Latency Fairness for MEC-Enabled Cell-free massive MIMO

ICA- and AI-based Approaches

Nguyen, Hieu V.; Bui, Van-Phuc; Le, Mai T. P.; Nguyen-Duy-Nhat, Vien; Nguyen-Le, Hung; Tran, Nghi H.

Published in:
IEEE Communications Letters

DOI (link to publication from Publisher):
10.1109/LCOMM.2025.3581730

Publication date:
2025

Document Version
Accepted author manuscript, peer reviewed version

Latency Fairness for MEC-Enabled Cell-free massive MIMO: ICA- and AI-based Approaches

Hieu V. Nguyen, Van-Phuc Bui, Mai T. P. Le, Vien Nguyen-Duy-Nhat, Hung Nguyen-Le, and Nghi H. Tran

***Abstract*—This paper investigates the latency minimization at the network edge in mobile edge computing (MEC)-enabled Cell-Free massive MIMO systems. We introduce a new edge computing model that integrates both task offloading and local execution. To minimize overall system latency while considering power allocation constraints, we formulate an optimization problem aimed at reducing maximum computing time. This mixed-integer non-convex problem is then reformulated into a more tractable form, which is solved using an iterative convex approximation method to achieve locally-optimal solutions. Additionally, we propose a convolutional neural network-based algorithm as an alternative solution to further improve system efficiency. Numerical results are provided to validate the theoretical framework and demonstrate the effectiveness of the proposed approaches in accelerating the data processing in MEC-enabled cell-free networks.**

***Index Terms*—Cell-free, CNN, ICA, latency, mobile edge computing, resource allocation.**

I. Introduction

THE upcoming generation of mobile communication networks, commonly referred to as sixth generation (6G) or NextG, is specifically designed to accommodate diverse service types with stringent requirements for throughput, reliability, and latency [1]. To address the challenges associated with computation-intensive and latency-sensitive applications, mobile edge computing (MEC) has emerged as a highly promising solution [2]. MEC systems enable users to offload resource-demanding tasks to nearby high-performance servers, significantly reducing latency compared to traditional cloud-based architectures [3].

Beyond conventional cellular massive multiple-input multiple-output (mMIMO), cell-free (CF) mMIMO has emerged as a key enabler for NextG networks due to its potential for substantial improvements in spectral and energy efficiency [4], [5]. In the context of MEC-enabled systems, CF-mMIMO offers enhanced transmission rates and scalability for large-scale user offloading. Despite its advantages, research on MEC-CF systems remains limited [6]–[9]. One of the earliest studies, by Mukherjee and Lee [6], analyzed success probabilities-defined as the successful completion of either communication or computation-using stochastic geometry and queueing theory. More recently, works such as [7], [8] have addressed resource optimization in MEC-CF networks, focusing on minimizing latency and energy consumption through joint task offloading and computational allocation. The study in [9] further examines heuristic NOMA pairing for MEC CF-mMIMO in a single-carrier setting.

Most prior studies have focused on optimizing data transmission under latency assumptions of tens of milliseconds, which are unrealistic for physical-layer implementations. In practice, a channel coherence block spans only half of a one-millisecond sub-frame, resulting in highly dynamic channel conditions. Consequently, latency must be managed within this coherence interval to ensure both spectral efficiency and reliable data rates. To address this challenge, this work investigates the latency optimization problem in a multi-carrier MEC-enabled CF massive MIMO network, with a particular emphasis on fairness under dense-user and fast-fading conditions. The main contributions of this paper are as follows:

- **Novel Framework:** We propose a new mathematical framework that jointly integrates mobile edge computing (MEC) and resource sharing within a CF-mMIMO architecture, with a core focus on latency fairness among users. The problem is formulated as a min–max latency optimization aiming to minimize the maximum computing latency across all users within the coherence time. This ensures equitable service levels, where no user experiences significantly higher latency than others. The formulation captures practical physical-layer constraints such as power allocation, dynamic offloading ratios, and resource limits. It results in a hard-solving mixed-integer non-convex (MINC) optimization problem class that reflects real-time edge processing requirements.
- **Efficient Solution Methodology:** We reformulate the problem into a tractable form and solve it using an iterative convex approximation (ICA) algorithm to obtain locally optimal solutions under realistic constraints. To further enhance scalability and inference speed, we design a convolutional neural network (CNN)-based supervised learning model that approximates the ICA solution. This AI-based model enables fast adaptation to varying channel states and supports generalization across different network configurations, facilitating practical deployment in large-scale MEC-enabled wireless systems.

II. System Model

In this work, we investigate a MEC-CF network with a high density of users. The MEC-CF system operates in a time-division duplexing (TDD) mode and is equipped with a set $\mathcal{M}$ of M access points (APs), where the AP set is denoted as $\mathcal{M} = \{1, 2, \ldots, M\}$. These APs simultaneously serve a set $\mathcal{K}$

This work was funded in part by the Postdoctoral Scholarship Programme of Vingroup Innovation Foundation (VINIF), code VINIF.2024.STS.05, in part by the Funds for Science and Technology Development of the University of Danang under Project B2021-DN02-06. *(Corresponding author: Hieu V. Nguyen.)*

Hieu V. Nguyen, Mai T. P. Le and Vien Nguyen-Duy-Nhat are with Faculty of Electronics and Telecommunication Engineering, University of Science and Technology, The University of Danang, Da Nang, Vietnam. ({nvhieu,lpmai,ndnvien}@dut.udn.vn)

Van-Phuc Bui is with Department of Electronic Systems, Aalborg University, Denmark. (vpb@es.aau.dk)

Hung Nguyen-Le is with University of Technology and Education, The University of Danang, Da Nang, Vietnam. (nlhung@ute.udn.vn)

Nghi H. Tran is with Department of Electrical and Computer Engineering, College of Engineering, University of Akron, OH, USA. (nghi.tran@uakron.edu)

of K edge user equipments (EUEs), where $\mathcal{K} = \{1, 2, \ldots, K\}$. The total number of APs' antennas is $L = \sum_{m\in\mathcal{M}} L_m$, where L_m is the number of antennas at AP m, while each EUE has a single-antenna. All APs are connected to the CPU, through perfect wide bandwidth backhaul links, which has a sufficiently large capacities to serve all EUEs.

A. Data Transmission in MEC-CF Networks

At the CPU, the offloading data sent by an EUE is decoded and processed by aggregating the received signals from all APs through the uplink (UL) data broadcast. The data transmission between the set of APs and EUEs occurs via N subcarriers within a given resource block. To accelerate the computational capacity of the EUEs, the MEC server equipped at the CPU fetches the offloading data from the EUEs.

In this UL data transmission setting, without loss of generality (w.l.o.g), the channel response from EUE k, $k \in \mathcal{K}$, to AP m through subcarrier n, where $n \in \mathcal{N} = \{1, 2, \ldots, N\}$, is denoted by $\mathbf{h}_{m,k,n} \in \mathbb{C}^{L_m\times 1}$. These channels are characterized by both small-scale and large-scale fading, such that $\mathbf{h}_{m,n,k} = \beta_{m,k}\bar{\mathbf{h}}_{m,n,k}$. Here, the large-scale fading $\beta_{m,k}$ specifies the path loss and shadowing, while the small-scale fading described by a vector $\bar{\mathbf{h}}_{m,n,k}$ of which each entry is independently drawn from a circularly symmetric complex Gaussian distributionn $\mathcal{CN}(0, 1)$.

Consider the data transmission from K EUEs to the AP $m,\ m \in \mathcal{M}$, for a given sub-carrier $n,\ n \in \mathcal{N}$. First, the transmitted signal $\mathbf{x}_n \in \mathbb{C}^{K\times 1}$ of K EUEs on sub-carrier n can be expressed as a Hadamard product $\circ$, i.e.,

$$\mathbf{x}_n = \boldsymbol{\alpha}_n \circ (\mathtt{diag}(\mathbf{p}_n)\tilde{\mathbf{x}}_n), \tag{1}$$

where $\mathbf{p}_n \triangleq \left[\sqrt{p_{n,1}}\ \sqrt{p_{n,2}}\ \cdots \sqrt{p_{n,K}}\right]$ is the power allocation vector, with $p_{n,k}$ being the power coefficient for EUE k on sub-carrier n. The vector $\tilde{\mathbf{x}}_n \in \mathbb{C}^{K\times 1}$ is the concatenation of transmitted symbols from K EUEs, of which the k-th entry $\tilde{x}_{n,k}$ represents the message sent from EUE k on sub-carrier n, with $\mathbb{E}\left[|\tilde{x}_{n,k}|^2\right] = 1$. For resource allocation, $\boldsymbol{\alpha}_n \triangleq [\alpha_{n,1}\ \cdots\ \alpha_{n,K}]$ is newly introduced as a binary variable vector, where $\alpha_{n,k} = 1$ if EUE k is allocated to sub-carrier n, and $\alpha_{n,k} = 0$ if otherwise.

Next, the received signal at AP m, denoted as $\mathbf{r}_{m,n} \in \mathbb{C}^{L_m\times 1}$, can be represented as $\mathbf{r}_{m,n} = \mathbf{H}_{m,n}\mathbf{x}_n + \mathbf{z}_{m,n}$, where $\mathbf{H}_{m,n} \triangleq [\mathbf{h}_{m,n,1}\ \mathbf{h}_{m,n,2}\ \cdots \mathbf{h}_{m,n,K}] \in \mathbb{C}^{L_m\times K}$ is the channel matrix between the EUEs and AP m on the sub-carrier n. The term $\mathbf{z}_{m,n} \in \mathbb{C}^{M\times 1}$ represents the additive white Gaussian noise (AWGN) at AP m for sub-carrier n, where each element of $\mathbf{z}_{m,n}$ follows $\mathcal{CN}(0, \sigma_m^2)$.

We suppose that a linear receiver (e.g., MRC or matched filter) is locally adopted at the APs, the post-processed messages from K UEs, denoted by $\mathbf{y}_n \in \mathbb{C}^{K\times 1}$, are thus expressed as

$$\mathbf{y}_n = \sum\nolimits_{m=1}^{M} \mathbf{A}_{m,n}^H \mathbf{r}_{m,n} \tag{2}$$

where $\mathbf{A}_{m,n} \in \mathbb{C}^{L_m\times K}$ is the linear receiver at AP m on the sub-carrier n, in which the column k is normalized as $\sum_{m=1}^{M} \|\mathbf{a}_{m,n,k}^H \mathbf{a}_{m,n,k}\|^2 = 1$. From (1) and (2), one achieves the corresponding signal-to-interference-plus-noise (SINR) for decoding the message:

$$\gamma_{n,k}(\boldsymbol{\alpha}_n, \mathbf{p}_n) = \frac{\alpha_{n,k} p_{n,k} \bar{P}_k |\sum_{m=1}^{M} \mathbf{a}_{m,n,k}^H \mathbf{h}_{m,n,k}|^2}{\phi_{n,k}(\boldsymbol{\alpha}_n, \mathbf{p}_n)}, \tag{3}$$

where the interference-plus-noise is determined by

$$\phi_{n,k}(\boldsymbol{\alpha}_n, \mathbf{p}_n) \triangleq \sum\nolimits_{k'=1,k'\neq k}^{K} \alpha_{n,k'} p_{n,k'} \tilde{P}_{n,k'} + \bar{\sigma}^2,$$

with $\tilde{P}_{n,k'} \triangleq \bar{P}_k |\sum_{m=1}^{M} \mathbf{a}_{m,n,k}^H \mathbf{h}_{m,n,k'}|^2,\ k \neq k'$. The normalized transmit power and noise power at AP m is represented as $\bar{P}_k \triangleq \frac{P_k^{\max}}{\min_{m\in\mathcal{M}}\{\sigma_m^2\}}$ and $\bar{\sigma}^2 \triangleq \frac{\sum_{m=1}^{M}\sigma_m^2}{\min_{m\in\mathcal{M}}\{\sigma_m^2\}}$, where $P_k^{\max}$ is the maximum transmit power of EUE k. Let $\mathbf{p} \triangleq [\mathbf{p}_1^T \cdots \mathbf{p}_N^T]^T$ and $\boldsymbol{\alpha} \triangleq [\boldsymbol{\alpha}_1^T \cdots \boldsymbol{\alpha}_N^T]^T$, the achievable data rate for the offloading process of EUE k can be formulated as

$$R_k(\boldsymbol{\alpha}, \mathbf{p}) = \sum\nolimits_{n=1}^{N} B \log_2\left(1 + \gamma_{n,k}(\boldsymbol{\alpha}_n, \mathbf{p}_n)\right), \text{ (bits/s)} \tag{4}$$

with B the system bandwidth.

B. Edge Computing Model

To accelerate computing capability, we consider both local computing and task offloading. Let Z_k and C_k denote the size (in bits) of the offloading data requested by EUE k and the number of CPU cycles required to execute one bit of that task, respectively. Additionally, $f^{\mathtt{SER}}$ and f_k represent the computational frequencies at the MEC server and EUE k, respectively. We define a new variable $\boldsymbol{\rho} \triangleq [\rho_k]_{k\in\mathcal{K}}$, where $0 \leq \rho_k \leq 1$), which represents the offloading ratio for the task of EUE k.

The data-processing request for EUE k involves a trade-off between the amount of data processed locally and the portion offloaded to the MEC server. The latencies at the edge and MEC server[1], denoted as $\tau_k^{\mathtt{loc}}$ and $\tau_k^{\mathtt{off}}$, are respectively given by

$$\tau_k^{\mathtt{loc}}(\boldsymbol{\rho}) = \frac{(1-\rho_k) Z_k C_k}{f_k}, \tag{5}$$

$$\tau_k^{\mathtt{off}}(\boldsymbol{\alpha}, \mathbf{p}, \boldsymbol{\rho}) = \frac{\rho_k Z_k}{R_k(\boldsymbol{\alpha}, \mathbf{p})} + \frac{\rho_k Z_k C_k}{f^{\mathtt{SER}}}. \tag{6}$$

The overall latency of EUE k is therefore defined as

$$\tau_k^{\mathtt{MEC}} = \max\left\{\tau_k^{\mathtt{loc}}(\boldsymbol{\rho}), \tau_k^{\mathtt{off}}(\boldsymbol{\alpha}, \mathbf{p}, \boldsymbol{\rho})\right\},\ \forall k \in \mathcal{K}. \tag{7}$$

C. Problem Formulation

Once the computational latency within each coherent processing block is reduced, the user experience in real-time applications is significantly improved. This work aims at enhancing the experience for a certain application through making the latency fairness among the EUEs. Therefore, a min-max optimization problem in reducing the execution time for all users can be formulated as

$$\begin{aligned}
\underset{\boldsymbol{\alpha},\mathbf{p},\boldsymbol{\rho}}{\mathtt{minimize}}\quad & \max_{k\in\mathcal{K}}\{\tau_k^{\mathtt{MEC}}\}, && \text{(8a)}\\
\mathtt{s.t.}\quad & \alpha_{n,k} \in \{0,1\},\ n \in \mathcal{N},\ k \in \mathcal{K}, && \text{(8b)}\\
& p_{n,k} \geq 0,\ n \in \mathcal{N},\ k \in \mathcal{K}, && \text{(8c)}\\
& \sum\nolimits_{n\in\mathcal{N}} p_{n,k} \leq 1,\ k \in \mathcal{K}, && \text{(8d)}\\
& 0 \leq \rho_k \leq 1,\ k \in \mathcal{K}. && \text{(8e)}
\end{aligned}$$

Constraints (8b)-(8d) refers to resource and power allocation, respectively. It should be emphasized that the power coefficient $p_{n,k}$ is normalized as the ratio of transmit power used by the EUE k over a predetermined maximum transmit power.

[1]We assume that the other delay (e.g., for queuing process) is a constant parameter, and thus the latency at MEC server depends on the transmission delay and processing time.

Constraint (8e) restricts the normalized amount of offloading data requested by EUE k. Hence, $\rho_k \in [0,1]$ in which $\rho_k = 0$ and $\rho_k = 1$ specify the fully-local and fully-offloading computing, respectively. Although the feasible set of problem (8) is quasi-convex, it is extremely hard to be solved due to the non-convex objective function and mixed-integer linear constraints.

III. Proposed Solution based on ICA Approach

Given the non-convex and mixed-integer nature of problem (8), standard methods like Lagrangian relaxation or the Alternating Direction Method of Multipliers (ADMM) often struggle with feasibility and convergence under tight variable coupling. To address this, we adopt the ICA method for its simplicity, scalability, and ability to efficiently yield locally optimal solutions—especially useful in latency-constrained MEC systems requiring large-scale AI training data. To solve the original problem, we propose a two-step approach for efficiently finding the locally-optimal solution as below.

A. Reduction of Binary Variables

The optimization problem in (8) belongs to the category of min-max programming, where the objective function takes the form of a supremum involving non-convex functions. Addressing this challenge involves transforming problem (8) into an equivalent formulation, which can be expressed as follows

$$\begin{aligned} \underset{\boldsymbol{\alpha},\mathbf{p},\boldsymbol{\rho},t}{\texttt{minimize}} \quad & t, && \text{(9a)}\\ \texttt{s.t.} \quad & (8\text{b})-(8\text{e}), && \text{(9b)}\\ & \tau_k^{\texttt{loc}} \le t,\ k\in\mathcal{K}, && \text{(9c)}\\ & \tau_k^{\texttt{off}}(\boldsymbol{\alpha},\mathbf{p},\boldsymbol{\rho}) \le t,\ k\in\mathcal{K}. && \text{(9d)} \end{aligned}$$

Before solving problem (9), we consider the following theorem.

Theorem 1: Let the feasible region of problem (9) be $\mathcal{F} \triangleq \{(\boldsymbol{\alpha},\mathbf{p},\boldsymbol{\rho},t)|(9\text{b})-(9\text{d})\}$. The optimal solution of problem (9), denoted as $\mathbf{s}^* \triangleq (\boldsymbol{\alpha}^*,\mathbf{p}^*,\boldsymbol{\rho}^*,t^*)$, can be found by solving a relaxation problem:

$$\begin{aligned} \underset{\boldsymbol{\alpha},\mathbf{p},\boldsymbol{\rho},t}{\texttt{minimize}} \quad & t, && \text{(10a)}\\ \texttt{s.t.} \quad & (8\text{c})-(8\text{e}),(9\text{c}),(9\text{d}), && \text{(10b)}\\ & \alpha_{n,k}\in[0,1],\ n\in\mathcal{N},\ k\in\mathcal{K}, && \text{(10c)} \end{aligned}$$

where the feasible set $\bar{\mathcal{F}} \triangleq \{(\boldsymbol{\alpha},\mathbf{p},\boldsymbol{\rho},t)|(10\text{b})-(10\text{c})\}$. Then, the optimal solution of (8) can be found in a tighter set

$$\mathbf{s}^* \in \hat{\mathcal{F}} \triangleq \left\{(\boldsymbol{\alpha},\mathbf{p},\boldsymbol{\rho},t)\in\bar{\mathcal{F}}|\boldsymbol{\alpha}=\mathbf{1}\right\}. \quad \text{(11)}$$

Proof: Please see Appendix. □

By utilizing Theorem 1, we can find the optimal solution for problem (9) through 2 steps:

- Assigning $\boldsymbol{\alpha}=\mathbf{1}$, the partially optimal solution $\hat{\mathbf{s}}^*$ can be calculated using the following problem:
$$\hat{\mathbf{s}}^* = \underset{\hat{\mathbf{s}}\in\hat{\mathcal{F}}}{\text{argmin}} \quad t, \quad \text{(12)}$$
where the feasible set $\hat{\mathcal{F}} \triangleq \{(\mathbf{p},\boldsymbol{\rho},t)\in(\mathcal{C})\}$, with the set of constraints $(\mathcal{C})$ is described as
$$(\mathcal{C}) \Leftrightarrow \begin{cases}(8\text{c})-(8\text{e}),(9\text{c}), & \text{(13a)}\\ \tau_k^{\texttt{off}}(\mathbf{1},\mathbf{p},\boldsymbol{\rho}) \le t,\ k\in\mathcal{K}, & \text{(13b)}\end{cases}$$
- $\boldsymbol{\alpha}$ is then recovered using the optimal solution of $\mathbf{p}$:
$$\alpha_{n,k} = \left[\frac{p_{n,k}-\varepsilon}{|p_{n,k}-\varepsilon|}\right]^+, \quad \text{(14)}$$
where ε is very small positive number.

It can be seen that constraints in (13a) are linear, but (13b) is still non-convex. Therefore, the following part focuses on addressing the non-convexity of (13b).

B. Addressing Non-convexity of Constraint (13b)

Before addressing constraint (13b), we introduce the two functions and their first-order approximations applied to solve problem (12). With given a convex quadratic-over-linear function $f(x,y) = \frac{x^2}{y},\ x,y>0$, and a concave product function $g(x,y) = \sqrt{xy},\ x,y>0$, the lower bound of $f(x,y)$ and upper bound of $g(x,y)$ around an arbitrary point $(x^{(i)},y^{(i)})$ can be respectively derived as

$$f(x,y) \ge \frac{2x^{(i)}}{y^{(i)}}x - \frac{(x^{(i)})^2}{(y^{(i)})^2}y := f^{(i)}(x,y), \quad \text{(15)}$$

$$g(x,y) \le \frac{\sqrt{y^{(i)}}}{2\sqrt{x^{(i)}}}x + \frac{\sqrt{x^{(i)}}}{2\sqrt{y^{(i)}}}y := g^{(i)}(x,y). \quad \text{(16)}$$

To convexify problem (12), constraint (13b) is first transformed into the equivalent equations as follows.

$$(13\text{b}) \Leftrightarrow \begin{cases}\sum_{n=1}^{N} B\log_2(1+\lambda_{k,n}) \ge \frac{1}{\mu_k}, & \text{(17a)}\\ \gamma_{k,n}(\mathbf{1},\mathbf{p}) \ge \lambda_{k,n}, & \text{(17b)}\\ Z_k\rho_k\mu_k + \rho_k\frac{Z_kC_k}{f^{\texttt{SER}}} \le t. & \text{(17c)}\end{cases}$$

It can be seen that (17a) is convex, given by the fact that $\log_2(1+\lambda_{k,n})$ is concave and $\frac{1}{\mu_k}$ is convex. In what follows, we need to convexify (17b) and (17c).

Convexifying (17b) *and* (17c)*:* We first introduce a new variable $\boldsymbol{\varphi} \triangleq \{\varphi_{n,k}\}_{n\in\mathcal{N},k\in\mathcal{K}}$. By applying (15), constraint (17b) is then approximated to the following convex constraints:

$$(17\text{b}) \Leftrightarrow \begin{cases} f^{(i)}(\sqrt{p_{n,k}},\varphi_{n,k}) \ge \lambda_{k,n}/\tilde{P}_{n,k},\ \forall n,k, & \text{(18a)}\\ \phi_{n,k}(\mathbf{1},\mathbf{p}_n) \le \varphi_{n,k},\ \forall n,k. & \text{(18b)}\end{cases}$$

By letting a new variable be $\boldsymbol{\nu} \triangleq \{n_k\}_{k\in\mathcal{K}}$ and applying (16), constraint (17c) is convexified as

$$(17\text{c}) \Leftrightarrow \begin{cases} Z_k\mu_k + \frac{Z_kC_k}{f^{\texttt{SER}}} \le \nu_k,\ \forall k, & \text{(19a)}\\ g^{(i)}(\rho_k^2,\nu_k^2) \le t,\ \forall k. & \text{(19b)}\end{cases}$$

Finally, the successive convex programming, in which each sub-problem provides a minimum majorant at iteration i, is formulated as

$$\underset{\substack{\mathbf{p},\boldsymbol{\rho},t,\\ \boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu},\boldsymbol{\varphi}}}{\texttt{minimize}} \quad t, \quad \texttt{s.t.}\ (8\text{c})-(8\text{e}),(9\text{c}),(17\text{a}),(18),(19),$$

where $\boldsymbol{\lambda} \triangleq \{\lambda_{n,k}\}_{n\in\mathcal{N},k\in\mathcal{K}}$ and $\boldsymbol{\mu} \triangleq \{\mu_k\}_{k\in\mathcal{K}}$.

Remark 1: The complexity of ICA-based algorithm is determined using the number of variables and constraints. Particularly, it takes $\mathcal{O}(\sqrt{(NK+2K)^5}(NK+K)^2\log(1/\epsilon))$ where the term of $\log(1/\epsilon)$ intuitively approximates the linear convergence rate for the outer loop.

IV. Proposed Solution Based on Data-Driven Power Allocation

Although ICA-based method provides a locally optimal solution with low-complexity execution, the new network architectures with dense-user communication require the faster

This article has been accepted for publication in IEEE Communications Letters. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/LCOMM.2025.3581730

processing. Therefore, we now explore a novel AI-driven approach that leverages the ICA solution during training. In what follows, we introduces a data-driven approach to optimize power allocation (p_k) in MEC-enabled CF-mMIMO networks. The objective is to minimize the maximum computational latency across all users ($k \in \mathcal{K}$) subject to channel conditions and power constraints as

$$\min_{\boldsymbol{\alpha},\mathbf{p},\boldsymbol{\rho}} \max_{k\in\mathcal{K}} \{\tau_k^{\text{MEC}}(\boldsymbol{\alpha},\mathbf{p},\boldsymbol{\rho})\}. \tag{20}$$

We propose a data-driven method that leverages the available channel state information (CSI), assumed to be perfectly known, as the input to our learning model. Specifically, the magnitude of the complex CSI is used as the primary feature, defined as $\mathbf{X}_i = |\mathbf{H}_i|$, where $\mathbf{H}_i \triangleq \{\mathbf{H}_n\}_{n\in\mathcal{N}}$ is the complex channel matrix for the i-th sample, and $|\cdot|$ represents the element-wise magnitude operation. To ensure numerical stability and facilitate efficient training, we apply min-max normalization to each element of $\mathbf{p}$ over the range of minimum and maximum values of $\mathbf{p}$ across all training samples. The model is trained using supervised learning, where the goal is to minimize the error between the predicted and ground-truth (optimal) solutions. These labels are obtained from the ICA-based algorithm, which guarantees locally optimal solutions for the original optimization problem. The training loss is the mean squared error (MSE), defined as

$$\mathcal{L}(\theta) = \frac{1}{S}\sum_{i=1}^{S}\left(\mathbf{s}_i^{\text{pred}} - \mathbf{s}_i^{\text{true}}\right)^2, \tag{21}$$

where vector $\mathbf{s}_i^{\text{true}} \triangleq [\texttt{vec}(\boldsymbol{\alpha}_i)^T \texttt{vec}(\mathbf{p}_i)^T \texttt{vec}(\boldsymbol{\rho}_i)^T]^T$ is resulted from the vectorization of ground-truth (optimal) solution for the sample i, $\mathbf{s}_i^{\text{pred}}$ is the corresponding predictions. To capture both spatial and temporal features of the wireless environment, we adopt a hybrid CNN-Long Short-Term Memory (LSTM) architecture. The CNN extracts spatial patterns across users and subcarriers, while the LSTM models temporal and structural variations. This design also enables transfer learning, allowing models trained on CF-mMIMO to be efficiently adapted to other architectures like conventional mMIMO with minimal retraining.

The significant advantage offered by AI-based method is the ability of extremely fast computation. We let k_s, B_s and F be the kernal size, mini-batch size, and the number of filters in the convolutional layer, respectively. $H_s \times W_s$ is the size of the activation map, while H_{LSTM} is the number of LSTM hidden units. With given N_s training samples, the computational complexity per training epoch of the proposed CNN combined with LSTM solution is $\mathcal{O}\left(\frac{N_s}{B_s}(Fk_s^2H_sW_s + H_{\text{LSTM}}^2)\right)$, specifically $\mathcal{O}(1.518 \times 10^3 N_s)$ for the given parameters. During inference, the complexity reduces significantly to $\mathcal{O}(Fk_s^2H_sW_s + H_{\text{LSTM}}^2)$, approximately $\mathcal{O}(4.86 \times 10^4)$, enabling sub-millisecond latency suitable for real-time MEC scenarios.

V. Numerical Results and Discussions

A. Simulation Settings

To evaluate the proposed computational model and algorithm, we conduct MATLAB simulations with 200 network topologies and 1,000 channel realizations per topology. Each topology comprises M single-antenna APs and K single-antenna EUEs, randomly distributed within a 1-kilometer radius. The system operates with a bandwidth of 20 MHz, noise power spectral density of -174 dBm/Hz, and a maximum EUE power budget of 10 dBm. The MEC server and EUEs have computing frequencies of $f_{\text{SER}} = 5$ GHz and $f_k = 1$ GHz, respectively, with $C_k = 10^3$ cycles/bit, and $Z_k = 10^3$ bits per task.

In the data-driven approach, a 2D convolutional layer (32 filters, kernel size 3) extracts features, followed by max-pooling, flattening, and an LSTM layer (50 units) to model temporal dependencies. The fully connected output layer predicts power allocation for all users and antennas. The model, optimized with the Adam algorithm and batch normalization, uses a dataset of 200,000 samples split into 80% training and 20% testing. We should emphasize that the dataset is independently collected for the below "BS-centered mMIMO" and "CF-mMIMO w/ local comp." methods.

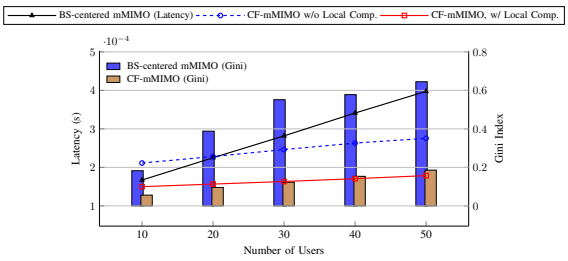

Fig. 1. Latency (left y-axis) and Gini fairness index (right y-axis) versus the number of EUEs.

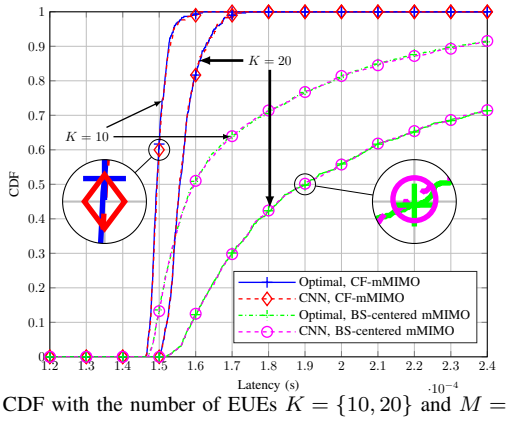

Fig. 2. CDF with the number of EUEs $K = \{10, 20\}$ and $M = 64$.

B. Performance Evaluation

The critical advantage of MEC-enabled CF-mMIMO is its ability to serve a large number of EUEs across a wide area. To evaluate its performance, we analyze the latency as the number of EUEs varies ($K = \{10, 20, \cdots, 50\}$) and compare the proposed approach "CF-mMIMO with local computing" (referred to as "CF-mMIMO w/ local comp.") with two alternative schemes:

- Traditional BS-centered mMIMO ("BS-centered mMIMO"): An MEC server at the macrocell center serves K EUEs, considering the offloading ratio.
- CF-mMIMO without local computing ("CF-mMIMO w/o local comp."): Inspired from the architecture in [10], this scheme employs CF-mMIMO for MEC but offloads all EUE data without local computing.

The proposed ICA-based algorithm is consistently developed across all scenarios, regardless of network architecture or computational technique. For comparison purpose, it is employed

in all three schemes. As shown in Fig. 1, system performance improves significantly with the CF-mMIMO network architecture. While the "BS-centered mMIMO" strategy outperforms "CF-mMIMO w/o local comp." for a small number of EUEs ($K = \{10, 20\}$), the performance advantage of CF-mMIMO becomes more pronounced in dense-user networks ($K = \{30, 40, 50\}$). Across all user densities, the proposed scheme and algorithm ("CF-mMIMO w/ local comp.") consistently provide a substantial performance gain over the other approaches. Remarkably, the fairness of latencies is evaluated using Gini fairness index. Because fairness is sensitive to network topology, we examine both conventional mMIMO and CF-mMIMO with local computing. Our results show that CF-mMIMO maintains high fairness even as in highly dense-user network, i.e., the Gini index remains below 20% for 50 users.

Next, we assess the robustness of the proposed CNN in enhancing computing capabilities. Fig. 2 presents the Cumulative Distribution Function (CDF), illustrating the effectiveness of the AI-based method by evaluating the gap between the CNN output and the optimal solutions. As expected, the proposed CNN achieves results within a very small margin of the ground truth, with a performance gap of less than 1%. More interesting, we also perform cross-testing, feeding the input data from "BS-centered mMIMO"into the CNN trained for "CF-mMIMO w/ local computing". The results confirm the adaptability of the proposed CNN, demonstrating its ability to generalize across different network topologies and outperform "BS-centered mMIMO".

VI. Conclusion

In this study, we present a novel edge computing model based on a MEC-enabled CF-mMIMO network, integrating both offloading and local execution time. The primary goal is to improve system performance by minimizing data processing latency for all EUEs. To address this optimization challenge, we developed an ICA-based algorithm that guarantees at least a locally optimal solution with low complexity. Additionally, to accelerate the optimization process, we implemented a hybrid learning model combining CNN and LSTM, using the ICA-based method's optimal solution as ground truth. The numerical results demonstrate that both approaches deliver substantial performance gains and maintain reliability, even as the number of EUEs increases.

Appendix
Proof of Theorem 1

First, we need to break the mixed-integer property of problem (9) while remaining the optimality. To do this, the feasible region of $\boldsymbol{\alpha}$ is extended to the close interval $[0, 1]$ from binary set $\{0, 1\}$. Thus, the relaxation problem can be rewritten as

$$\underset{\boldsymbol{\alpha},\mathbf{p},\boldsymbol{\rho},t}{\texttt{minimize}} \quad t, \tag{22a}$$

$$\texttt{s.t.} \quad \text{(8c)} - \text{(8e)}, \text{(9c)}, \text{(9d)}, \tag{22b}$$

$$\alpha_{n,k} \in [0, 1], \ n \in \mathcal{N}, \ k \in \mathcal{K}, \tag{22c}$$

with the feasible set being $\bar{\mathcal{F}} \triangleq \{(\boldsymbol{\alpha}, \mathbf{p}, \boldsymbol{\rho}, t) | (22\text{b}) - (22\text{c})\}$. It is true that any optima $\mathbf{s}^* \triangleq (\boldsymbol{\alpha}^*, \mathbf{p}^*, \boldsymbol{\rho}^*, t^*)$ of problem (9), for $\mathbf{s}^* \in \mathcal{F} \triangleq \{(\boldsymbol{\alpha}, \mathbf{p}, \boldsymbol{\rho}, t) | (9\text{b}) - (9\text{d})\}$, belongs to the feasible set $\bar{\mathcal{F}}$, owing to $\mathcal{F} \subset \bar{\mathcal{F}}$. Consequently, the optimality of (9) as well as that of the original problem (8) are hold. For transformation and proof techniques, the feasible set should be a connected set, and thus, a relaxation step is necessary to tackle the feasible discrete set of integer variables.

Next, we consider the relationship between two variables for resource allocation, i.e., $\boldsymbol{\alpha}$ and $\mathbf{p}$ for channel and power allocation, respectively. For simplicity, we let $\bar{\boldsymbol{\alpha}} \triangleq \boldsymbol{\alpha} \circ \mathbf{p}$ (as $\bar{\boldsymbol{\alpha}} \equiv (\boldsymbol{\alpha}, \mathbf{p})$), with $\bar{\alpha}_{n,k}$ being the entry at row n and column k of $\bar{\boldsymbol{\alpha}}$. Then, the objective value of problem (22) can be formulated as

$$P^* = \inf_{\mathbf{s} \in \bar{\mathcal{F}}} \quad t = \inf_{\mathbf{s} \in \bar{\mathcal{F}}} \quad \sup_{(\bar{\boldsymbol{\alpha}}, \boldsymbol{\rho})} \Big\{ \max_{k \in \mathcal{K}} \big\{ \tau_k^{\texttt{loc}}, \tau_k^{\texttt{off}}(\bar{\boldsymbol{\alpha}}, \boldsymbol{\rho}) \big\} \Big\}. \tag{23}$$

From (6), $\tau_k^{\texttt{off}}(\bar{\boldsymbol{\alpha}}, \boldsymbol{\rho})$ is inversely proportional to $R_k(\bar{\boldsymbol{\alpha}})$, while (23) indicates that P^* is obtained through $\sup_{(\bar{\boldsymbol{\alpha}}, \boldsymbol{\rho})} \tau_k^{\texttt{off}}(\bar{\boldsymbol{\alpha}}, \boldsymbol{\rho}), \ \forall k \in \mathcal{K}$. Recall (4), the primal value is associated with $\inf_{\bar{\boldsymbol{\alpha}}} R_k(\bar{\boldsymbol{\alpha}}), \ \forall k \in \mathcal{K}$, where $\bar{\boldsymbol{\alpha}}_n$ is the row n of $\bar{\boldsymbol{\alpha}}$. Connecting to the feasible set $\mathcal{F}$ of problem (9), when $\alpha_{n,k} \to 1$, $\gamma_{n,k}(\bar{\boldsymbol{\alpha}}_n)$ as well as P^* totally depend on $p_{n,k}$. To derive (11), we need to prove that the optimality still holds even when $\boldsymbol{\alpha} = \mathbf{1}$. We assume, w.l.o.g, that the optima in problem (22) is provided by

$$R_k^*(\bar{\boldsymbol{\alpha}}) \in \Big\{ R_k(\bar{\boldsymbol{\alpha}}) \mid \exists \bar{\alpha}_{n,k} = 0 \Big\}. \tag{24}$$

It is clear that $\bar{\alpha}_{n,k} = 0$ iff $\alpha_{n,k} = 0$ or $p_{n,k} = 0$. Therefore, if an optima contains $\bar{\alpha}_{n,k}^* = 0$, the optimality is totally guaranteed with $\alpha_{n,k} > 0$ and $p_{n,k} = 0$. This leads to the fact that there exists an optimal solution with $\alpha_{n,k} = 1, \ \forall n \in \mathcal{N}, \ k \in \mathcal{K}$, which completes the proof.

References

[1] C.-X. Wang, X. You, X. Gao, X. Zhu, Z. Li, C. Zhang, H. Wang, Y. Huang, Y. Chen, H. Haas, J. S. Thompson, E. G. Larsson, M. D. Renzo, W. Tong, P. Zhu, X. Shen, H. V. Poor, and L. Hanzo, "On the road to 6g: Visions, requirements, key technologies, and testbeds," *IEEE Commun. Surv. & Tutor.*, vol. 25, no. 2, pp. 905–974, 2023.

[2] L. Ma, N. Cheng, C. Zhou, X. Wang, N. Lu, N. Zhang, K. Aldubaikhy, and A. Alqasir, "Dynamic neural network-based resource management for mobile edge computing in 6g networks," *IEEE Trans. Cognitive Commun. and Netw.*, vol. 10, no. 3, pp. 953–967, 2024.

[3] S. S. Yilmaz, B. Özbek, and R. Mumtaz, "Delay Minimization for Massive MIMO Based Cooperative Mobile Edge Computing System With Secure Offloading," *IEEE Open J. Veh. Technol.*, vol. 4, pp. 149–161, 2023.

[4] 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 Commun. Surv. & Tutor.*, vol. 24, no. 1, pp. 611–652, 2022.

[5] C. Wei, K. Xu, X. Xia, Q. Su, M. Shen, W. Xie, and C. Li, "User-centric access point selection in cell-free massive MIMO systems: A game-theoretic approach," *IEEE Commun. Lett.*, vol. 26, no. 9, pp. 2225–2229, 2022.

[6] S. Mukherjee and J. Lee, "Edge Computing-Enabled Cell-Free Massive MIMO Systems," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2884–2899, 2020.

[7] G. Femenias and F. Riera-Palou, "Mobile Edge Computing Aided Cell-Free Massive MIMO Networks," *IEEE Trans. on Mobile Comput.*, vol. 23, no. 2, pp. 1246–1261, 2024.

[8] G. Interdonato and S. Buzzi, "Joint Optimization of Uplink Power and Computational Resources in Mobile Edge Computing-Enabled Cell-Free Massive MIMO," *IEEE Trans. Commun.*, vol. 72, no. 3, pp. 1804–1820, 2024.

[9] T. V. Thai, M. T. P. Le, H. V. Nguyen, and O.-S. Shin, "NOMA-Aided Cell-Free Massive MIMO with MEC: A Trade-Off Between Latency and Energy Consumption," in *2024 IEEE Intl. Conf. on Consumer Electronics-Asia (ICCE-Asia)*, 2024, pp. 1–5.

[10] M. Zeng, W. Hao, O. A. Dobre, and H. V. Poor, "Delay Minimization for Massive MIMO Assisted Mobile Edge Computing," *IEEE Trans. Veh. Technol.*, vol. 69, no. 6, pp. 6788–6792, 2020.