Joint Computation and Communication Resource Allocation With NOMA and OMA Offloading for Multi-Server Systems in F-RAN

SALWA MOSTAFA 1,2, (Student Member, IEEE), CHI WAN SUNG 1, (Senior Member, IEEE), AND YONGNA GUO 1, (Student Member, IEEE)

1Department of Electrical Engineering, City University of Hong Kong, Hong Kong, SAR, China
2Faculty of Electronic Engineering, Menofia University, Menouf 32951, Egypt

Corresponding author: Chi Wan Sung (albert.sung@cityu.edu.hk)

This work was supported in part by the City University of Hong Kong under Project 7005448.

ABSTRACT Since mobile devices typically have limited computation resources, offloading computation tasks to fog access points (F-APs) is a promising approach to support delay-sensitive and computation-intensive applications. This paper considers joint computation and communication resource allocation for multiuser multi-server systems, which aims to maximize the number of users being served and minimize the total energy consumption subject to delay tolerance constraints. The joint computation and communication resource allocation problem is solved optimally for both non-orthogonal multiple access (NOMA) and orthogonal multiple access (OMA) schemes. The joint user pairing and fog access point assignment problem for NOMA is proved to be NP-hard. For both NOMA and OMA, heuristic and optimal algorithms based on graph matching are designed. The optimal algorithms, though of high complexity, allow NOMA and OMA to be compared at their full potential and serve as benchmarks for evaluating the heuristic algorithms. Simulation results show that NOMA significantly outperforms OMA in terms of outage probability and energy consumption, especially for tight delay tolerance constraints and large computational tasks. Simulation results also demonstrate that our proposed NOMA and OMA schemes significantly outperform the swap-enabled matching algorithm widely used in the literature.

INDEX TERMS Multiuser multi-server computation offloading, mobile edge computing, NOMA, OMA, weighted hypergraph matching.

I. INTRODUCTION

Modern mobile applications, such as augmented reality, face recognition, assisted driving, and interactive gaming, are increasingly computation-intensive and latency-critical [1], [2]. As mobile devices have limited power and computation resources, their computation-intensive tasks need to be offloaded to powerful computing nodes for computation. However, offloading the computation tasks introduces high latency and energy consumption [3]–[5]. To meet the latency requirement of those applications with a limited power budget, fog radio access network (F-RAN) is a promising network architecture to use. It consists of a cloud server equipped with high storage, computing, and signal processing capabilities [6]. The cloud is connected to densely deployed fog access points (F-APs) via wireless fronthaul links allowing joint processing and cooperation among multiple F-APs. The F-APs are equipped with caching and computing capabilities to bring the network functions close to mobile users. By densely deploying fog access points (F-APs) with high computing capabilities at the network edge, mobile users can offload their computation tasks to nearby F-APs for fast execution and battery power saving [7], [8].

Efficient utilization of the network resources, including both computation and communication, plays a vital role in offloading the computational tasks with low latency and energy consumption. The computation resources include the CPU computational resources at the users and the F-APs. The communication resources may include the transmission power, the channel bandwidth, and the assignment of F-APs to the users in multi-server systems. Non-orthogonal multiple access (NOMA) is one of the promising communication schemes for computation offloading as it increases the
spectrum efficiency and system capacity. NOMA adopts superposition coding at the transmitter and successive interference cancellation (SIC) at the receiver, which allows multiple users to transmit on the same time and frequency resource [9], [10]. In contrast, OMA shares the resources via time or frequency division multiple access (TDMA or FDMA).

Computation offloading is mainly divided into two categories: full offloading (a.k.a. binary offloading) and partial offloading. In this study, we focus on computationally intensive tasks that require a high number of CPU cycles for execution, and their data bits are dependent (i.e., no partitioning is allowed). Hence, the computational tasks need to be fully offloaded for execution. Computation offloading has been widely studied for single-server and multi-server systems. There are different problem formulations, and the objectives are minimizing the energy consumption under a delay constraint, minimizing the delay with a given energy budget [11], [12], or minimizing both delay and energy [13]–[15]. The key issue is to strike a balance between energy consumption and computation delay.

For single-server systems, the work in [16], [17] studied orthogonal multiple access for computation offloading. The authors in [16] focused on minimizing both the delay and energy consumption for all mobile users. The study in [17] considered minimizing the energy consumption under a delay tolerance using TDMA and orthogonal frequency division multiple access (OFDMA). The authors in [18]–[22] considered joint computation and communication resource allocation using NOMA. The resource allocation of the sub-channels, the transmission power, and the computational resource are solved separately in [18]. The studies in [19], [20] jointly optimized the resources, but the proposed NOMA user pairing and sub-channel assignment schemes have very high complexity. Besides, the OMA scheme discussed assumed fixed bandwidth allocation. The main weakness of the studies in [16]–[22] is that only a single server is considered, which does not reflect the real deployment of mobile networks.

For multi-server systems supporting multiple users, the authors in [23], [24] investigated the use of OFDMA for computation offloading. The work in [25]–[28] investigated the problem of jointly optimizing the radio and computation resources using NOMA. The studies in [25], [26] investigated minimizing the energy consumption under a delay tolerance constraint. However, the authors in [25] assumed that user association to the F-APs has already been done before allocating the computation resources and transmission power. This assumption degrades the system performance as jointly optimizing the association of the users to the F-APs with the computation resources, and transmission power can further reduce the energy consumption. The work in [26] focused on forming collaborative clusters of F-APs to serve the users, which significantly increases the computational complexity of the problem. The authors in [27] aimed at maximizing the energy efficiency under a delay tolerance constraint. The work in [28] focused on minimizing the total delay under the power budget at the users and base stations. Both studies incorporated a swap-enabled matching algorithm to solve the NOMA resource allocation problem.

Unlike single server systems studies, we considered multi-server systems and investigated the problem of user pairing and F-APs assignment. We tackle the problem of joint communication and computation resource allocation in a multi-user multi-server system. Our objective is to maximize the number of served users such that the total energy consumption from the users is minimized, subject to delay tolerance constraints. By exploiting the convexity of the sub-problems of the formulated problem, both optimal algorithms and low-complexity sub-optimal algorithms are proposed. Our numerical results demonstrated the outperformance of our proposed heuristic over the swap-enabled matching proposed in [22], [27], [28]. Our main contribution can be summarized as follows:

1) For NOMA offloading, the joint communication and computation resource allocation is investigated and solved optimally. For pairing the users and assigning them to the F-APs, a framework based on hypergraph theory is proposed. The problem is proved to be NP-hard, and a low-complexity heuristic algorithm is proposed. Simulation results show that its energy and outage performance is close to the optimal solution obtained by binary integer programming (BIP), which has high complexity. Besides, it outperforms the swap-enabled matching algorithm used in the literature.

2) For OMA offloading, a heuristic for F-AP assignment based on weighted bipartite graph matching is proposed. After associating users to F-APs, the computation resources are allocated optimally via one-dimensional search, and the bandwidth allocation problem at each F-AP is shown to be convex and can be efficiently solved. Simulation results show that our proposed method performs close to the optimal OMA solution obtained by BIP. Moreover, it outperforms the stable matching algorithm used in the literature.

3) The performance of NOMA and OMA is compared. Simulation results show that NOMA yields much lower outage probability and total energy consumption than OMA, especially when the task delay tolerance is short and the computation requirement is high. The same conclusion holds for both optimal solutions and both heuristics. For example, if the task delay tolerance is 470 ms, the NOMA heuristic reduces the total energy consumption of the OMA heuristic by 85.9%. If the required computation is \(40 \times 10^7\) CPU cycles, the NOMA heuristic reduces the total energy consumption of the OMA heuristic by 78%. Simulation results also demonstrate that our proposed NOMA and OMA schemes significantly outperform the swap-enabled matching and stable matching algorithms widely used in the literature.
The rest of the paper is organized as follows. In Section II, we state our system model and present our uplink schemes. In Section III, we state our problem formulation and present our proposed solutions. In Section IV, we investigate the user pairing and F-AP assignment problem for NOMA and OMA uplink schemes. Section V provides our simulation model and results. Finally, we conclude the paper in Section VI.

II. SYSTEM MODEL

In this section, we first describe the network architecture and the channel model for an F-RAN. Then, we will present two multiple access schemes and their achievable rates.

A. NETWORK ARCHITECTURE AND CHANNEL MODEL

Consider the uplink transmission in an F-RAN, which consists of a cloud server, $M$ F-APs each equipped with $L$ antennas and a computing server, and $N$ users each with single-antenna devices as shown in Fig 1. Denote the index sets of F-APs, antennas and users by $\mathcal{M} \triangleq \{1, 2, \ldots, M\}$, $\mathcal{L} \triangleq \{1, 2, \ldots, L\}$ and $\mathcal{N} \triangleq \{1, 2, \ldots, N\}$, respectively. We assume that adjacent F-APs operate on different frequency bands to avoid inter-cell interference. Each F-AP has channel bandwidth $W$ and a central processing unit (CPU) with computational capacity $C$ CPU cycles per second. Each user $n \in \mathcal{N}$ transmits with power $p_n$ and has a maximum transmit power $P_{\text{max}}$. Each user has a computational task of parameters $(S_n, C_n, D_n)$, where $S_n$ is the computational task size in bits, $C_n$ is the number of CPU cycles required to complete the execution of the task, and $D_n$ is the task delay tolerance in seconds. Due to the limited computational capability and battery power of users, their computational tasks are offloaded to the F-APs for execution, where task splitting is not allowed. Since the F-APs have higher transmit power than the users and the size of a computational result is typically much smaller than the corresponding uploaded data, we ignore the download time. The execution time of a task on an F-AP is equal to $C_n/f_n$, where $f_n$ (cycle/s) is the CPU frequency allocated to user $n$. Thus, the sum of the offloading time and the execution time has to be no larger than $D_n$.

We consider a time-slotted system and adopt the block fading channel model. During a time slot, all channel gains stay constant and the users offload their computational tasks in parallel to their associated F-APs. The uplink channel is thus modeled as a time-invariant Gaussian channel with additive white Gaussian noise vector $z_m \sim \mathcal{CN}(0, \sigma^2 I_L)$ at each F-AP $m$, where $\sigma^2 = \mathbb{E}[w]/N_0/2$ is the noise power spectral density. The channel gain vector between user $n$ and F-AP $m$ is given by $h_{mn} \in \mathbb{C}^{L \times 1}$ for $n \in \mathcal{N}$ and $m \in \mathcal{M}$. Let the $L \times N$ matrix $H_m \triangleq [h_{mn}]$ be the channel gain matrix at each F-AP $m$. For convenience, important symbols are summarized in Table 1.

B. UPLINK MULTIPLE ACCESS SCHEMES

We consider two uplink transmission schemes, NOMA and OMA, for computational task offloading. Following the 3GPP standard [29], we allow at most two users to be superposed in NOMA, which can reduce error propagation and decoding complexity of SIC. To offload the tasks from the $N$ users, we pair them up and assign each pair to an F-AP, where an F-AP serves at most a pair of users. For comparison purpose, we also assume for OMA offloading that two users are assigned to an F-AP.

1) NOMA

Assume users $n$ and $n'$ are paired and assigned to F-AP $m$, and user $n$ has higher channel gain than user $n'$, i.e., $|h_{mn}|^2 \geq |h_{mn'}|^2$. The paired users $(n, n')$ transmit to F-AP $m$ on the whole channel bandwidth concurrently. The received signal at F-AP $m$ is thus

$$y_m = H_m[n, n']x_m + z_m,$$

where $H_m[n, n']$ is the submatrix obtained from $H_m$ by preserving the $n$-th and $n'$-th columns with the $n$-th column putting as the first column and the $n'$-th column the second column. In addition, $x_m \defined (\sqrt{p_{n,m}}x_n, \sqrt{p_{n',m}}x_{n'})$ represents the

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{M}$</td>
<td>Set of F-APs indices</td>
</tr>
<tr>
<td>$\mathcal{N}$</td>
<td>Set of users indices</td>
</tr>
<tr>
<td>$\mathcal{L}$</td>
<td>Set of antennas indices</td>
</tr>
<tr>
<td>$\mathcal{F}$</td>
<td>Set of replicated F-APs</td>
</tr>
<tr>
<td>$\mathcal{V}$</td>
<td>Set of vertices</td>
</tr>
<tr>
<td>$E$</td>
<td>Set of hyperedges</td>
</tr>
<tr>
<td>$H_{mn}$</td>
<td>Channel gain matrix of F-AP $m$</td>
</tr>
<tr>
<td>$h_{mn}$</td>
<td>Channel gain vector between F-AP $m$ and user $n$</td>
</tr>
<tr>
<td>$M$</td>
<td>Number of F-APs</td>
</tr>
<tr>
<td>$N$</td>
<td>Number of Users</td>
</tr>
<tr>
<td>$L$</td>
<td>Number of Antennas</td>
</tr>
<tr>
<td>$P_{\text{max}}$</td>
<td>User Maximum Power</td>
</tr>
<tr>
<td>$\mathbb{R}_0$</td>
<td>Noise power spectral density</td>
</tr>
<tr>
<td>$W$</td>
<td>Channel Bandwidth</td>
</tr>
<tr>
<td>$F$</td>
<td>F-AP Computational Capacity</td>
</tr>
<tr>
<td>$S_n$</td>
<td>Task Size</td>
</tr>
<tr>
<td>$C_n$</td>
<td>Task’s Computational Requirement</td>
</tr>
<tr>
<td>$D_n$</td>
<td>Task’s Delay Tolerance</td>
</tr>
<tr>
<td>$p_n$</td>
<td>Task offloading time</td>
</tr>
<tr>
<td>$f_n$</td>
<td>Transmit power of user $n$</td>
</tr>
<tr>
<td>$\lambda_n$</td>
<td>CPU frequency allocated to user $n$</td>
</tr>
<tr>
<td>$z_m$</td>
<td>Additive white Gaussian noise vector at F-AP $m$</td>
</tr>
</tbody>
</table>
transmit signals of the two users. Note that we use the notation \((a_1, a_2, \ldots, a_l)\) to represent a column vector of length \(l\).

To decode their signals successfully at F-AP \(m\), the linear decorrelator receiver with successive interference cancellation (SIC) is used, where we assume \(L \geq 2\) for successful decorrelator decoding. Consider the orthogonal complement of the one-dimensional space spanned by \(h_n\). Let \(Q^m\) be an \((L - 1) \times L\) matrix, the rows of which form an orthonormal basis of its. In particular, when \(L = 2\), \(Q^m\) degenerates to a row vector given by

\[
Q^m = \frac{[h_{n,2} - h_{n,1}]}{\|h_n\|}.
\]

The decorrelator projects \(h_n\) onto it by \(Q^m h_n\) and then uses a matched filter for detection, yielding an SNR of \(\|Q^m h_n\|^2 p_n / \sigma^2\). After detecting user \(n\)’s signal, the receiver applies SIC to subtract it from the received signal. Therefore, the signal of user \(n\)’s signal (i.e., the weak user) can be decoded without interference. The achievable data rates of users \(n\) and \(n'\) are, respectively,

\[
R_{n}^{\text{NOMA}} = W \log_2(1 + \frac{\|Q^m h_n\|^2}{\sigma^2} p_n), \quad (1)
\]

\[
R_{n'}^{\text{NOMA}} = W \log_2(1 + \frac{\|h_n\|^2}{\sigma^2} p_{n'}). \quad (2)
\]

2) OMA

Assume users \(n\) and \(n'\) are paired and assigned to F-AP \(m\). The system bandwidth at F-AP \(m\) is divided into two sub-channels with bandwidth \(\alpha_n W, \alpha_n' W\) for users \(n\) and \(n'\), respectively, such that \(\alpha_n + \alpha_n' = 1\). Each user offloads his computational task on a distinct sub-channel. Thus, the achievable data rate of any user, say \(n\), on F-AP \(m\) is

\[
R_{n}^{\text{OMA}} = \alpha_n W \log_2(1 + \frac{\|h_n\|^2}{\alpha_n \sigma^2} p_n). \quad (3)
\]

The objective of this work is to first maximize the number of served users and then minimize the total energy consumption, subject to the delay tolerance constraints. We divide the problem into two sub-problems: the resource allocation problem for users \(n\) and \(n'\) and then minimize the total energy consumption, subject to the delay tolerance constraints.

We divide the problem into two sub-problems: the resource allocation problem for users \((n, n')\) on F-AP \(m\) can be formulated as follows:

\[
E_{\text{Scheme}}(n, n') \triangleq \min_{p, f} P_n S_n + P_{n'} S_{n'}
\]

subject to \(C1: \frac{S_n}{R_n} + \frac{C_n}{f_n} \leq D_n\),

\[
C2: \frac{S_{n'}}{R_{n'}} + \frac{C_{n'}}{f_{n'}} \leq D_{n'},
\]

\[
C3: f_n + f_{n'} \leq F,
\]

\[
C4: p_n, p_{n'} \leq p_{\text{max}},
\]

\[
C5: p_n, p_{n'} \geq 0,
\]

\[
C6: f_n, f_{n'} \geq 0,
\]

where \(p \triangleq (p_n, p_{n'})\), \(f \triangleq (f_n, f_{n'})\) are, respectively, the power and CPU frequency allocation vectors for paired users \(n\) and \(n'\), and “Scheme” refers to OMA or NOMA. When a result applies to both OMA and NOMA, we may skip the superscript for brevity. Note that \(E_m(n, n')\) is the minimum transmit energy required when assigning users \(n\) and \(n'\) to F-AP \(m\).

\[\text{Lemma 1: There exists an optimal solution to (4), and at any optimal solution, C1, C2, and C3 must all hold with equality.}\]

\[\quad \text{Proof: See Appendix VI-A.}\]

The resource allocation problem in (4) can be reduced to a power allocation problem parameterized by a single variable \(f_n\). To see this, suppose \(f_n\) is a fixed value. Since C3 must hold with equality, we can set \(f_{n'} = F - f_n\). C1 and C2 become \(S_n / R_n \leq D_n \triangleq D_n - C_n / f_n\) and \(S_{n'} / R_{n'} \leq D_{n'} \triangleq D_{n'} - C_{n'} / (F - f_n)\), respectively. In other words, C1 and C2 reduce to

\[
C1': R_n \geq S_n / D_n \triangleq r_n,
\]

\[
C2': R_{n'} \geq S_{n'} / D_{n'} \triangleq r_{n'},
\]

where \(r_n\) and \(r_{n'}\) are the minimum required data rates for task offloading. Given a value of \(f_n\), we call the minimization problem in (4) with \(C1', C2', C4,\) and \(C5\) the power allocation subproblem. It is clear that \(C1'\) and \(C2'\) must hold with equality at an optimal solution. If the power allocation subproblem can be solved, then the whole problem can be solved by determining the optimal value of \(f_n\) within the interval \([C_n / D_n, F]\). For the NOMA case, the optimal value of \(f_n\) can be obtained by convex optimization techniques while for the OMA case, it can be obtained by one-dimensional grid search.

\[\text{A. NOMA RESOURCE ALLOCATION}\]

For NOMA, the optimal solution to the power allocation subproblem can be explicitly found:

\[
\begin{align*}
    p_n^{\text{NOMA}}(m) &= \frac{\sigma^2 (2^{f_n / W} - 1)}{\|Q^m h_n\|^2}, \\
    p_{n'}^{\text{NOMA}}(m) &= \frac{\sigma^2 (2^{f_{n'} / W} - 1)}{\|h_n\|^2}.
\end{align*}
\]

Based on (4) and (5), we can obtain the minimum sum energy as a function of \(f_n\), denoted by \(E_{\text{NOMA}}(f_n)\).

\[\text{Theorem 2: The minimum sum energy, } E_{\text{NOMA}}(f_n), \text{ is a convex function of } f_n.\]

\[\quad \text{Proof: See Appendix VI-B.}\]
Since the explicit form of $E^*_{\text{OMA}}(f_n)$ is known, and the function is convex, the optimal value of $f_n$ can be efficiently determined by standard convex optimization techniques.

**B. OMA RESOURCE ALLOCATION**

First, assume $f_n$ is given and $f_n' = F - f_n$. For OMA, according to (3), the optimal power of any user, say $n$, is given by

$$p_{\text{OMA}(m)}^n = \frac{\alpha_n \sigma^2 (2n/(\alpha W) - 1)}{\| h_n^n \|^2}, \quad (6)$$

where $\alpha_n$ can be optimized to minimize the total energy consumption. Assume users $n$ and $n'$ are paired and assigned to F-AP $m$. According to (6), for $i \in \{n, n'\}$, we define

$$p_i(\alpha) \triangleq \frac{\sigma^2 \alpha}{\| h_n^n \|^2} \left[ \exp \left( \frac{r_i \ln 2}{\alpha W} \right) - 1 \right],$$

which is the power of user $i$, expressed as a function of a bandwidth sharing factor $\alpha \in (0, 1]$. The channel bandwidth optimization subproblem (OMA-OPT) can be formulated mathematically as follows:

$$\begin{align*}
\min_{\alpha_n, \alpha_{n'}} & \quad p_n(\alpha_n)D_n + p_{n'}(\alpha_{n'})D_{n'}, \\
\text{subject to} & \quad C1: \alpha_n + \alpha_{n'} = 1, \\
& \quad C2: \alpha_n, \alpha_{n'} \geq \epsilon, \\
& \quad C3: p_n(\alpha_n), \quad p_{n'}(\alpha_{n'}) \leq P_{\text{max}},
\end{align*}$$

where $\epsilon$ is an arbitrarily small positive constant.

**Theorem 3:** The OMA-OPT problem is convex.

**Proof:** See Appendix VI-C.

To jointly optimize the computation and communication resources, we may use, for example, the one-dimensional grid search to find the optimal value of $f_n$ within the interval $[C_n/D_n, F]$. For each value of $f_n$, we obtain $D_n, D_{n'}, r_n$ and $r_{n'}$, from the constraints C1 and C2 in problem (4). Afterward, we can solve the bandwidth allocation problem, OMA-OPT, to optimize the allocated bandwidth and power. By Lemma 1, the energy minimization problem for the OMA scheme is reduced to minimizing the following single-valued function:

$$E^*_{\text{OMA}}(f_n) = \frac{p_n^*(D_n f_n - C_n)}{f_n} + \frac{p_{n'}^*(D_{n'}(F - f_n) - C'_{n'})}{(F - f_n)}, \quad (7)$$

where $p_n^*$ and $p_{n'}^*$ are obtained for each value of $f_n$ by solving the convex subproblem OMA-OPT. Empirically, we found that $E^*_{\text{OMA}}(f_n)$ is unimodal in $f_n$, as shown in Fig. 2 for a particular instance. Thus, the golden section search method, which is an efficient searching technique, can be used [30]. Our simulation results, to be presented in a later section, show that the golden section search has the same performance as the more time-consuming one-dimensional grid search.

**IV. USER PAIRING AND ASSIGNMENT PROBLEM**

In this section, we consider the user pairing and assignment optimization problem for both NOMA and OMA. For the NOMA scheme, it is related to the well-known problem called weighted 3-uniform hypergraph matching. While both NOMA and OMA can be solved optimally by binary integer programming (BIP), we propose heuristic algorithms to reduce the computation time. Throughout this paper, we assume $M$ is proportional to $N$.

**A. NOMA MATCHING**

First, we formulate the user pairing and assignment problem for the NOMA scheme as a hypergraph matching problem. The following concept from graph theory is needed [31]:

**Definition 1:** A weighted hypergraph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ is a set of vertices $\mathcal{V}$ and a collection $\mathcal{E}$ of subsets of the vertex set called hyperedges with a weight $w_e$ assigned to each $e \in \mathcal{E}$. A hypergraph is called $k$-uniform if each hyperedge has exactly $k$ vertices. The degree of a vertex is the number of hyperedges to which it belongs. A matching on $\mathcal{H}$ is a subset of $\mathcal{E}$, in which every two hyperedges are disjoint.

We construct a hypergraph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ such that the vertex set $\mathcal{V} = \mathcal{N} \cup \mathcal{M}$ represents the union of the set of users and the set of F-APs. The subset $[n, n', m] \subseteq \mathcal{V}$, where $n, n' \in \mathcal{N}$ (with $n \neq n'$) and $m \in \mathcal{M}$ form a hyperedge if users $n$ and $n'$ can be paired and assigned to F-AP $m$ and their computational tasks can be offloaded and executed within the delay tolerance. The associated energy is given by $E_m(n, n')$. Let $E_{\text{max}}$ be the maximum energy among all hyperedges. A non-negative weight $w_{[n, n', m]} \triangleq E_{\text{max}} - E_m(n, n')$ is assigned to each hyperedge $[n, n', m] \in \mathcal{E}$. The decision version of the problem is formally stated as follows:

**Problem:** NOMA-Matching

**Instance:** A 3-uniform hypergraph $\mathcal{H}(\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \mathcal{N} \cup \mathcal{M}$ and $\mathcal{E}$ consists of $[n, n', m]$ if it is feasible to pair up users $n$ and $n'$ and assign them to F-AP $m$ under NOMA scheme with $L$ antennas at each F-AP.

**Question:** Given an integer $k$, is there a matching with size greater than or equal to $k$?

We are going to show that it is NP-complete by reduction from the classical NP-complete problem called 3-dimensional matching, stated below:

**Problem:** 3D-Matching

**Instance:** Three finite disjoint sets, $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{Z}$ are given. In addition, $\mathcal{T}$, a subset of $\mathcal{X} \times \mathcal{Y} \times \mathcal{Z}$, is also given.
Question: Let $I \subseteq T$ be a 3-dimensional matching if for any two triples $(x_1, y_1, z_1), (x_2, y_2, z_2) \in I$, we have $x_1 \neq x_2, y_1 \neq y_2, z_1 \neq z_2$. Given an integer $k$, is there a 3-dimensional matching with size greater than or equal to $k$?

Theorem 4: NOMA-Matching is NP-complete.

Proof: Given an instance of 3D-Matching, let $N = \mathcal{X} \cup \mathcal{Y}$ and $M = \mathcal{Z}$. Let $F$ be very large so that the computation delay can be neglected. Furthermore, let all users have the same value for $S_n/D_n$ so that they all have the same minimum rate requirement, i.e., $r_n = r$ for all $n \in N$. The sets $\mathcal{X}$ and $\mathcal{Y}$ are interpreted as the set of weak users and the set of strong users, respectively. Let $L = D + 2$, where $D = |\mathcal{X} \times \mathcal{M}|$. Furthermore, let $\{h^m_n : n \in \mathcal{X}, m \in \mathcal{M}\}$ be the natural basis for the subspace of the first $D$ dimensions, and let the $(D + 1)$-th component of each of them be 1. Divide each of them by $\sqrt{2}$ such that they all become unit-norm vectors. Following the second equation in (5), we let $P_{\text{max}} = \sigma^2(2I/W - 1)$ so that a weak user has just enough power to communicate with any F-AP and any two users in $\mathcal{X}$ cannot be paired up since their channel gain vectors are non-orthogonal. For each $n \in \mathcal{Y}$ and $m \in \mathcal{M}$, let $\|h^m_n\| = 1 + \delta$, where $\delta$ is an arbitrarily small positive constant. Note that only the norm of $h^m_n$ is fixed while the exact vector is to be determined. Let the $(D + 2)$-th component of each of them be non-zero so that any two users in $\mathcal{Y}$ cannot be paired up.

For each $n \in \mathcal{Y}$ and $m \in \mathcal{Z}$, we check whether $(n', n, m) \in T$ for each $n' \in \mathcal{X}$. If so, we let $h^m_{n'}$ be any vector orthogonal to $h^m_n$, so that there is no mutual interference between them; the use of decorrelator is not needed and the setting of $P_{\text{max}}$ is large enough to assign the pair $(n', m)$ to F-AP $m$. Otherwise, we let $h^m_{n'}$ be any vector non-orthogonal to $h^m_n$, so that the decorrelator will reduce the effective SNR of user $n$, making it infeasible to pair up users $n'$ and $n$ since $\delta$ is arbitrarily small. With such an assignment, it is clear that users $n'$ and $n$ cannot be paired up and assigned to F-AP $m$ if and only if $(n', n, m) \in T$. It is also clear that the answer to the given instance of 3D-Matching is Yes if and only if the answer to the constructed instance of NOMA-Matching is Yes. Hence, NOMA-Matching is NP-complete.

Example 1: In this example, we demonstrate how to construct an instance of NOMA-Matching. Given the instance of 3D-Matching where $\mathcal{X} = \{1, 2\}$, $\mathcal{Y} = \{3, 4\}$, $\mathcal{Z} = \{5, 6\}$, and $T = \{(1, 3, 5), (1, 4, 5), (2, 4, 6)\}$, we construct an instance of NOMA-Matching with four users and two F-APs, indexed by $N = \{1, 2, 3, 4\}$ and $M = \{5, 6\}$, respectively. Each user has one single antenna while each F-AP has $L = |\mathcal{X} \times \mathcal{M}| + 2 = 6$ antennas. Choose the data rate, noise power, and maximum power so that $P_{\text{max}} = \sigma^2(2I/W - 1)$. Let $h^5_1 = \frac{1}{\sqrt{2}}(1, 0, 0, 0, 1, 0), h^5_2 = \frac{1}{\sqrt{2}}(0, 1, 0, 0, 1, 0), h^5_2 = \frac{1}{\sqrt{2}}(0, 0, 1, 0, 1, 0),$ and $h^6_2 = \frac{1}{\sqrt{2}}(0, 0, 0, 1, 1, 0)$.

The next step is to determine the gain vectors from the near users $\mathcal{Y} = \{3, 4\}$ to the F-APs. We need to ensure that $h_5^3$ is orthogonal to $h_5^1$ but non-orthogonal to $h_5^2$. Therefore, we choose $h_5^3 = \frac{1}{\sqrt{2}}(0, 0, 1, 0, 0, 1)$, where $1^{+} \triangleq 1 + \delta$ for some small value of $\delta$. Similarly, as we need to ensure that $h_6^3$ is non-orthogonal to both $h_6^1$ and $h_6^2$, we choose $h_6^3 = \frac{1}{\sqrt{2}}(0, 0, 1, 0, 0, 1)$. By the same method, we choose $h_4^3 = \frac{1}{\sqrt{2}}(0, 0, 1, 0, 0, 1)$ and $h_4^3 = \frac{1}{\sqrt{2}}(0, 0, 1, 0, 0, 1)$. This completes the reduction procedure. It can be seen that the answer to the given instance of 3D-Matching is yes if and only if the answer to the constructed instance of NOMA-Matching is yes.

Remark 1: Since the decision version of our problem is NP-complete, the corresponding optimization version, which aims to maximize the number of served users, is NP-hard.

In general, there are many solutions that can maximize the number of served users. Among those solutions, we would like to find the one that minimizes the total transmit energy. We now show that this problem can be reduced to the maximum weighted hypergraph matching problem, which aims to find a matching in a given weighted hypergraph that maximizes the sum of weights. To perform the reduction, we add a large constant value

$$Q \geq \sum_{[n, n', m] \in E} w_{[n, n', m]}$$

to the weights of all hyperedges. The new weight becomes

$$w_{[n, n', m]} = w_{[n, n', m]} + Q$$

for each $[n, n', m] \in E$. Since all hyperedges have large weights, the maximum weighted matching must be a matching of maximum cardinality.

The maximum weighted hypergraph matching problem can be formulated as a BIP problem [32] as follows:

$$\text{max}_{[x_e \in \mathbb{E}]} \sum_{e \in E} w_e x_e$$

subject to

$$C_1 : \sum_{e \in \mathbb{E}(v)} x_e \leq 1, \quad \forall v \in \mathcal{V},$$

$$C_2 : x_e \in \{0, 1\}, \quad \forall e \in \mathcal{E},$$

where the objective function in (8) represents the sum of the weights of the chosen hyperedges in the matching, $x_e$ is a binary decision variable indicating whether the hyperedge $e$ belongs to the matching, and $\mathbb{E}(v) \subseteq \mathcal{E}$ is the set of hyperedges containing vertex $v$. The problem can be solved optimally via the branch and bound (BnB) method [32], but the computational complexity grows exponentially with the problem size.

To solve the problem with low computational complexity, we propose a heuristic algorithm to find a suboptimal solution. It works as follows: First, we assign to each F-AP the user with the best channel gain to it and remove the user from the set of users. The set $\mathcal{A}$ denotes the set of paired F-APs and users, while the set $\mathcal{N}^c$ denotes the set of users that have not been assigned to any F-AP. Next, we form a weighted bipartite graph $G'((\mathcal{M}, \mathcal{N}^c), \mathcal{E}')$. There exists an edge $(m, n') \in \mathcal{E}'$ between F-AP $m$ and user $n'$ if user $n'$ can be
Algorithm 1: Heuristic Algorithm for NOMA-Matching

Input: A set of users $\mathcal{N}$, a set of F-APs $\mathcal{M}$, channel gain matrices $H_1, H_2, \ldots, H_M$.

Output: NOMA-Matching $\mathcal{U}$;

1: $\mathcal{A} := \emptyset$ and $\mathcal{N'} := \mathcal{N}$;
2: for $m = 1 : |\mathcal{M}|$ do
3: $n := \arg \max_{n \in \mathcal{N'}} |h_n|^2$;
4: $\mathcal{A} := \mathcal{A} \cup \{(m, n)\}$;
5: $\mathcal{N} := \mathcal{N} \setminus \{n\}$;
6: end for
7: Construct a weighted bipartite graph $G'(\mathcal{M}, \mathcal{N'}, \mathcal{E'})$ with $w_e$ for each $e \in \mathcal{E'}$;
8: Find the maximum weighted matching $\mathcal{W}$ for $G'$;
9: $\mathcal{U} := \{(m, n, n') | (m, n) \in \mathcal{A}$ and $(m, n') \in \mathcal{W}\}$;
10: return $\mathcal{U}$;

paired with user $n$ assigned before to F-AP $m$ and both can offload and execute their computational tasks within the delay tolerance. The weight of each edge is defined similarly as the weight for each hyperedge. Afterward, we find the maximum weighted matching $\mathcal{W}$ for the bipartite graph. Finally, the solution to the optimization version of NOMA-Matching is given by $\mathcal{U} = \{(m, n, n') \in \mathcal{M} \times (\mathcal{N} \setminus \mathcal{N'}) \times \mathcal{N'} | (m, n) \in \mathcal{A}$ and $(m, n') \in \mathcal{W}\}$.

The pseudo-code is described in Algorithm 1. The for-loop in Step 2 runs for $O(MN)$ times. Constructing the weighted bipartite graph in Step 7 takes $O(MN)$. The maximum weighted matching in Step 8 can be optimally solved by the Hungarian algorithm [33] in $O(N^3)$. Step 9 takes $O(M)$ times. The overall time complexity of Algorithm 1 is $O(N^3)$.

B. OMA MATCHING

For the OMA scheme, the problem can also be formulated as maximum weighted hypergraph matching and solved by BIP. The weight of each hyperedge $(m, n, n')$, if feasible, can be obtained by allocating the computation resources via one-dimensional grid search or golden section search and optimizing the bandwidth allocation OMA-OPT. Due to its high complexity, we propose a heuristic as follows: First, we set $f_n = f_{n'} = F/2$, $\alpha_n = \alpha_{n'} = 0.5$ for all possible pairs of users $(n, n')$ associated with any F-AP, which is the same as dividing the computation resources and the channel bandwidth into two equal halves at each F-AP. Then, we create a set $\mathcal{F}$ of cardinality $\mathcal{M}$, which represents a set of virtual F-APs replicating the original F-APs. A weighted bipartite graph $G(\mathcal{N}', \mathcal{M}', \mathcal{E})$ is then constructed, where $\mathcal{N}'$ represents the set of users and $\mathcal{M}' \triangleq \mathcal{M} \cup \mathcal{F}$ represents the set of the original F-APs and the replicated F-APs. The set of edges is defined as

$$\mathcal{E} = \{(n, m) \in \mathcal{N'} \times \mathcal{M'} | R_{OMA}(m) \geq r_n\}.$$ 

Let $E_n(n) = p_{n, OMA(m)} D_n$ be the transmit energy associated with each edge $(n, m)$ and $E_{\text{max}}$ be the maximum energy among all edges. A non-negative weight $w_{(n, m)} = E_{\text{max}} - E_m(n)$ is assigned to each edge $(n, m) \in \mathcal{E}$. To find the association of users to the F-APs such the number of served users is maximized and the total energy consumption is minimized, the weight of each edge is re-defined to be

$$w'_{(n, m)} = w_{(n, m)} + Q,$$

where $Q \geq \sum_{n,n'\in\mathcal{E}} w_{(n, m)}$. The optimal association under the bipartite graph can be obtained via maximum weighted matching in $O(N^3)$. After we obtain the association of the users to the F-APs, for each association, we allocate the computation resources via one-dimensional grid search or golden section search and solve the OMA-OPT problem by applying the CVX solver to optimize the allocated bandwidth between the associated users.

Remark 2: Our proposed hypergraph framework can be generalized to group any number of users by applying $k$-dimensional matching [34], where $k - 1$ is the number of users in a group. It also can be applied to assign more than one group of users to an F-AP by dividing the bandwidth of each F-AP into multiple sub-carriers and considering each subcarrier a virtual F-AP. The computation resources, in this case, can be divided equally between the sub-carriers.

V. SIMULATION MODEL AND RESULTS

In this section, simulation results are presented to evaluate the performance of our proposed schemes. We consider a single cell, in which the F-APs and users are randomly distributed inside the cell according to the homogeneous Poisson point process (PPP). The fading models and simulation parameters are listed in Table 2 [35]. We consider the same task size and task delay tolerance for all users. We assume the number of users doubles the number of F-APs, i.e., $N = 2M$. A user is said to be in outage if his computational task cannot be offloaded and executed within the delay tolerance. Outage probability is defined as the average proportion of users that are in outage. The total energy consumption in the system is taken over those instances in which no users are in outage for all schemes. The results are obtained by averaging over 500 random realizations.

<table>
<thead>
<tr>
<th>Table 2. Simulation parameters.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameters</td>
</tr>
<tr>
<td>Cell Radius</td>
</tr>
<tr>
<td>Number of F-APs ($M$)</td>
</tr>
<tr>
<td>Number of Users ($N$)</td>
</tr>
<tr>
<td>Number of Antennas ($E_r$)</td>
</tr>
<tr>
<td>Distance-dependent Path loss</td>
</tr>
<tr>
<td>Small Scale Fading</td>
</tr>
<tr>
<td>User Maximum Power ($P_{\text{max}}$)</td>
</tr>
<tr>
<td>Channel Bandwidth ($W$)</td>
</tr>
<tr>
<td>F-AP Computational Capacity ($F$)</td>
</tr>
<tr>
<td>Task Size ($S$)</td>
</tr>
<tr>
<td>Task’s Computation Requirement ($C$)</td>
</tr>
<tr>
<td>Task’s Delay Tolerance ($D$)</td>
</tr>
<tr>
<td>Noise Power Spectral Density</td>
</tr>
</tbody>
</table>
The schemes proposed in [27], [28] are used for comparison with our proposed NOMA and OMA schemes, respectively. The swap-enabled matching algorithm is used in both studies to solve the user pairing and F-APs assignment problem. The main idea behind the swap-enabled matching is to find a stable matching, where the users are assigned to the F-APs such that no user prefers another F-AP than the one currently assigned to him, and the F-APs do not prefer other users. However, in NOMA, the energy consumption of users depends on the other users assigned to the same F-AP. Thus, after finding the initial stable matching, a swap stage is executed to try every possible swap between the users and accept the swap that reduces the energy consumption of both users without increasing the energy consumption of other users.

For fair comparison with [27], we set the number of sub-channels on each F-AP to one and ignore the inter-cell interference term in equation (1). The optimal power and computation resource allocation method used under our scheme is adopted. The reason is that the proposed resource allocation method in [27] relies on the non-convexity of the power allocation due to the interference term, which is not considered in our work. For fair comparison with [28], we assume that each user has only one task to be offloaded, the downlink data is set to zero, each F-AP has two sub-channel, and the computation resources are divided equally between the paired users. Their proposed scheme is adopted to be compared with our OMA scheme, where the energy consumption of the users is independent of the other users. Thus, the swap stage is not needed, and the stable matching stage only is executed.

Fig. 3 shows the performance of both one-dimensional grid search and golden section search with tolerance $10^{-10}$, $P_{\text{max}} = 80$ mW, $L = 2$, $M = 7$, $N = 14$, $F = 9$ GHz, and $C = 4 \times 10^7$ CPU cycles.

OMA schemes. The reason is that NOMA allows each user to transmit over the whole channel bandwidth while OMA shares the channel bandwidth between the users. Thus, the users under NOMA can achieve the required data rates with lower transmit power by sending their data over a larger bandwidth. As we can see, our proposed heuristic NOMA scheme, with low computational complexity, gives a reasonably good performance compared with the optimal solution obtained by the BIP. For OMA, both fixed and optimized bandwidth and computation allocation schemes have the same outage as both use bipartite matching for user assignment.

As we can also observe in Fig. 4, the schemes in [27], [28] give high outage probability compared to our proposed schemes as the authors did not consider maximizing the number of served users in their problem formulation and proposed solution as in our work. Besides, our proposed OMA schemes and heuristic NOMA implemented weighted bipartite matching, which solves the subproblem of maximizing the number of served users and minimizing the energy consumption optimally compared to the swap-enabled matching.
Fig. 5 demonstrates the task delay tolerance effect on the total energy consumption. As we can see, as the task delay tolerance increases, the required data rates are low, so the total energy decreases. As we can observe, NOMA schemes outperform the OMA schemes in minimizing the total energy consumption. Moreover, NOMA schemes are more beneficial when the task delay tolerance is short. For example, if the task delay tolerance is 0.47 sec, the BIP NOMA, the heuristic NOMA, and the swap-enabled matching schemes outperform the BIP OMA scheme by 83.5%, 79.6%, and 50% respectively. For OMA, optimizing the bandwidth and computation resource allocation gives better performance than allocating fixed bandwidth and computation resources to the users as it adapts the shared channel bandwidth and computation resource based on the channel conditions of users. Besides, our proposed optimized OMA scheme gives a reasonably good performance when compared with the optimal BIP OMA scheme. At 0.47 sec, the optimized OMA scheme increases the energy consumption by 7% only.

As we can also see in Fig. 5, although the power and computation resources are allocated optimally under both heuristic NOMA, and the scheme proposed in [27], our heuristic outperforms the swap-enabled matching algorithm. The reason is that our heuristic implemented weighted bipartite matching, which solves part of the assignment problem optimally. Similarly, under OMA, the bipartite matching solves the assignment problem optimally compared to the stable matching.

Fig. 6 shows the total energy consumption versus the F-AP computational capacity. We can observe that as the computational capacity increases, the total energy consumption decreases. This is because more CPU cycles can be allocated, which reduces the computation time, allowing more time for offloading and thus reducing transmission energy. We can also observe that both NOMA schemes outperform the OMA schemes in minimizing the total energy consumption. The reason is as explained before. Besides, NOMA offloading efficiently reduces the energy consumption when the computation capacity is limited compared to OMA offloading.

Fig. 7 demonstrates the total energy consumption versus the task computation requirement. Obviously, as the number of CPU cycles required to execute a task increases, the total energy consumption increases. The reason is that more time is required to execute the task, which in turn tightens the offloading time and increases the energy required for offloading the tasks. Moreover, NOMA offloading is more beneficial when the number of CPU cycles required to execute a task is high.

Fig. 8 shows the total energy consumption versus the number of antennas. Obviously, as the number of antennas increases, the total energy consumption decreases due to the power gain obtained from the spatial diversity of antennas. Moreover, NOMA offloading is more beneficial when the number of antennas is small.

VI. CONCLUSION

For multiuser multi-server computation offloading, the problem of joint computation and communication resource allocation for maximizing the number of served users and then minimizing the total energy consumption using NOMA within delay tolerance constraints is proved to be NP-hard. An optimal scheme using binary integer programming is used.
as a benchmark, and a low-complexity heuristic algorithm is proposed to solve the problem. The simulation results show that our proposed NOMA heuristic performs close to the optimal solution and significantly outperforms the OMA schemes. Besides, the OMA bandwidth allocation problem is proved to be convex, and a low complexity scheme with reasonably good performance is proposed. Simulation results also demonstrated that our proposed NOMA and OMA schemes significantly outperform the swap-enabled matching algorithm implemented in the literature.

Striking a balance between minimizing the energy consumption and delay is of great interest; thus, further study on multi-objective optimization for computation is needed. Besides, investigating how to jointly optimize the communication and computation resource allocation assuming computational capability at mobile users is an interesting topic that deserves much attention in a future study.

APPENDIX

A. PROOF OF LEMMA 1

Proof: Since the constraint set is compact and the objective function is continuous in $p$ and $f$, by the Weierstrass Theorem [36, Theorem 3.1], an optimal solution exists. Now we prove by contradiction that C1, C2, and C3 must hold with equality. Let $f_n^*, f_{n'}^*$, $p_n^*$, and $p_{n'}^*$ be an optimal solution. Assume that C1 does not hold with equality. Since $p_n/R_n$ is a strictly increasing function in $p_n$, we can reduce $p_n^*$ to a smaller value $p_n$ such that the objective function value becomes smaller without violating C1. That shows that C1 must hold with equality. By the same argument, C2 must also hold with equality. Finally, suppose C3 does not hold with equality at the optimal solution. Then we can increase both $f_n^*, f_{n'}^*$, violating that C1 and C2 must hold with equality. □

B. PROOF OF THEOREM 2

Proof: Since the constraints C1, C2 and C3 hold with equality, we have

$$R_{n}^{\text{NOMA}}(f_{n}) = \frac{S_m f_n}{D_m f_n - C_n}$$

and

$$R_{n'}^{\text{NOMA}}(f_{n'}) = \frac{S_m (F - f_n)}{D_m (F - f_n) - C_{n'}}.$$

For the optimization problem to be feasible, we must have $C_n/D_n < f_n < F - C_{n'}/D_{n'}$. Differentiating each function twice, we obtain

$$\frac{\partial^2 R_n}{\partial f_n^2} = \frac{2 S_m C_n D_n}{(D_m f_n - C_n)^3} > 0,$$

$$\frac{\partial^2 R_{n'}}{\partial f_n' ^2} = \frac{2 S_m C_n D_{n'}}{(D_m (F - f_n) - C_{n'})^3} > 0,$$

showing that they are convex functions of $f_n$.

Let $a \triangleq \ln(2)/W$, $b \triangleq \frac{\sigma^2 S_m}{\|D_m f_n^0\|^2}$, and $b' \triangleq \frac{\sigma^2 S_m}{\|D_m f_{n'}^0\|^2}$. The minimum sum energy can be written as

$$E_n^{\text{NOMA}}(f_n) = b R_n^{-1}(e^{a R_n} - 1) + b' R_{n'}^{-1}(e^{a R_{n'}} - 1). \tag{9}$$

Let $h(x) \triangleq x^{-1}(e^{ax} - 1)$. Then

$$h'(x) = \frac{1}{x^2}[(ax - 1)e^{ax} + 1].$$

If $ax \geq 1$, it is obvious that $h'(x) \geq 0$. Suppose $ax < 1$. We claim that

$$(1 - ax)e^{ax} \leq 1.$$

Taking the natural logarithm, the expression in the left-hand side becomes $\ln(1 - ax) + ax \leq -ax + ax = 0$, where the inequality follows from Lemma 2.29 in [37]. Hence, the claim is proved, and thus $h'(x) \geq 0$ for all $x \geq 0$. Differentiating it again, we have

$$h''(x) = \frac{1}{x^3}\left[e^{ax}(ax - 1)^2 + e^{ax} - 2\right],$$

for all $x \geq 0$. Let $\phi(x) \triangleq e^{ax}(ax - 1)^2 + e^{ax} - 2$. Since $\phi'(x) = a^2 x^2 e^{ax} \geq 0$ and $\phi(0) = 0$, $\phi(x)$ is monotonically increasing and thus $\phi(x) \geq 0$ for all $x \geq 0$. Hence, $h''(x) \geq 0$ for all $x \geq 0$, showing that $h(x)$ is convex.

Rewrite (9) as

$$E_n^{\text{NOMA}}(f_n) = b(h \circ R_n)(f_n) + b'(h \circ R_{n'})(f_n),$$

where $\circ$ denotes function composition. Since $h(x)$ is convex and non-decreasing while $R_n(f_n)$ and $R_{n'}(f_n)$ are both convex, each of the two terms in the right-hand side is a convex function of $f_n$ [38, p. 84]. Since $b, b' \geq 0$ and non-negative weighted sum preserves convexity, the function $E_n^{\text{NOMA}}(f_n)$ is convex in $f_n$. □

C. PROOF OF THEOREM 3

Proof: First, we need to check that the constraints form a convex set. The constraints C1 and C2 are linear. Thus, we need to check whether the region constrained by C3 is convex or not. Differentiating $p_i(\alpha)$ twice, we obtain

$$\frac{d^2 p_i(\alpha)}{d \alpha^2} = \frac{\sigma^2}{\|h_i^m\|^2} \exp \left( \frac{r_i \ln 2}{\alpha W} \right) \left[ 1 - \frac{r_i \ln 2}{\alpha W} \right] - \frac{\sigma^2}{\|h_i^m\|^2},$$

$$\frac{d^2 p_i(\alpha)}{d \alpha^2} = \frac{\sigma^2 r_i^2 \ln^2 2}{\|h_i^m\|^2 \alpha^3 W^2} \exp \left( \frac{r_i \ln 2}{\alpha W} \right) > 0,$$

since $\alpha > 0$. Hence, the feasible region is convex. Since the offloading time of tasks $\hat{D}_n$ and $\hat{D}_{n'}$ are positive and non-negative weighted sum preserves convexity, the objective function is strictly convex. □

REFERENCES


**S. Mostafa et al.: Joint Computation and Communication Resource Allocation**