{X}_n)$. Sets are denoted using calligraphic font, such as $\mathcal{X}$, and the cardinality of a set $\mathcal{X}$ is denoted by $|\mathcal{X}|$.

II. User-Centric Cell-Free Massive MIMO Model

We consider a user-centric CFmMIMO network with K UEs, each equipped with a single antenna, and L APs, each equipped with N antennas, with $NL \gg K$. The subset of APs serving UE k is denoted by $\mathcal{A}_k$, and the subset of UEs served by AP j is denoted by $\mathcal{U}_j$. The UEs and APs are randomly distributed over a large area, and the APs are connected to the CPU with ideal fronthaul links. We denote the uplink channel between UE k and AP j by $\mathbf{h}_{j,k} \in \mathbb{C}^N$. The collective channel of UE k to all APs is denoted by $\mathbf{h}_k = \left[\mathbf{h}_{1,k}^T \ \ldots \ \mathbf{h}_{L,k}^T\right]^T \in \mathbb{C}^{NL}$, and the channels from all UEs to AP j are collected into a matrix $\mathbf{H}_j = \left[\mathbf{h}_{j,1} \ \ldots \ \mathbf{h}_{j,K}\right] \in \mathbb{C}^{N\times K}$. We adopt the standard block fading model, where channels are constant over time-frequency blocks (coherence blocks) of τ_c samples. Since we focus on the uplink, we assume that the coherence blocks are used only for uplink pilot and data transmission, disregarding downlink data transmission. The pilot transmission phase uses τ_p samples, and the uplink data transmission phase uses $\tau_u = \tau_c - \tau_p$ samples.

We adopt the following definition of scalability, which was considered in [15] and [17]:

Definition 1. *A cell-free network is scalable if the following tasks are practically implementable as $K \to \infty$: signal processing for channel estimation; data reception and transmission; fronthaul overhead for data and CSI sharing; and power allocation optimization.*

A. Channel Model

We model the uplink channel between UE k and AP j in a considered coherence block with correlated Rayleigh fading

$$\mathbf{h}_{j,k} \sim \mathcal{N}_{\mathbb{C}}\left(\mathbf{0}_N, \mathbf{R}_{j,k}\right), \tag{1}$$

where $\mathbf{R}_{j,k} \in \mathbb{C}^{N\times N}$ is the covariance matrix and captures the spatial properties of the channel, geometric pathloss, and shadowing. The channel realizations $\mathbf{h}_{j,k} \in \mathbb{C}^N$ model the small-scale fading effects and are drawn independently. We further assume that the covariance matrices $\{\mathbf{R}_{j,k} : j \in [L], k \in [K]\}$ remain constant over a large number of coherence blocks. In practice, these spatial statistics vary after several thousands of coherence blocks [25], [26]; however, for conciseness, we consider them time-invariant [26], [27].

B. Pilot Transmission and Channel Estimation

We assume that τ_p mutually orthogonal pilot signals $\boldsymbol{\phi}_{t_k} \in \mathbb{C}^{\tau_p}$ are used during pilot transmission, with $t_k \in [\tau_p]$ the pilot index of UE i, and $\mathbb{E}\{\|\boldsymbol{\phi}_{t_k}\|^2\} = \tau_p$. Each UE is assigned a pilot; a possible pilot assignment algorithm is described in [15, Sec. V.A]. We consider a large network where $K > \tau_p$, which means that multiple UEs share the same pilot signal, i.e., $t_k = t_i$, for some pairs of UEs (k,i). In the pilot transmission phase, AP j receives $\mathbf{Y}_j^{\text{pilot}} \in \mathbb{C}^{N\times\tau_p}$, given by

$$\mathbf{Y}_j^{\text{pilot}} = \sum_{i=1}^{K} \sqrt{p_i}\mathbf{h}_{j,i}\boldsymbol{\phi}_{t_i}^T + \mathbf{N}_j\,, \tag{2}$$

where p_i is the transmit power of UE i, and $\mathbf{N}_j \in \mathbb{C}^{N\times\tau_p}$ is the receiver noise with independent $\mathcal{N}_{\mathbb{C}}\left(0,\sigma^2\right)$ elements. For the channel estimation of UE k, AP j despreads $\mathbf{Y}_j^{\text{pilot}}$ with the normalized pilot of UE k, $\boldsymbol{\phi}_{t_k}^*/\sqrt{\tau_p}$ [17, Sec. 4] to obtain

$$\mathbf{y}_{j,t_k}^{\text{pilot}} = \mathbf{Y}_j^{\text{pilot}}\boldsymbol{\phi}_{t_k}^*/\sqrt{\tau_p} = \sum_{i\in\mathcal{P}_k}\sqrt{p_i\tau_p}\mathbf{h}_{j,i} + \mathbf{n}_{j,t_k}\,, \tag{3}$$

where $\mathcal{P}_k$ is the set of UEs with the same pilot as UE k (including UE k itself), and $\mathbf{n}_{j,t_k} \sim \mathcal{N}_{\mathbb{C}}\left(\mathbf{0}_N, \sigma^2\mathbf{I}_N\right)$ is the independent noise after despreading. The MMSE channel estimate of UE k at AP j is given by [15]

$$\widehat{\mathbf{h}}_{j,k} = \sqrt{p_k\tau_p}\,\mathbf{R}_{j,k}\boldsymbol{\Psi}_{j,t_k}^{-1}\mathbf{y}_{j,t_k}^{\text{pilot}}\,, \tag{4}$$

where

$$\boldsymbol{\Psi}_{j,t_k} = \mathbb{E}\left\{\mathbf{y}_{j,t_k}^{\text{pilot}}\left(\mathbf{y}_{j,t_k}^{\text{pilot}}\right)^H\right\} = \sum_{i\in\mathcal{P}_k} p_i\tau_p\mathbf{R}_{j,i} + \sigma^2\mathbf{I}_N\,. \tag{5}$$

We note that the channel estimate in (4) and the channel estimation error $\widetilde{\mathbf{h}}_{j,k} = \mathbf{h}_{j,k} - \widehat{\mathbf{h}}_{j,k}$ are independent random vectors [17] and are distributed as $\widehat{\mathbf{h}}_{j,k} \sim \mathcal{N}_{\mathbb{C}}\left(\mathbf{0}_N, p_k\tau_p\mathbf{R}_{j,k}\boldsymbol{\Psi}_{j,t_k}^{-1}\mathbf{R}_{j,k}\right)$ and $\widetilde{\mathbf{h}}_{j,k} \sim \mathcal{N}_{\mathbb{C}}\left(\mathbf{0}_N, \mathbf{C}_{j,k}\right)$, where

$$\mathbf{C}_{j,k} = \mathbf{R}_{j,k} - p_k\tau_p\mathbf{R}_{j,k}\boldsymbol{\Psi}_{j,t_k}^{-1}\mathbf{R}_{j,k}\,. \tag{6}$$

Equations (4) and (5) indicate that AP j requires knowledge of the covariance matrices $\{\mathbf{R}_{j,i} : i \in \mathcal{P}_k\}$, which is assumed to be available; see [28], [29] for possible methods to estimate these matrices. The channel estimates of AP j are collected into a matrix $\widehat{\mathbf{H}}_j = \left[\widehat{\mathbf{h}}_{j,1} \ \ldots \ \widehat{\mathbf{h}}_{j,K}\right] \in \mathbb{C}^{N\times K}$.

C. Uplink Data Transmission and Receive Combining

In the uplink data transmission phase, the UEs send their data signals to the APs. The received signal $\mathbf{y}_j \in \mathbb{C}^N$ of AP j is given by

$$\mathbf{y}_j = \sum_{i=1}^{K}\mathbf{h}_{j,i}s_i + \mathbf{n}_j\,, \tag{7}$$

where $s_i \sim \mathcal{N}_{\mathbb{C}}\left(0, p_i\right)$ denotes the data signal of UE i, and $\mathbf{n}_j \sim \mathcal{N}_{\mathbb{C}}\left(\mathbf{0}_N, \sigma^2\mathbf{I}_N\right)$ is the independent receiver noise. Here, we define the matrix $\mathbf{D}_{j,k}$ which is equal to the identity matrix $\mathbf{I}_N$ if UE k is served by AP j (i.e., $j \in \mathcal{A}_k$ and $k \in \mathcal{U}_j$) and $\mathbf{0}_{N\times N}$ otherwise, and $\mathbf{D}_k = \mathtt{blkdiag}\left\{\mathbf{D}_{j,k}\right\}_{j=1}^L$.

1) Centralized Operation: In centralized operation, the APs forward their locally received signals $\mathbf{y}_j$ to the CPU[2], which concatenates them into a long vector $\mathbf{y} = \begin{bmatrix} \mathbf{y}_1^T & \ldots & \mathbf{y}_L^T \end{bmatrix}^T \in \mathbb{C}^{LN}$, i.e.,

$$\mathbf{y} = \sum_{i=1}^{K} \mathbf{h}_i s_i + \mathbf{n}, \tag{8}$$

where $\mathbf{n} = \begin{bmatrix} \mathbf{n}_1^T & \ldots & \mathbf{n}_L^T \end{bmatrix}^T$. The CPU then estimates the signal s_k by linearly combining its received signals using the receive combiner $\mathbf{D}_k \mathbf{v}_k$, and obtains the signal estimate

$$\hat{s}_k = \mathbf{v}_k^H \mathbf{D}_k \mathbf{y}. \tag{9}$$

Note that the receive combiner includes the matrix $\mathbf{D}_k$ because each UE is served by only a subset of APs, and the product $\mathbf{D}_k \mathbf{y}$ sets the signals of the APs $j \notin \mathcal{A}_k$ to zero. An achievable SE of UE k in the centralized operation is given by [17]

$$\mathrm{SE}_k^{(\mathrm{c})} = \frac{\tau_u}{\tau_c} \mathbb{E}\left\{ \log_2 \left(1 + \mathrm{SINR}_k^{(\mathrm{c})} \right) \right\}, \tag{10}$$

where

$$\mathrm{SINR}_k^{(\mathrm{c})} = \frac{p_k \left| \mathbf{v}_k^H \mathbf{D}_k \widehat{\mathbf{h}}_k \right|^2}{\sum\limits_{\substack{i=1 \\ i \neq k}}^{K} p_i \left| \mathbf{v}_k^H \mathbf{D}_k \widehat{\mathbf{h}}_i \right|^2 + \mathbf{v}_k^H \mathbf{Z}_k \mathbf{v}_k + \sigma^2 \left\| \mathbf{D}_k \mathbf{v}_k \right\|^2} \tag{11}$$

with $\mathbf{Z}_k = \sum_{i=1}^{K} p_i \mathbf{D}_k \mathbf{C}_i \mathbf{D}_k$, $\mathbf{C}_i = \mathtt{blkdiag}\{\mathbf{C}_{j,i}\}_{j=1}^{L}$, and $\widehat{\mathbf{h}}_i = \begin{bmatrix} \widehat{\mathbf{h}}_{1,i}^T & \ldots & \widehat{\mathbf{h}}_{L,i}^T \end{bmatrix}^T$. The optimal linear C-MMSE combiner $\mathbf{v}_k^{\text{C-MMSE}} \in \mathbb{C}^{LN}$ maximizes (10) and is defined as

$$\mathbf{v}_k^{\text{C-MMSE}} = p_k \mathbf{R}_{\mathbf{y}}^{-1} \mathbf{D}_k \widehat{\mathbf{h}}_k, \tag{12}$$

where

$$\mathbf{R}_{\mathbf{y}} = \sum_{i=1}^{K} p_i \mathbf{D}_k \left(\widehat{\mathbf{h}}_i \widehat{\mathbf{h}}_i^H + \mathbf{C}_i \right) \mathbf{D}_k + \sigma^2 \mathbf{I}_{LN}. \tag{13}$$

It is evident from (12) and (13) that C-MMSE uses global CSI, which makes it non-scalable by Definition 1. A scalable alternative is to use information only from the UEs that are served by the APs that also serve UE k, i.e., UEs in the set $\mathcal{S}_k = \bigcup_{j \in \mathcal{A}_k} \mathcal{U}_j$, to obtain the P-MMSE combiner $\mathbf{v}_k^{\text{P-MMSE}} \in \mathbb{C}^{LN}$, given by

$$\mathbf{v}_k^{\text{P-MMSE}} = p_k \overline{\mathbf{R}}_{\mathbf{y}}^{-1} \mathbf{D}_k \widehat{\mathbf{h}}_k, \tag{14}$$

where

$$\overline{\mathbf{R}}_{\mathbf{y}} = \sum_{i \in \mathcal{S}_k} p_i \mathbf{D}_k \left(\widehat{\mathbf{h}}_i \widehat{\mathbf{h}}_i^H + \mathbf{C}_i \right) \mathbf{D}_k + \sigma^2 \mathbf{I}_{LN}, \tag{15}$$

in which the summation is over the UEs in $\mathcal{S}_k$ rather than all UEs in (13).

[2]The CPU should be viewed as a collection of centralized processing tasks that need to be executed somewhere in the network and not as a physical processing unit [15]. For one UE, these tasks can be executed at one of the serving APs of that UE, which has been called *master AP* [15] or *master base station* [30] in the literature. However, this is *only one way* to divide these tasks in the network, and other alternatives exist [15].

2) Distributed Operation: In distributed operation, APs locally estimate the signal s_k by linearly combining their local received signals using a combiner $\mathbf{D}_{j,k} \mathbf{v}_{j,k}$, i.e.,

$$\hat{s}_{j,k} = \mathbf{v}_{j,k}^H \mathbf{D}_{j,k} \mathbf{y}_j. \tag{16}$$

The use of $\mathbf{D}_{j,k}$ ensures that AP j only estimates the signals of the UEs that it serves. The APs then use the L-MMSE combiner $\mathbf{v}_{j,k}^{\text{L-MMSE}} \in \mathbb{C}^N$, given by

$$\mathbf{v}_{j,k}^{\text{L-MMSE}} = p_k \mathbf{R}_{\mathbf{y}_j}^{-1} \mathbf{D}_{j,k} \widehat{\mathbf{h}}_{j,k}, \tag{17}$$

where

$$\mathbf{R}_{\mathbf{y}_j} = \mathbb{E}\{\mathbf{y}_j \mathbf{y}_j^H\} = \sum_{i=1}^{K} p_i \left(\widehat{\mathbf{h}}_{j,i} \widehat{\mathbf{h}}_{j,i}^H + \mathbf{C}_{j,i} \right) + \sigma^2 \mathbf{I}_N. \tag{18}$$

As in C-MMSE, the summation over all UEs in (18), makes (17) non-scalable. Two scalable alternatives are the LP-MMSE, given by

$$\mathbf{v}_{j,k}^{\text{LP-MMSE}} = p_k \overline{\mathbf{R}}_{\mathbf{y}_j}^{-1} \mathbf{D}_{j,k} \widehat{\mathbf{h}}_{j,k}, \tag{19}$$

where

$$\overline{\mathbf{R}}_{\mathbf{y}_j} = \sum_{i \in \mathcal{U}_j} p_i (\widehat{\mathbf{h}}_{j,i} \widehat{\mathbf{h}}_{j,i}^H + \mathbf{C}_{j,i}) + \sigma^2 \mathbf{I}_N, \tag{20}$$

in which the summation is over the UEs in $\mathcal{U}_j$, and the MR combiner, given by

$$\mathbf{v}_{j,k}^{\text{MR}} = \widehat{\mathbf{h}}_{j,k}. \tag{21}$$

The signal estimates in (16) are then sent to the CPU, which linearly combines these to obtain the final estimate

$$\hat{s}_k = \sum_{j=1}^{L} a_{j,k}^* \hat{s}_{j,k}. \tag{22}$$

We define $\mathbf{a}_k = \begin{bmatrix} a_{1,k} & \ldots & a_{L,k} \end{bmatrix}^T \in \mathbb{C}^L$ and

$$\mathbf{g}_{ki} = \begin{bmatrix} \mathbf{h}_{1,i}^H \mathbf{D}_{1,k} \mathbf{v}_{1,k} & \ldots & \mathbf{h}_{L,i}^H \mathbf{D}_{L,k} \mathbf{v}_{L,k} \end{bmatrix}^H \in \mathbb{C}^L, \tag{23}$$

and note that an achievable SE of UE k in the distributed operation is given by [17]

$$\mathrm{SE}_k^{(\mathrm{d})} = \frac{\tau_u}{\tau_c} \log_2 \left(1 + \mathrm{SINR}_k^{(\mathrm{d})} \right), \tag{24}$$

where

$$\mathrm{SINR}_k^{(\mathrm{d})} = \frac{p_k |\mathbf{a}_k^H \mathbb{E}\{\mathbf{g}_{kk}\}|^2}{\mathbf{a}_k^H (\mathbf{J}_k + \mathbf{N}_k) \mathbf{a}_k}, \tag{25}$$

with $\mathbf{J}_k = \sum_{i=1}^{K} p_i \mathbb{E}\{\mathbf{g}_{ki} \mathbf{g}_{ki}^H\} - p_k \mathbb{E}\{\mathbf{g}_{kk}\} \mathbb{E}\{\mathbf{g}_{kk}^H\}$, and $\mathbf{N}_k = \sigma^2 \mathtt{diag}\{\mathbb{E}\{\|\mathbf{D}_{j,k} \mathbf{v}_{j,k}\|^2\}\}_{j=1}^{L}$. The optimal (opt) weights that maximize (24) are given by [17]

$$\mathbf{a}_k^{\text{opt}} = p_k \left(\sum_{i=1}^{K} p_i \mathbb{E}\{\mathbf{g}_{ki} \mathbf{g}_{ki}^H\} + \mathbf{N}_k \right)^{\dagger} \mathbb{E}\{\mathbf{g}_{kk}\}, \tag{26}$$

where the pseudoinverse is used because the matrix can be singular, due to $\mathbf{D}_{j,k}$. Again, the summation over all UEs makes the computation non-scalable. A scalable alternative is to restrict the summation to the UEs in $\mathcal{S}_k$ to obtain the nearly optimal (n-opt) weights [17] given by

$$\mathbf{a}_k^{\text{n-opt}} = p_k \left(\sum_{i \in \mathcal{S}_k} p_i \mathbb{E}\{\mathbf{g}_{ki} \mathbf{g}_{ki}^H\} + \mathbf{N}_k \right)^{\dagger} \mathbb{E}\{\mathbf{g}_{kk}\}. \tag{27}$$

Remark 1. *The signal models in (2) and (7) assume a time-synchronized network. Methods to synchronize the AP clocks in time can be found in [10], [31]. Under this assumption, the models in (2) and (7) are sufficiently accurate for the performance analysis in this paper.*

III. Master-Assisted Distributed Uplink Operation (MADUO)

We propose a novel uplink operation for CFmMIMO networks, called *Master-Assisted Distributed Uplink Operation* (MADUO). In MADUO, each UE is assigned a *master AP* (MAP), which is responsible for its service. We denote the MAP of a UE k by l_k or l when the UE in question is easily inferred. Note that $l_k \in \mathcal{A}_k$. All other APs that serve UE k, i.e., the APs in the set $\mathcal{A}_k \setminus \{l_k\}$, are referred to as *additional serving APs* (ASAPs). MADUO is specified by the following steps:

I) AP $j \in [L]$ estimates the channels of its serving UEs $\{\mathbf{h}_{j,i} : i \in \mathcal{U}_j\}$;

II) ASAP $j \in \mathcal{A}_k \setminus \{l_k\}$ estimates the signal s_k of UE k using *any scalable* local combiner $\mathbf{D}_{j,k}\mathbf{v}_{j,k}$ (including but not limited to those defined in Section II-C). Note that $\mathbf{D}_{j,k} = \mathbf{I}_N$ for $j \in \mathcal{A}_k$, hence $\mathbf{D}_{j,k}\mathbf{v}_{j,k} = \mathbf{v}_{j,k}$, but that $\mathbf{D}_{j,k}$ is nevertheless included here to simplify the formulas;

III) ASAP $j \in \mathcal{A}_k \setminus \{l_k\}$ sends its estimate $\hat{s}_{j,k}$, given by (16), as well as its fused channel estimates $\hat{\gamma}^i_{j,k} \in \mathbb{C}$ for $i \in \mathcal{U}_j$, where

$$\hat{\gamma}^i_{j,k} = \mathbf{v}^H_{j,k}\mathbf{D}_{j,k}\mathbf{D}_{j,i}\widehat{\mathbf{h}}_{j,i}\,, \tag{28}$$

and the Hermitian product $\mu_{j,k} \in \mathbb{C}$, given by

$$\mu_{j,k} = \mathbf{v}^H_{j,k}\mathbf{D}_{j,k}\left(\sum_{i=1}^{K} p_i \mathbf{C}_{j,i} + \sigma^2 \mathbf{I}_N\right)\mathbf{D}_{j,k}\mathbf{v}_{j,k}\,, \tag{29}$$

to MAP l_k;

IV) MAP l_k estimates the data signal s_k of UE k using $\mathbf{y}_{l_k}$, $\hat{s}_{j,k}$, $\hat{\gamma}^i_{j,k}$ and $\mu_{j,k}$ received from the ASAPs.

To highlight the difference between the distributed operation and MADUO, suppose that UE k is served by MAP l_k and a set of ASAPs, and that MAP l_k executes the CPU tasks related to UE k. In distributed operation, all serving APs of UE k (all APs in the set $\mathcal{A}_k$) estimate s_k, and then the ASAPs send their signal estimates to MAP l_k. The final data detection is performed at MAP l_k, which uses *its own signal estimate* and the signal estimates of the ASAPs. In MADUO, first, *only* the ASAPs (the APs in the set $\mathcal{A}_k \setminus \{l_k\}$) estimate the signal of UE k and then send their signal estimates to MAP l_k. The data detection is, again, performed at the MAP l_k, but now it uses *its own local signals* $\mathbf{y}_{l_k}$ together with the received signal estimates from the ASAPs $\{\hat{s}_{j,k}\}_{j \neq l_k}$, the fused channel estimates $\hat{\gamma}^i_{j,k}$ in (28), and $\mu_{j,k}$ in (29).

Remark 2. *The MAP l_k of UE k must be selected from one of the APs in $\mathcal{A}_k$, which are expected to be in the proximity of UE k. Extensive simulations indicate that the specific MAP selection strategy results in negligible performance variations. Therefore, we simplify the MAP selection process by adopting a MAP assignment algorithm from the literature, as discussed in Section V.*

A. Master AP Receive Combining

The MAP l of UE k has access to its own local signals $\mathbf{y}_l$ and the signal estimates from the ASAPs $\{\hat{s}_{j,k}\}_{j \neq l}$. Note that (16) shows that the signal estimate of AP j is zero if it does not serve UE k. Hence, without loss of generality, MAP l has access to $\widetilde{\mathbf{y}}_l = \begin{bmatrix}\mathbf{y}_l^T & \hat{s}_{1,k} & \cdots & \hat{s}_{l-1,k} & \hat{s}_{l+1,k} & \cdots & \hat{s}_{L,k}\end{bmatrix}^T \in \mathbb{C}^{N+L-1}$, which can be written as

$$\widetilde{\mathbf{y}}_l = \sum_{i=1}^{K} \mathbf{z}_{l,ki} s_i + \widetilde{\mathbf{n}}_l\,, \tag{30}$$

where $\mathbf{z}_{l,ki} \in \mathbb{C}^{N+L-1}$ is given by

$$\mathbf{z}_{l,ki} = \begin{bmatrix}\mathbf{h}_{l,i}^T & \mathbf{g}_{ki,-l}^T\end{bmatrix}^T\,, \tag{31}$$

with $\mathbf{g}_{ki,-l} \in \mathbb{C}^{L-1}$ given by

$$\mathbf{g}_{ki,-l} = \begin{bmatrix}\gamma^i_{1,k} & \cdots & \gamma^i_{l-1,k} & \gamma^i_{l+1,k} & \cdots & \gamma^i_{L,k}\end{bmatrix}^T\,, \tag{32}$$

where $\gamma^i_{j,k} = \mathbf{v}^H_{j,k}\mathbf{D}_{j,k}\mathbf{h}_{j,i}$, and $\widetilde{\mathbf{n}}_l = \begin{bmatrix}\mathbf{n}_l^T & \mathbf{w}_k^T\end{bmatrix}^T \in \mathbb{C}^{N+L-1}$, with $\mathbf{w}_k \in \mathbb{C}^{L-1}$ given by

$$\mathbf{w}_k = \left[\mathbf{v}^H_{r,k}\mathbf{D}_{r,k}\mathbf{n}_r\right]_{r \in [L]\setminus\{l\}}\,, \tag{33}$$

i.e., the r-th entry of $\mathbf{w}_k$ is given by $\mathbf{v}^H_{r,k}\mathbf{D}_{r,k}\mathbf{n}_r$ for $r \in [L]\setminus l$. The receive combining vector of MAP l is denoted by $\widetilde{\mathbf{v}}_k \in \mathbb{C}^{N+L-1}$ and can be written as

$$\widetilde{\mathbf{v}}_k = \begin{bmatrix}\mathbf{v}_{l,k}^T & \boldsymbol{\alpha}_k^T\end{bmatrix}^T\,, \tag{34}$$

where $\mathbf{v}_{l,k} \in \mathbb{C}^N$ combines the local signals $\mathbf{y}_l$, and $\boldsymbol{\alpha}_k = \begin{bmatrix}\alpha_{1,k} & \cdots & \alpha_{l-1,k} & \alpha_{l+1,k} & \cdots & \alpha_{L,k}\end{bmatrix} \in \mathbb{C}^{L-1}$ combines the signal estimates. The final signal estimate for UE k is given by

$$\hat{s}_k = \widetilde{\mathbf{v}}_k^H\widetilde{\mathbf{y}}_l = \widetilde{\mathbf{v}}_k^H\sum_{i=1}^{K}\mathbf{z}_{l,ki}s_i + \widetilde{\mathbf{v}}_k^H\widetilde{\mathbf{n}}_l\,. \tag{35}$$

We aim to compute the receive combiner $\widetilde{\mathbf{v}}_k$ by maximizing the capacity lower bound known as SE.

Lemma 1. *An achievable SE of UE k in MADUO is*

$$SE_k = \frac{\tau_u}{\tau_c}\mathbb{E}\left\{\log_2\left(1 + SINR_k\right)\right\}\,, \tag{36}$$

where the expectation is with respect to channel realizations, and the instantaneous effective SINR is given by

$$SINR_k = \frac{p_k|\widetilde{\mathbf{v}}_k^H\widehat{\mathbf{z}}_{l,kk}|^2}{\widetilde{\mathbf{v}}_k^H\widetilde{\mathbf{B}}_k\widetilde{\mathbf{v}}_k}\,, \tag{37}$$

with $\widehat{\mathbf{z}}_{l,ki} \in \mathbb{C}^{N+L-1}$ given by

$$\widehat{\mathbf{z}}_{l,ki} = \begin{bmatrix}\widehat{\mathbf{h}}_{l,i}^T & \widehat{\mathbf{g}}_{ki,-l}^T\end{bmatrix}^T\,, \tag{38}$$

where $\widehat{\mathbf{g}}_{ki,-l} \in \mathbb{C}^{L-1}$ is given by

$$\widehat{\mathbf{g}}_{ki,-l} = \begin{bmatrix}\hat{\gamma}^i_{1,k} & \cdots & \hat{\gamma}^i_{l-1,k} & \hat{\gamma}^i_{l+1,k} & \cdots & \hat{\gamma}^i_{L,k}\end{bmatrix}^T\,. \tag{39}$$

Moreover, $\widetilde{\mathbf{B}}_k \in \mathbb{C}^{(N+L-1)\times(N+L-1)}$ *is given by*

$$\widetilde{\mathbf{B}}_k = \begin{bmatrix} \mathbf{R}_{\mathbf{y}_l} - p_k \widehat{\mathbf{h}}_{l,k}\widehat{\mathbf{h}}_{l,k}^H & \widehat{\mathbf{H}}_{l,-k}\mathbf{P}_{-k}\widehat{\mathbf{G}}_{k,-k}^H \\ \widehat{\mathbf{G}}_{k,-k}\mathbf{P}_{-k}\widehat{\mathbf{H}}_{l,-k}^H & \widehat{\mathbf{G}}_{k,-k}\mathbf{P}_{-k}\widehat{\mathbf{G}}_{k,-k}^H + \mathbf{F}_k \end{bmatrix}, \quad (40)$$

where $\widehat{\mathbf{H}}_{l,-k} \in \mathbb{C}^{N\times(K-1)}$ *is given by*

$$\widehat{\mathbf{H}}_{l,-k} = \begin{bmatrix} \widehat{\mathbf{h}}_{l,1} & \dots & \widehat{\mathbf{h}}_{l,k-1} & \widehat{\mathbf{h}}_{l,k+1} & \dots & \widehat{\mathbf{h}}_{l,K} \end{bmatrix}, \quad (41)$$

$\mathbf{P}_{-k} \in \mathbb{C}^{(K-1)\times(K-1)}$ *is given by*

$$\mathbf{P}_{-k} = diag\{p_i\}_{i\neq k}, \quad (42)$$

$\widehat{\mathbf{G}}_{k,-k} \in \mathbb{C}^{(L-1)\times(K-1)}$ *is given by*

$$\widehat{\mathbf{G}}_{k,-k} = \begin{bmatrix} \widehat{\mathbf{g}}_{k1,-l} \cdots \widehat{\mathbf{g}}_{k(k-1),-l} & \widehat{\mathbf{g}}_{k(k+1),-l} \cdots \widehat{\mathbf{g}}_{kK,-l} \end{bmatrix}, \quad (43)$$

and $\mathbf{F}_k \in \mathbb{C}^{(L-1)\times(L-1)}$ *is given by*

$$\mathbf{F}_k = diag\{\mu_{j,k}\}_{j\neq l}. \quad (44)$$

Proof: The proof is given in Appendix A. ∎

Henceforth, we refer to (36) as the *achievable SE*. It is noted that the instantaneous effective SINR in (37) has the form of a generalized Rayleigh quotient, which is easy to maximize. Hence, the receive combiner that maximizes the instantaneous effective SINR in (37), and thus the achievable SE in (36), is given by the following corollary:

Corollary 1. *The effective SINR in* (37) *is maximized by*

$$\widetilde{\mathbf{v}}_k = p_k \widetilde{\mathbf{B}}_k^{-1} \widehat{\mathbf{z}}_{l,kk}. \quad (45)$$

Proof: The proof follows from [17, Lem. 3.7]. ∎

It is noted that Rayleigh quotients are scale-invariant [17]. It can be proved that the combiner

$$\widetilde{\mathbf{v}}_k = \left(\widetilde{\mathbf{B}}_k + p_k \widehat{\mathbf{z}}_{l,kk}\widehat{\mathbf{z}}_{l,kk}^H\right)^{-1} \widehat{\mathbf{z}}_{l,kk} = \mathbf{B}_k^{-1}\widehat{\mathbf{z}}_{l,kk}, \quad (46)$$

where $\mathbf{B}_k$ can be written as

$$\mathbf{B}_k = \begin{bmatrix} \mathbf{R}_{\mathbf{y}_l} & \widehat{\mathbf{H}}_l\mathbf{P}\widehat{\mathbf{G}}_k^H \\ \widehat{\mathbf{G}}_k\mathbf{P}\widehat{\mathbf{H}}_l^H & \widehat{\mathbf{G}}_k\mathbf{P}\widehat{\mathbf{G}}_k^H + \mathbf{F}_k \end{bmatrix}, \quad (47)$$

with $\mathbf{P} = \mathtt{diag}\{p_i\} \in \mathbb{C}^{K\times K}$ and $\widehat{\mathbf{G}}_k \in \mathbb{C}^{(L-1)\times K}$ given by

$$\widehat{\mathbf{G}}_k = \begin{bmatrix} \widehat{\mathbf{g}}_{k1,-l} & \dots & \widehat{\mathbf{g}}_{kK,-l} \end{bmatrix},$$

is a scaled version of the combiner in (45), hence also a maximizer of (37), and the minimizer of the conditional mean squared error $\mathbb{E}\left\{|s_k - \widetilde{\mathbf{v}}_k^H\widetilde{\mathbf{y}}_l|^2 \middle| \{\widehat{\mathbf{z}}_{l,ki}\}_{i=1}^K, \{\mu_{j,k}\}_{j\neq l}\right\}$. The former follows directly from [17, Lem. 3.7], and the latter can be proved using the same methodology as in [4, Proofs of Cor. 4.2-4.3]. Note that the combiner in (46) is non-scalable by Definition 1, since the top-left block in (47) involves a summation over all UEs, the off-diagonal blocks depend on all channel estimates of MAP l, and the bottom-right block involves a summation over all UEs, due to $\mathbf{F}_k$, defined in (44), and $\mu_{j,k}$, defined in (29). To make the combiner scalable, we replace $\mathbf{B}_k$ in (47) with $\overline{\mathbf{B}}_k \in \mathbb{C}^{(N+L-1)\times(N+L-1)}$, given by

$$\overline{\mathbf{B}}_k = \begin{bmatrix} \overline{\mathbf{R}}_{\mathbf{y}_l} & \widehat{\overline{\mathbf{H}}}_l\mathbf{P}\widehat{\mathbf{G}}_k^H \\ \widehat{\mathbf{G}}_k\mathbf{P}\widehat{\overline{\mathbf{H}}}_l^H & \widehat{\mathbf{G}}_k\mathbf{P}\widehat{\mathbf{G}}_k^H + \overline{\mathbf{F}}_k \end{bmatrix}, \quad (48)$$

where $\overline{\mathbf{R}}_{\mathbf{y}_l}$ is given by (20) for $j = l$, $\widehat{\overline{\mathbf{H}}}_l \in \mathbb{C}^{N\times K}$ is given by

$$\widehat{\overline{\mathbf{H}}}_l = \begin{bmatrix} \mathbf{D}_{l,1}\widehat{\mathbf{h}}_{l,1} & \dots & \mathbf{D}_{l,K}\widehat{\mathbf{h}}_{l,K} \end{bmatrix}, \quad (49)$$

and $\overline{\mathbf{F}}_k \in \mathbb{C}^{(L-1)\times(L-1)}$ given by

$$\overline{\mathbf{F}}_k = \mathtt{diag}\{\bar{\mu}_{j,k}\}_{j\neq l}, \quad (50)$$

with

$$\bar{\mu}_{j,k} = \mathbf{v}_{j,k}^H\mathbf{D}_{j,k}\left(\sum_{i\in\mathcal{U}_j} p_i\mathbf{C}_{j,i} + \sigma^2\mathbf{I}_N\right)\mathbf{D}_{j,k}\mathbf{v}_{j,k}, \quad (51)$$

where the summation is over the served UEs $i \in \mathcal{U}_j$ instead of all K UEs in the networks in Step III. Henceforth, we assume that the MAPs use the scalable combiner given by

$$\overline{\mathbf{v}}_k = \overline{\mathbf{B}}_k^{-1}\widehat{\mathbf{z}}_{l,kk}. \quad (52)$$

Finally, it is noted that if a UE k is only served by its MAP, i.e., $\mathcal{A}_k = \{l\}$, only the top-left block of (48) is non-zero. This condition reduces (52) to the LP-MMSE combiner in (19).

B. Imperfect Statistical CSI

In Section II-B, we have assumed that the APs have perfect knowledge of the covariance matrices $\{\mathbf{R}_{j,i} : i \in \mathcal{P}_k\}$ used for channel estimation in (4) and (5), which allowed us to derive the achievable SE for MADUO (see Lemma 1). In the case of imperfect knowledge of these covariance matrices, we resort to the *use-and-then-forget* (UatF) lower bound of the capacity, which assumes that the MAP forgets the channel estimates before the final data detection [4].

Lemma 2. *The UatF bound for UE* k *in MADUO is*

$$SE_k^{UatF} = \frac{\tau_u}{\tau_c}\log_2\left(1 + SINR_k^{UatF}\right), \quad (53)$$

where $SINR_k^{UatF}$ *is given by* (55) *at the top of the next page, with* $\mathbf{T}_k \in \mathbb{C}^{(N+L-1)\times(N+L-1)}$ *given by*

$$\mathbf{T}_k = \sigma^2 blkdiag\left(\mathbf{I}_N, diag\left\{\|\mathbf{D}_{j,k}\mathbf{v}_{j,k}\|^2\right\}_{j\neq l}\right). \quad (54)$$

Proof: In (35), we add and subtract $\mathbb{E}\{\widetilde{\mathbf{v}}_k^H\mathbf{z}_{l,kk}\}s_k$ and use [17, Lem. 3.3] to derive the final result. ∎

It is noted that the maximizer of (53) is difficult to obtain due to the expectations in the SINR expression in (55). Moreover, the UatF bound is less tight than the bound in (36), but can be used for any receive combiner and does not require the use of MMSE channel estimation [4].

C. The Two Special Cases

In the sequel, we examine two special cases of MADUO. In the first case, the ASAPs use the LP-MMSE local combiner, defined in (19), and in the second case, the ASAPs use the MR local combiner, defined in (21). In both cases, the MAP uses the combiner in (52). The first case is denoted by MADUO$_{\text{MMSE}}$ and the second case is denoted by MADUO$_{\text{MR}}$.

$$\mathrm{SINR}_k^{\mathrm{UatF}} = \frac{p_k \left| \mathbb{E}\left\{ \widetilde{\mathbf{v}}_k^H \mathbf{z}_{l,kk} \right\} \right|^2}{\sum\limits_{i=1}^{K} p_i \mathbb{E}\left\{ \left| \widetilde{\mathbf{v}}_k^H \mathbf{z}_{l,ki} \right|^2 \right\} - p_k \left| \mathbb{E}\left\{ \widetilde{\mathbf{v}}_k^H \mathbf{z}_{l,kk} \right\} \right|^2 + \mathbb{E}\left\{ \widetilde{\mathbf{v}}_k^H \mathbf{T}_k \widetilde{\mathbf{v}}_k \right\}}, \tag{55}$$

D. Computational Complexity

We assess the computational complexity of the two special cases of MADUO, using the methodology described in [4, App. B.1.1]. Specifically, we count the number of complex multiplications required per coherence block to compute the combining vector of an arbitrary UE k, which is served by $|\mathcal{A}_k| - 1$ ASAPs and one MAP. We denote the computational complexity for the receive combiner in one ASAP by $\mathcal{C}(\mathbf{v})$, where $\mathbf{v}$ can be the LP-MMSE or MR combiner. For LP-MMSE, the computational complexity can be found in [17, Table 5.3]. For MR, $\mathcal{C}(\mathbf{v}^{\mathrm{MR}}) = 0$. The computational complexity for MADUO is

$$\Lambda_k(\mathbf{v}) + \Gamma_k \,, \tag{56}$$

where

$$\Gamma_k = N \sum_{j \in \mathcal{A}_k \setminus \{l_k\}} |\mathcal{U}_j| + \frac{N^2+N}{2} |\mathcal{U}_{l_k}| + \sum_{j \in \mathcal{A}_k \setminus \{l_k\}} \sum_{\substack{q \in \mathcal{A}_k \setminus \{l_k\} \\ q \geq j}} |\mathcal{U}_{j,q}| + N \sum_{j \in \mathcal{A}_k \setminus \{l_k\}} |\mathcal{U}_{l_k,j}| \,, \tag{57}$$

where $\mathcal{U}_{j,q} = \mathcal{U}_j \cap \mathcal{U}_q$ is the set of UEs served by both APs j and q ($\mathcal{U}_{l_k,j}$ is defined similarly), and

$$\Lambda_k(\mathbf{v}) = (N + |\mathcal{A}_k| - 1)^2 + (|\mathcal{A}_k| - 1)(N^2 + N + \mathcal{C}(\mathbf{v})) + \frac{(N + |\mathcal{A}_k| - 1)^3 - N - |\mathcal{A}_k| + 1}{3} \,, \tag{58}$$

which depends on the choice of the receive combiner at the ASAPs. For details, see Appendix B.

E. Fronthaul Overhead

We quantify the fronthaul overhead of an operation by counting the number of complex scalars sent through fronthaul in a single coherence block. We denote the fronthaul overhead of an operation by $\texttt{front}(\cdot)$. Assume that AP j serves $|\mathcal{U}_j|$ UEs. We denote the set of UEs for which AP j acts as MAP by $\mathcal{U}_j^{\mathrm{MAP}}$. Hence, AP j is an ASAP for $|\mathcal{U}_j| - |\mathcal{U}_j^{\mathrm{MAP}}|$ UEs. In Step III of MADUO, AP j sends τ_u signal estimates, $|\mathcal{U}_j|$ fused channel estimates, and the Hermitian product (one scalar) in (51) to the MAPs of $|\mathcal{U}_j| - |\mathcal{U}_j^{\mathrm{MAP}}|$ UEs in one coherence block. Thus, the fronthaul overhead is

$$\texttt{front}\,(\mathrm{MADUO}) = \sum_{j=1}^{L} \left(\tau_u + |\mathcal{U}_j| + 1\right) \left(|\mathcal{U}_j| - |\mathcal{U}_j^{\mathrm{MAP}}|\right) \tag{59}$$

complex scalars per coherence block. The fronthaul overhead for all operations is presented in Table I.

TABLE I: Number of complex scalars exchanged through fronthaul links.

Operation	Fronthaul Overhead
Centralized	$\tau_c N L$
Distributed	$\tau_u \sum_{j=1}^{L} |\mathcal{U}_j|$
MADUO	$\sum_{j=1}^{L} \left(\tau_u + |\mathcal{U}_j| + 1\right) \left(|\mathcal{U}_j| - |\mathcal{U}_j^{\mathrm{MAP}}|\right)$

We compare the fronthaul overhead of MADUO with that of the distributed operation using the formulas in Table I. Further discussion for the centralized operation is presented in Section V-B. We start by subtracting $\texttt{front}$(Distributed) from $\texttt{front}$(MADUO) to obtain

$$\texttt{front}\left(\mathrm{MADUO}^{\mathrm{scl}}\right) - \texttt{front}\,(\mathrm{Distributed}) = \sum_{j=1}^{L} \left(|\mathcal{U}_j|^2 + |\mathcal{U}_j| - |\mathcal{U}_j||\mathcal{U}_j^{\mathrm{MAP}}|\right) - \tau_u(K+1) \,, \tag{60}$$

where we have used $\sum_{j=1}^{L} |\mathcal{U}_j^{\mathrm{MAP}}| = K$, since each UE is appointed one MAP. From (60), we deduce that MADUO has a lower fronthaul overhead than the distributed operation when the DCC sufficiently restricts the number of served UEs of each AP such that the first term of the right-hand-side of (60) is smaller than the second. Furthermore, equation (60) indicates that for larger τ_u (in more stationary environments) and larger K (more UEs), $\texttt{front}$(MADUO) is expected to be significantly smaller than $\texttt{front}$ (Distributed). Numerical experiments and comparisons are presented in Section V-B.

Finally, note that the distributed operation, described in Section II-C2, can readily be implemented on a MAP, which will perform the tasks of the CPU. In that case, the fronthaul overhead for distributed operation is $\sum_{j=1}^{L} \tau_u(|\mathcal{U}_j| - |\mathcal{U}_j^{\mathrm{MAP}}|)$, which is smaller than that of MADUO.

IV. A Statistical Cooperation Framework for MADUO

As shown in Section III-E, the fronthaul overhead for MADUO, given by (59), can be significant. This observation motivates the development of a cooperation framework to control the fronthaul overhead in MADUO. To this end, this section introduces a statistical cooperation framework for MADUO and explores the trade-off between fronthaul overhead and SE performance. The main idea is to control the amount of information sent from the ASAPs to the MAP, thereby controlling fronthaul overhead, and to study the impact on SE performance. We start by assuming that the ASAPs send only signal estimates to the MAP, and hence, the conditional MSE expression becomes

$$\mathrm{MSE}\left(\widetilde{\mathbf{v}}_k; \widehat{\mathbf{H}}_l\right) = \mathbb{E}\left\{ |s_k - \widetilde{\mathbf{v}}_k^H \widetilde{\mathbf{y}}_l|^2 \middle| \widehat{\mathbf{H}}_l \right\} . \tag{61}$$

Corollary 2. *The receive combiner that minimizes (61) is given by*

$$\widetilde{\mathbf{v}}_k^{\textit{level-x}} = \breve{\mathbf{B}}_k^{-1} \begin{bmatrix} \widehat{\mathbf{h}}_{l,k} \\ \mathbb{E}\{\mathbf{g}_{kk,-l}\} \end{bmatrix}, \tag{62}$$

where $\mathbf{g}_{kk,-l}$ given by (32) for $i = k$, and $\breve{\mathbf{B}}_k \in \mathbb{C}^{(N+L-1)\times(N+L-1)}$ is given by

$$\breve{\mathbf{B}}_k = \begin{bmatrix} \mathbf{R}_{\mathbf{y}_l} & \widehat{\mathbf{H}}_l \mathbf{P} \mathbb{E}\{\mathbf{G}_k^H\} \\ \mathbb{E}\{\mathbf{G}_k\} \mathbf{P} \widehat{\mathbf{H}}_l^H & \mathbb{E}\{\mathbf{G}_k \mathbf{P} \mathbf{G}_k^H\} + \mathbb{E}\{\mathbf{F}_k\} \end{bmatrix}, \tag{63}$$

with $\mathbf{R}_{\mathbf{y}_l}$ given by (18) for $j = l$.

Proof: The proof follows from the standard linear MMSE estimation [17, Sec. 3.1], but the expectation is only conditioned on the local channel estimates of MAP l $\widehat{\mathbf{H}}_l$. Since $\widehat{\mathbf{g}}_{kk,-l}$, $\widehat{\mathbf{G}}_k$, and $\mathbf{F}_k$ are independent of $\widehat{\mathbf{H}}_l$, the expectation is taken over all the remaining associated sources of randomness. ∎

We study three cases, depending on the information exchanged between ASAPs and the MAP, thus defining three cooperation levels. At all levels, the MAP is assumed to receive the ASAPs' signal estimates $\hat{s}_{j,k}$. Furthermore, we follow the same methodology as in Section III, i.e., we derive non-scalable combiners which can then straightforwardly become scalable by replacing the appropriate blocks in (63) with those in (48).

A. Cooperation Level 1

In Level 1, ASAP j sends fused instantaneous CSI of all UEs $\{\hat{\gamma}_{j,k}^i\}_{i\in\mathcal{U}_j}$ and the Hermitian product in (29) to the MAP, corresponding to the original MADUO. Since the MAP receives this additional information, all expectations in (61)–(63) are conditioned on $\{\widehat{\mathbf{z}}_{l,ki}\}_{i=1}^K$ and $\{\mu_{j,k}\}_{j\neq l}$. Thus, $\breve{\mathbf{B}}_k$ in (63) reduces to $\mathbf{B}_k$ in (47) and the combiner in (62) reduces to the combiner in (46). Hence, in Level 1, $\breve{\mathbf{B}}_k = \mathbf{B}_k$, $\widetilde{\mathbf{v}}_k^{\text{level-1}} = \widetilde{\mathbf{v}}_k$, and the fronthaul overhead is given by (59).

B. Cooperation Level 2

In Level 2, ASAP j sends fused instantaneous CSI of all UEs $\{\hat{\gamma}_{j,k}^i\}_{i\in\mathcal{U}_j}$ but ***not*** the Hermitian product in (29). Since the MAP receives this additional information, all expectations in (61)–(63) are conditioned on $\{\widehat{\mathbf{z}}_{l,ki}\}_{i=1}^K$. Hence, in Level 2, the matrix $\breve{\mathbf{B}}_k$ in (63) becomes

$$\breve{\mathbf{B}}_k = \begin{bmatrix} \mathbf{R}_{\mathbf{y}_l} & \widehat{\mathbf{H}}_l \mathbf{P} \widehat{\mathbf{G}}_k^H \\ \widehat{\mathbf{G}}_k \mathbf{P} \widehat{\mathbf{H}}_l^H & \widehat{\mathbf{G}}_k \mathbf{P} \widehat{\mathbf{G}}_k^H + \mathbb{E}\{\mathbf{F}_k\} \end{bmatrix} \tag{64}$$

and the combiner in (62) becomes $\widetilde{\mathbf{v}}_k^{\text{level-2}} = \breve{\mathbf{B}}_k^{-1} \widehat{\mathbf{z}}_{l,kk}$, with $\breve{\mathbf{B}}_k$ given by (64). In that case, ASAP j needs to send the expected value of the Hermitian product in (29) to the MAP, to compute $\mathbb{E}\{\mathbf{F}_k\} = \text{diag}\{\mathbb{E}\{\mu_{j,k}\}\}_{j\neq l}$ in (64). However, the statistics change much more slowly than instantaneous CSI. In particular, the statistics remain unchanged for any given location realization of APs and UEs, and hence can be shared with minimal associated fronthaul overhead [32]. Therefore, the fronthaul overhead in Level 2 is given by

$$\sum_{j=1}^{L} (\tau_u + |\mathcal{U}_j|) \left(|\mathcal{U}_j| - |\mathcal{U}_j^{\text{MAP}}| \right),$$

where we have removed the terms associated with the exchange of the Hermitian product from (59).

C. Cooperation Level 3

In Level 3, ASAP j does not send any instantaneous CSI to the MAP. Hence, $\breve{\mathbf{B}}_k$ and the combiner $\widetilde{\mathbf{v}}_k^{\text{level-3}}$ are given by (63) and (62), respectively. In that case, ASAP j also needs to send the statistical CSI for its serving UEs $i \in \mathcal{U}_j$, and the expected value of the Hermitian product $\mu_{j,k}$ in (29) to the MAP. Therefore, the fronthaul overhead in Level 3 is given by

$$\sum_{j=1}^{L} \tau_u \left(|\mathcal{U}_j| - |\mathcal{U}_j^{\text{MAP}}| \right),$$

where we have removed the terms associated with the instantaneous CSI exchange and the Hermitian product from (59).

From Level 1 to Level 3, the MAP uses less instantaneous CSI and the associated fronthaul overhead decreases. Hence, the SE performance is expected to drop from Level 1 to Level 3, while the fronthaul requirements decrease. Numerical experiments are presented Section V-B.

V. Numerical Experiments

In this section, we compare the two special cases of MADUO described in Section III-C with centralized and distributed operation in terms of SE performance, fronthaul overhead, and computational complexity, using numerical simulations. For centralized operation, we consider P-MMSE [15] combining, and for the distributed operation, we consider LP-MMSE [17].

We consider a CFmMIMO network with L APs, each equipped with $N = 4$ antennas, and a different number of UEs (K) uniformly distributed within a simulation square of area 2×2 km^2. The network has a wrap-around topology, as explained in [17, Sec. 5.3]. The APs are placed 10 meters above the UEs. The DCC, MAP assignment, and pilot assignment are jointly implemented as described in [15, Sec. V.A]. Specifically, UE k selects the AP with the largest large-scale fading coefficient $\beta_{l,k} = \text{tr}\left(\mathbf{R}_{l,k}\right)/N$ as its MAP. The MAP assigns the pilot signal with the least pilot contamination to UE k, and the neighboring APs decide whether to join it in serving UE k based on the channel conditions of UE k and the pilot signal assigned to it by the MAP. The APs serve up to τ_p UEs, we set $\tau_c = 200$, $\tau_p = 10$, $p = p_k = 100$ mW, and we adopt the same propagation model as in [4, Section 4.1.3], but with a minimum distance of 10 meters, since the APs are 10 meters above the UEs.

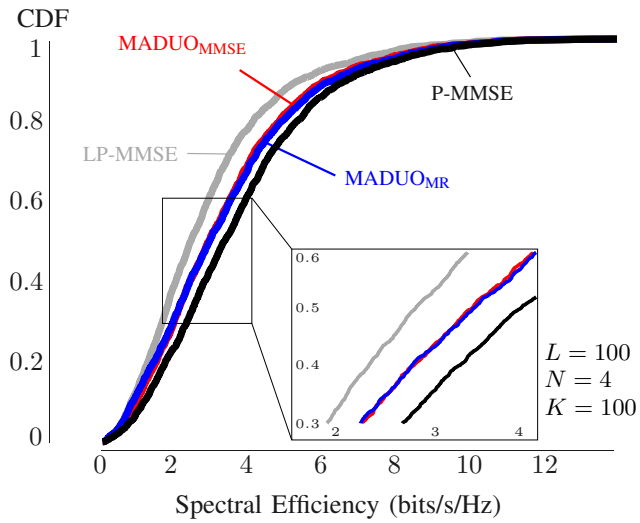

Fig. 1: CDF of the per-UE SE for $L = 100$, $N = 4$, and $K = 100$. The special cases of MADUO closely approach the performance of the centralized operation.

A. Spectral Efficiency

In Fig. 1, we plot the cumulative distribution function (CDF) of the per-UE SE. We compare the special cases of MADUO with the centralized and distributed operations, for $L = 100$, $N = 4$ and $K = 100$. For centralized operation, we use the achievable SE expression of (10), and for distributed operation, we use (24). It is evident that MADUO$_{\text{MMSE}}$ and MADUO$_{\text{MR}}$ outperform the distributed operation and closely approach the centralized operation. Interestingly, MADUO$_{\text{MR}}$ eventually outperforms MADUO$_{\text{MMSE}}$ for 50% of the UEs. It is noted that the region where MADUO$_{\text{MR}}$ performs worse than MADUO$_{\text{MMSE}}$ corresponds to the UEs that experience poor SINR conditions, since their achievable SEs are at the low end of the graph. From this, we conclude that MADUO$_{\text{MR}}$ performs better than MADUO$_{\text{MMSE}}$ for UEs with good SINR conditions and performs worse than MADUO$_{\text{MMSE}}$ for UEs with poor SINR conditions.

In Fig. 2, the average SE for varying N (left), K (middle), and L (right) are shown. From the left plot, we observe that, as N increases, the special cases of MADUO closely follow the centralized operation, with a small but increasing gap, and eventually approach the performance of distributed operation. This gap is attributed to $(N + |\mathcal{A}_k| - 1)/N \overset{N\to\infty}{\to} 1$, which indicates that the contribution of the $|\mathcal{A}_k| - 1$ ASAPs is dominated by the contribution of the N-dimensional locally received signal of the MAP, hence reducing the benefit of including the $|\mathcal{A}_k|-1$ signal estimates at the MMSE estimation of the MAP. Moreover, MADUO$_{\text{MMSE}}$ performs better than MADUO$_{\text{MR}}$ (on average) as N increases. This observation is attributed to the greater capacity of APs to cancel interference when equipped with more antennas and using MMSE-based combiners [4]. The distributed operation demonstrates significantly inferior performance compared to the special cases of MADUO and the centralized operation. From the plot on the right, we conclude that the special cases of MADUO, again, closely follow centralized operation as L increases. In this scenario, MADUO$_{\text{MR}}$ performs better than MADUO$_{\text{MMSE}}$ as L grows. As explained before, MADUO$_{\text{MR}}$ is expected to perform better than MADUO$_{\text{MMSE}}$ for UEs with good SINR conditions. The growing number of APs ensures that almost all UEs in the network experience good SINR conditions, since an arbitrary UE is located in the proximity of more APs as L increases. This is the reason for the improved performance of MADUO$_{\text{MR}}$ compared to MADUO$_{\text{MMSE}}$. The middle plot shows that as K increases, the performance of all operations degrades due to increased interference. The special cases of MADUO, again, follow the centralized operation and significantly outperform the distributed operation. In this scenario, MADUO$_{\text{MMSE}}$ eventually outperforms MADUO$_{\text{MR}}$, because the UEs are subject to more interference as K increases, which deteriorates the SINR conditions of all UEs.

We further investigate the reasons for the improved performance of MADUO$_{\text{MR}}$ compared to MADUO$_{\text{MMSE}}$ in certain scenarios, by considering a very simple setup with $K = L = 2$, $N = 4$, $\sigma = 1$, $\mathbf{h}_{j,i} \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}_N, \mathbf{I}_N)$, and perfect CSI at both APs. Both UEs are randomly assigned the same MAP, and the other AP is the unique ASAP for both UEs. Note that the random assignment of the MAP is justified because $\beta_{j,k} = \beta = 1$. Fig. 3 shows the probability density function (PDF) of the SINR for both MADUO$_{\text{MMSE}}$ and MADUO$_{\text{MR}}$. We also plot the SINR PDF for the centralized operation using P-MMSE for reference. We observe that the PDF of MADUO$_{\text{MR}}$ has a more compact support compared to MADUO$_{\text{MMSE}}$, despite the larger mean value of MADUO$_{\text{MMSE}}$. This observation reveals that MADUO$_{\text{MMSE}}$ tends to have a PDF with long tails, whereas MADUO$_{\text{MR}}$ tends to have a PDF with compact support. We attribute the improved performance of MADUO$_{\text{MR}}$ over MADUO$_{\text{MMSE}}$ in high-SINR scenarios to the long tails of MADUO$_{\text{MMSE}}$ and the compact support of MADUO$_{\text{MR}}$.

In Fig. 4, we plot the per-UE achievable SE (36) and the UatF bound (53) for MADUO$_{\text{MMSE}}$ and MADUO$_{\text{MR}}$, for $L = 100$, $N = 4$, and $K = 40$. The plot shows that the UatF bound closely follows the achievable SE, which means that it is roughly as tight as the achievable SE for both MADUO$_{\text{MMSE}}$ and MADUO$_{\text{MR}}$. For the centralized operation, the tightness of the UatF bound was established in [17]. For the distributed operation, the state-of-the-art bound is the UatF bound, which is given by (24). Hence, we conclude that, with imperfect statistical CSI, the UatF bound accurately reflects the performance of the operations, as it yields the same comparative results as the achievable SE.

In Fig. 5, we plot the average SE for all cooperation levels of MADUO, for varying N, $L = 80$, and $K = 40$. It is evident that Levels 1 and 2 perform almost identically and significantly outperform Level 3, which slightly outperforms distributed operation. However, Level 3 is expected to reduce fronthaul overhead compared to Levels 1 and 2. This behavior was expected, since in Level 3, the MAP uses almost the same amount of information as the distributed operation.

B. Fronthaul Overhead

In Fig. 6, we plot the number of complex scalars sent through the fronthaul links in each coherence block over the

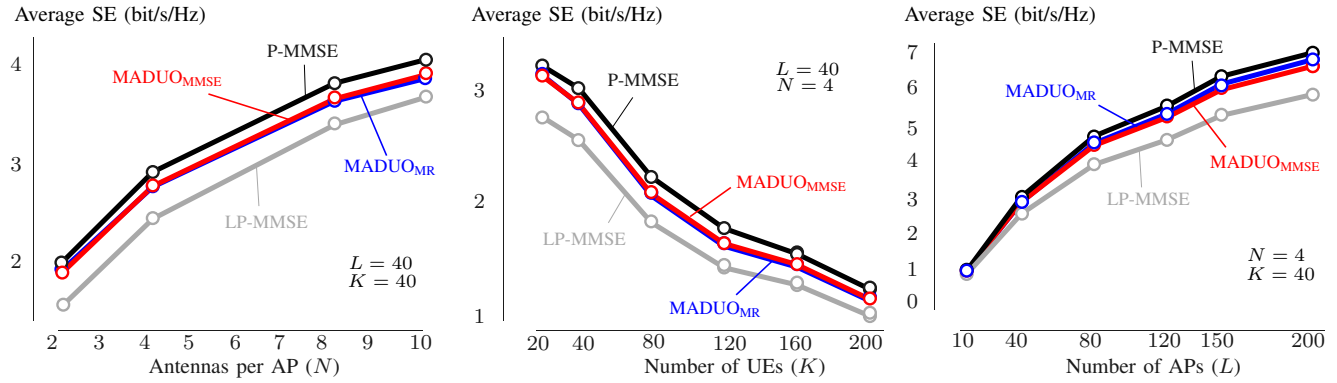

Fig. 2: Average SE for varying N (left), K (middle), and L (right). In all scenarios, the special cases of MADUO closely follow the performance of the centralized operation and outperform the distributed operation.

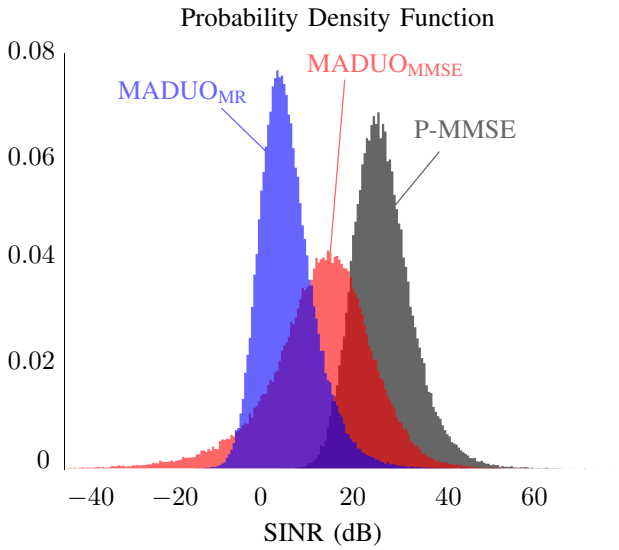

Fig. 3: PDF of the SINR for MADUO$_{\text{MMSE}}$, MADUO$_{\text{MR}}$, and centralized operation with P-MMSE combining, with $K = L = 2$, $N = 4$, $\sigma = 1$, $\mathbf{h}_{j,i} \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}_N, \mathbf{I}_N)$, and perfect CSI at the APs. MADUO$_{\text{MMSE}}$ exhibits large SINR variations.

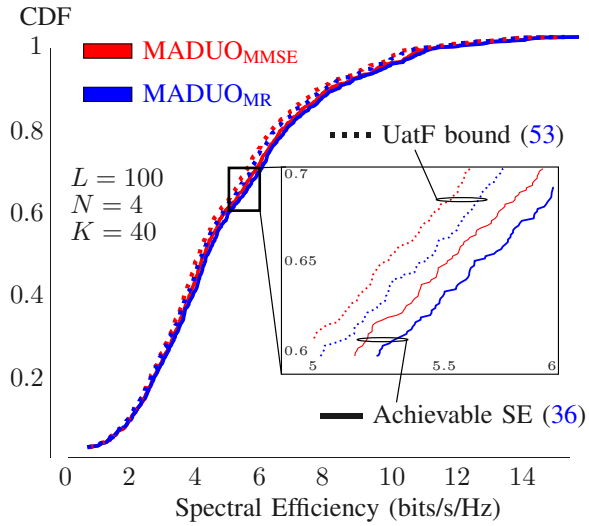

Fig. 4: CDF of the per-UE achievable SE and UatF bound of MADUO$_{\text{MMSE}}$ and MADUO$_{\text{MR}}$, for $L = 100$, $N = 4$, and $K = 40$. The UatF bound closely follows the achievable SE.

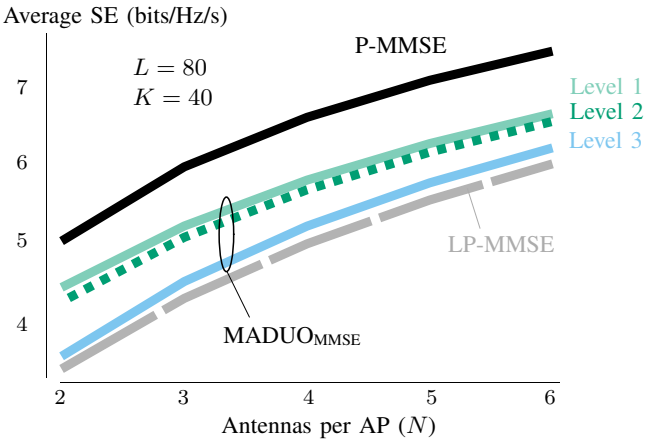

Fig. 5: Average SE for all cooperation levels of MADUO$^{\text{scl}}_{\text{MMSE}}$ and varying N. Levels 1 and 2 outperform Levels 3.

network for $L = 100$, $N = 4$, and varying K. For centralized and distributed operation, we use the fronthaul overhead expressions presented in Table I. We focus on MADUO$_{\text{MMSE}}$. It is evident that MADUO requires significantly less fronthaul overhead than distributed operation, with an increasing gap as K increases. Interestingly, MADUO exhibits a steady decrease with K. To explain this behavior, we note that as K increases, an arbitrary AP l becomes the MAP of more UEs, since it is located in the proximity of more UEs, i.e., $|\mathcal{U}_j^{\text{MAP}}|$ increases for all j. From (59), we see that this phenomenon decreases the fronthaul overhead. We also note that we have used K values comparable to L in this experiment. Hence, for K comparable to L, the fronthaul overhead decreases for MADUO as K grows. We highlight that realistic scenarios for CFmMIMO networks assume $L < K < LN$ [7]. From Fig. 6, we conclude that in realistic scenarios ($K > L$), MADUO significantly reduces the fronthaul overhead compared to the distributed operation. Furthermore, Levels 1 and 2 require almost identical fronthaul overhead, but more than Level 3. We report an average 5% relative decrease of Level 3 compared to Levels 1 and 2, which remains constant as K varies, an average of 6% relative decrease of Levels 1 and 2 compared to distributed operation, which increases with K, and an average of 11%

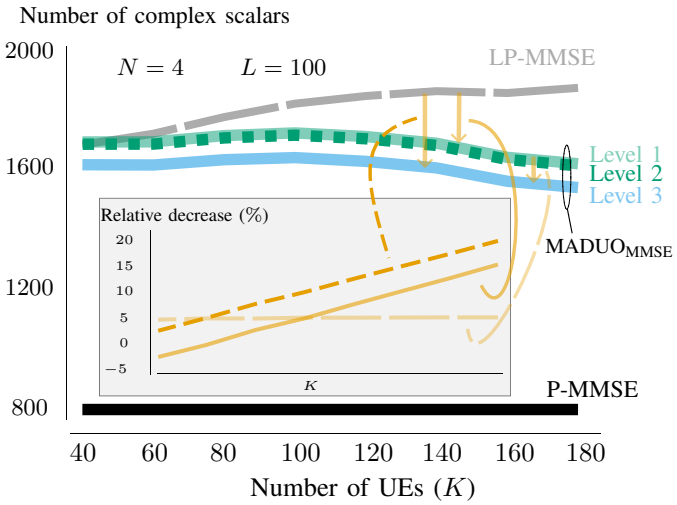

Fig. 6: Fronthaul overhead for varying K. All cooperation levels of MADUO have a lower fronthaul overhead than distributed operation.

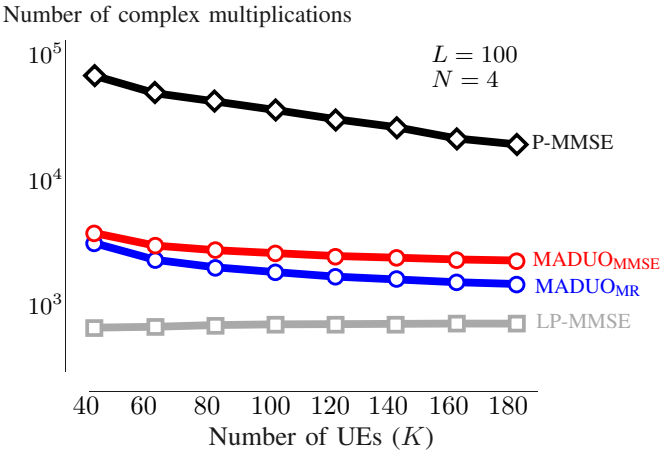

Fig. 7: Number of complex multiplications for one UE per coherence block for varying K. MADUO needs considerably fewer computations than the centralized operation.

relative decrease of Level 3 compared to distributed operation, which also increases with K.

C. Computational Complexity

In Fig. 7, we plot the number of complex multiplications to estimate the signal of one UE per coherence block, for varying numbers of UEs (K). For the centralized and distributed operation, we use the computational complexity expressions from [17, Table 5.3]. It is evident that both special cases of MADUO have a significantly smaller computational complexity than the centralized operation. Interestingly, both for MADUO$_{\text{MMSE}}$ and MADUO$_{\text{MR}}$ the complexity decreases as K increases, which can be explained by the decrease of $|\mathcal{A}_k|$ with growing K [17]. As shown in (58), the complexity of MADUO$_{\text{MMSE}}$ and MADUO$_{\text{MR}}$ scales cubically with $|\mathcal{A}_k|$, which leads to the steady decrease, observed in Fig. 7. Moreover, we observe that the complexity of MADUO$_{\text{MR}}$ is consistently smaller than that of MADUO$_{\text{MMSE}}$, thanks to the use of the simpler MR combiner, which does not require any computations (except for the computations required for channel estimation).

VI. Conclusion

In this paper, we have proposed a novel distributed uplink operation for CFmMIMO networks, which we call *Master-Assisted Distributed Uplink Operation* (MADUO). In MADUO, a MAP and a set of ASAPs are assigned to each UE. We have identified two special cases of MADUO, based on the choice of the receive combiners at the ASAPs. Both special cases outperform the distributed operation and closely follow the performance of the centralized operation. Interestingly, the use of MR combining at the ASAPs proved beneficial compared with MMSE-based combining for UEs with high SINR. Moreover, we have proposed a statistical cooperation framework that controls the trade-off between SE performance and fronthaul overhead, as the instantaneous CSI exchange between the ASAPs and the MAP can be controlled. The main conclusion of this work is that MADUO can achieve near-centralized performance with lower fronthaul overhead than in distributed operation in realistic scenarios (where there are more UEs than APs in the network) and with fewer computations than in centralized operation.

Appendix A
Proof of Lemma 1

Starting from (30), notice that the signal model is the same as in (8), which is used in centralized operation. Hence, the SINR and SE expressions for centralized operation, given in (11) and (10), respectively, can be readily adapted to MADUO. We define

$$\widetilde{\mathbf{Z}}_{ki} = \mathbb{E}\{\widetilde{\mathbf{z}}_{l,ki}\widetilde{\mathbf{z}}_{l,ki}^H\} = \mathtt{blkdiag}\left(\mathbf{C}_{l,i}, \mathtt{diag}\{\mathbf{v}_{j,k}^H\mathbf{C}_{j,i}\mathbf{v}_{j,k}\}_{j\neq l}\right), \tag{A1}$$

where $\widetilde{\mathbf{z}}_{l,ki}$ is given by,

$$\widetilde{\mathbf{z}}_{l,ki} = \left[\widetilde{\mathbf{h}}_{l,i}^T \quad \widetilde{\gamma}_{1,k}^i \quad \dots \quad \widetilde{\gamma}_{l-1,k}^i \quad \widetilde{\gamma}_{l+1,k}^i \quad \dots \quad \widetilde{\gamma}_{L,k}^i\right]^T, \tag{A2}$$

with $\widetilde{\gamma}_{j,k}^i = \mathbf{v}_{j,k}^H\mathbf{D}_{j,k}\widetilde{\mathbf{h}}_{j,i}$ for $j \neq l$, and

$$\widetilde{\mathbf{N}}_k = \mathbb{E}\{\widetilde{\mathbf{n}}_l\widetilde{\mathbf{n}}_l^H\} = \mathtt{blkdiag}\left(\mathbf{I}_N, \mathtt{diag}\{\|\mathbf{D}_{j,k}\mathbf{v}_{j,k}\|^2\}_{j\neq l}\right). \tag{A3}$$

The denominator in (11) for MADUO can be written as

$$\widetilde{\mathbf{v}}_k^H \underbrace{\left(\sum_{i\neq k} p_i\hat{\mathbf{z}}_{l,ki}\hat{\mathbf{z}}_{l,ki}^H + \sum_{i=1}^{K} p_i\widetilde{\mathbf{Z}}_{ki} + \sigma^2\widetilde{\mathbf{N}}_k\right)}_{\widetilde{\mathbf{B}}_k} \widetilde{\mathbf{v}}_k, \tag{A4}$$

where the matrix in the parentheses can be easily shown to be equal to $\widetilde{\mathbf{B}}_k$ in (40). The numerator in (11) for MADUO can be written as $p_k|\widetilde{\mathbf{v}}_k\hat{\mathbf{z}}_{l,kk}|^2$. Hence, the SINR expression for MADUO is given by (37), and the SE expression is straightforwardly derived by adopting (10) to MADUO, thereby resulting in (36).

APPENDIX B
COMPUTATIONAL COMPLEXITY OF MADUO

ASAP $j \in \mathcal{A}_k \setminus \{l_k\}$ performs

$$\underbrace{N|\mathcal{U}_j|}_{\text{fusion of channel estimate (28)}} + \underbrace{(N^2+N)}_{\text{Hermitian product (51)}} + \underbrace{\mathcal{C}(\mathbf{v})}_{\text{local receive combiner}} \tag{B1}$$

complex multiplications. The total number of complex multiplications performed at the ASAPs is given by

$$N \sum_{j \in \mathcal{A}_k \setminus \{l_k\}} |\mathcal{U}_j| + (|\mathcal{A}_k| - 1)(N^2 + N + \mathcal{C}(\mathbf{v})) \,. \tag{B2}$$

For the calculation of $\overline{\mathbf{B}}_k$ in (48), MAP l_k requires

$$\begin{aligned} &\overbrace{\frac{N^2+N}{2}|\mathcal{U}_{l_k}|}^{\text{top-left block}} + \overbrace{N \sum_{j \in \mathcal{A}_k \setminus \{l_k\}} |\mathcal{U}_{l_k,j}|}^{\text{off-diagonal blocks}} \\ &\quad + \underbrace{\sum_{j \in \mathcal{A}_k \setminus \{l_k\}} \sum_{\substack{q \in \mathcal{A}_k \setminus \{l_k\} \\ q \geq j}} |\mathcal{U}_{j,q}|}_{\text{bottom-right block}} \end{aligned} \tag{B3}$$

complex multiplications. Finally, MAP l_k requires

$$\underbrace{(\dim(\overline{\mathbf{B}}_k))^2}_{\text{matrix-vector multiplication in (52)}} + \underbrace{\frac{(\dim(\overline{\mathbf{B}}_k))^3 - \dim(\overline{\mathbf{B}}_k)}{3}}_{\text{matrix inversion in (52)}}, \tag{B4}$$

complex multiplications for the calculation of the receive combiner in (52), where $\dim(\overline{\mathbf{B}}_k) = N + |\mathcal{A}_k| - 1$ is the order of the square matrix $\overline{\mathbf{B}}_k$. The summation of (B2)–(B4) gives the final result in (56).

REFERENCES

[1] A. Angelou, P. Behmandpoor, and M. Moonen, "Master-assisted distributed uplink operation for Cell-Free Massive MIMO networks," in *ICASSP*, 2026, accepted for publication. [Online]. Available: https://arxiv.org/abs/2601.19518

[2] D. Gesbert, M. Kountouris, R. W. Heath, C.-B. Chae, and T. Salzer, "Shifting the mimo paradigm," *IEEE signal processing magazine*, vol. 24, no. 5, pp. 36–46, 2007.

[3] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE transactions on wireless communications*, vol. 9, no. 11, pp. 3590–3600, 2010.

[4] E. Björnson, J. Hoydis, L. Sanguinetti *et al.*, "Massive mimo networks: Spectral, energy, and hardware efficiency," *Foundations and Trends® in Signal Processing*, vol. 11, no. 3-4, pp. 154–655, 2017.

[5] D. Gesbert, S. Hanly, H. Huang, S. S. Shitz, O. Simeone, and W. Yu, "Multi-cell mimo cooperative networks: A new look at interference," *IEEE journal on selected areas in communications*, vol. 28, no. 9, pp. 1380–1408, 2010.

[6] D. Wang, J. Wang, X. You, Y. Wang, M. Chen, and X. Hou, "Spectral efficiency of distributed mimo systems," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 10, pp. 2112–2127, 2013.

[7] H. Q. Ngo, G. Interdonato, E. G. Larsson, G. Caire, and J. G. Andrews, "Ultradense cell-free massive mimo for 6g: Technical overview and open questions," *Proceedings of the IEEE*, 2024.

[8] H. Q. Ngo, A. Ashikhmin, H. Yang, E. G. Larsson, and T. L. Marzetta, "Cell-free massive mimo: Uniformly great service for everyone," in *2015 IEEE 16th international workshop on signal processing advances in wireless communications (SPAWC)*. IEEE, 2015, pp. 201–205.

[9] ——, "Cell-free massive mimo versus small cells," *IEEE Transactions on Wireless Communications*, vol. 16, no. 3, pp. 1834–1850, 2017.

[10] G. Interdonato, E. Björnson, H. Quoc Ngo, P. Frenger, and E. G. Larsson, "Ubiquitous cell-free massive mimo communications," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 1, pp. 1–13, 2019.

[11] E. Björnson and L. Sanguinetti, "Making cell-free massive mimo competitive with mmse processing and centralized implementation," *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 77–90, 2019.

[12] J. Zhang, S. Chen, Y. Lin, J. Zheng, B. Ai, and L. Hanzo, "Cell-free massive mimo: A new next-generation paradigm," *IEEE access*, vol. 7, pp. 99 878–99 888, 2019.

[13] H. A. Ammar, R. Adve, S. Shahbazpanahi, G. Boudreau, and K. V. Srinivas, "User-centric cell-free massive mimo networks: A survey of opportunities, challenges and solutions," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 1, pp. 611–652, 2021.

[14] J. Zheng, J. Zhang, H. Du, D. Niyato, B. Ai, M. Debbah, and K. B. Letaief, "Mobile cell-free massive mimo: Challenges, solutions, and future directions," *IEEE Wireless Communications*, vol. 31, no. 3, pp. 140–147, 2024.

[15] E. Björnson and L. Sanguinetti, "Scalable cell-free massive mimo systems," *IEEE Transactions on Communications*, vol. 68, no. 7, pp. 4247–4261, 2020.

[16] E. Bjornson, N. Jalden, M. Bengtsson, and B. Ottersten, "Optimality properties, distributed strategies, and measurement-based evaluation of coordinated multicell ofdma transmission," *IEEE Transactions on Signal Processing*, vol. 59, no. 12, pp. 6086–6101, 2011.

[17] Ö. T. Demir, E. Björnson, L. Sanguinetti *et al.*, "Foundations of user-centric cell-free massive mimo," *Foundations and Trends® in Signal Processing*, vol. 14, no. 3-4, pp. 162–472, 2021.

[18] J. Zhang, J. Zhang, E. Björnson, and B. Ai, "Local partial zero-forcing combining for cell-free massive mimo systems," *IEEE Transactions on Communications*, vol. 69, no. 12, pp. 8459–8473, 2021.

[19] Z. Wang, J. Zhang, E. Björnson, D. Niyato, and B. Ai, "Optimal bilinear equalizer for cell-free massive mimo systems over correlated rician channels," *IEEE Transactions on Signal Processing*, 2025.

[20] J. Zheng, J. Zhang, and B. Ai, "Team-optimal mmse combining for cell-free massive mimo systems," in *ICC 2022-IEEE International Conference on Communications*. IEEE, 2022, pp. 1306–1311.

[21] G. Femenias, F. Riera-Palou, and E. Björnson, "Another twist to the scalability in cell-free massive mimo networks," *IEEE Transactions on Communications*, vol. 71, no. 11, pp. 6793–6804, 2023.

[22] I. Kanno, M. Ito, T. Ohseki, K. Yamazaki, Y. Kishi, T. Choi, W.-Y. Chen, and A. F. Molisch, "Fronthaul load-reduced scalable cell-free massive mimo by uplink hybrid signal processing," in *2022 IEEE 95th Vehicular Technology Conference:(VTC2022-Spring)*. IEEE, 2022, pp. 1–5.

[23] L. P. Schulz, C. Schappmann, and G. Bauch, "Scalable cell-free massive mimo with fully distributed large-scale fading decoding," in *2024 19th International Symposium on Wireless Communication Systems (ISWCS)*. IEEE, 2024, pp. 1–6.

[24] Z. H. Shaik, S. S. Thoota, E. Bj, E. G. Larsson *et al.*, "Over-the-air fronthaul signaling for uplink cell-free massive mimo systems," *IEEE Transactions on Wireless Communications*, 2025.

[25] I. Viering, H. Hofstetter, and W. Utschick, "Spatial long-term variations in urban, rural and indoor environments," in *Proc. 5th COST*, vol. 273, 2002, pp. 1–10.

[26] R. Van Rompaey and M. Moonen, "Gevd-based low-rank channel covariance matrix estimation and mmse channel estimation for uplink cellular massive mimo systems," in *2022 30th European Signal Processing Conference (EUSIPCO)*. IEEE, 2022, pp. 1641–1645.

[27] ——, "Distributed mmse-based uplink receive combining, downlink transmit precoding and optimal power allocation in cell-free massive mimo systems," *EURASIP Journal on Wireless Communications and Networking*, vol. 2023, no. 1, p. 48, 2023.

[28] D. Neumann, M. Joham, and W. Utschick, "Covariance matrix estimation in massive mimo," *IEEE Signal Processing Letters*, vol. 25, no. 6, pp. 863–867, 2018.

[29] K. Upadhya and S. A. Vorobyov, "Covariance matrix estimation for massive mimo," *IEEE Signal Processing Letters*, vol. 25, no. 4, pp. 546–550, 2018.

[30] E. Björnson, E. Jorswieck *et al.*, "Optimal resource allocation in coordinated multi-cell systems," *Foundations and Trends® in Communications and Information Theory*, vol. 9, no. 2–3, pp. 113–381, 2013.

[31] R. Rogalin, O. Y. Bursalioglu, H. Papadopoulos, G. Caire, A. F. Molisch, A. Michaloliakos, V. Balan, and K. Psounis, "Scalable synchronization and reciprocity calibration for distributed multiuser mimo," *IEEE transactions on wireless communications*, vol. 13, no. 4, pp. 1815–1831, 2014.

[32] Z. Wang, E. Björnson, J. Zhang, P. Zhang, V. Petrov, and B. Ai, "Statistics approximation-enabled distributed beamforming for cell-free massive mimo," *arXiv preprint arXiv:2602.03590*, 2026.