Energy Efficiency of RIS-Assisted NOMA-Based MEC Networks in the Finite Blocklength Regime

Yang Yang, Yulin Hu, and M. Cenk Gursoy

***Abstract*—In this paper, we investigate a reconfigurable intelligent surface (RIS)-assisted mobile edge computing (MEC) network aiming to maximize the energy efficiency in the finite blocklength (FBL) regime under both coding length and maximum decoding error rate constraints. We first analyze the single user equipment (UE) case and propose a three-step alternating optimization algorithm to solve the problem. Extending the system model, we subsequently investigate a network with multiple UEs, in which non-orthogonal multiple access (NOMA) transmission is adopted. In this more general setting, we also conduct a convergence analysis. Furthermore, we introduce a UE-grouping scheme for hybrid NOMA-TDMA transmission and develop a dynamic CPU frequency allocation algorithm at the mobile edge computing (MEC) server. Numerical results show that the proposed algorithms solve the problem efficiently. Via numerical results, we also identify the impact of various parameters (e.g., coding blocklength, the number of RIS elements, computational resources, number of UEs) on the energy efficiency. Furthermore, with the numerical results, we verify the validity of UE grouping method and demonstrate that the proposed dynamic CPU frequency allocation can enhance the performance substantially.**

***Index Terms*—Energy efficiency, edge computing, finite blocklength regime, non-orthogonal multiple access (NOMA), reconfigurable intelligent surface (RIS).**

I. Introduction

Mobile edge computing (MEC) is an architecture that can be utilized to alleviate communication and computation bottlenecks experienced due to increasing demand on high data traffic and growing number of applications with high computational requirements [1]. In MEC, the user equipments (UEs) can fully or partially offload their services/tasks to the edge nodes of networks rather than the remote cloud center [2][3]. The MEC servers are usually deployed at the base stations (BSs) to process the users' offloaded tasks to mitigate the congestion in the network [4]. A hierarchical architecture can be further formed by the data center, BSs and UEs to improve the energy efficiency and storage capacity.

As another novel technology, reconfigurable intelligent surfaces (RISs) are considered as an effective means to improve both the spectral efficiency and coverage of wireless communication systems [5]. In particular, the propagation environment can be significantly enhanced by properly setting the phase shift matrix at the RIS [6]. This is accomplished via the meta-surface, whose phase and amplitude responses can be adjusted by a programmable controller so that the RIS can modify how the incident signal is reflected. In [7], the authors have provided an overview of the RIS technology, including its main applications in wireless communications, competitive advantages over other technologies, its hardware architecture and the corresponding new signaling models. The authors in [8] have studied an RIS-enhanced multiple-input single-output (MISO) wireless system where an RIS is deployed to assist the communication from a multi-antenna access point (AP) to a single-antenna user. Note that the phase shift matrix of the RIS should be properly adjusted since the reflected signals from various paths can be combined coherently so that it can enhance the link achievable rate at the MEC receiver [9].

Non-orthogonal multiple access (NOMA) is also increasingly being adopted as a promising technology to improve the spectral usage of the network by allowing multiple users to perform simultaneous transmissions in the same bandwidth [10]. It is expected that when NOMA transmission with successive interference cancellation (SIC) at the receiver is combined with the RIS, the wireless propagation environment will be further improved.

Ultra-reliable and low latency communication (URLLC) [11] is considered as one of the three main service categories that can satisfy the critical requirements when mission-critical and delay-sensitive applications in both communication and computation are addressed [12]. The achievable rate and decoding error probability when the coding blocklength is finite have been explicitly investigated in [13] and the authors in [14] have combined the RIS technology with short packet transmissions in the finite blocklength (FBL) regime (as required in URLLC settings) and investigated the average achievable rate and error probability. RIS-aided downlink multi-user communication from a multi-antenna BS is investigated in [15], where the authors have proposed a realistic power consumption model for RIS-based systems and analyzed the performance of the proposed methods in a realistic outdoor environment.

A promising study in [16] demonstrates that an RIS-aided MIMO system can attain the same rate performance as the benchmark massive MIMO system without employing IRS, but with much fewer active antennas/RF chains. In [17], an RIS-assisted wireless powered hybrid NOMA and time division multiple access (TDMA) network has been studied, where the authors have designed the time allocation to maximize the throughput of the network. A similar analysis is conducted in [18], where an RIS-aided wireless powered mobile edge computing (WP-MEC) system is considered and the authors have investigated which multiple access scheme among NOMA and TDMA is superior for MEC uplink offloading. Another study in [19] has revealed that RIS-assisted NOMA leads to a higher uplink sum rate compared with the RIS-assisted orthogonal multiple access (OMA). Moreover, a novel RIS-aided NOMA downlink transmission framework is proposed in [20] where the authors have utilized a deep deterministic policy gradient (DDPG) algorithm to collaboratively control multiple reflecting elements (REs) of the RIS. The authors in [20] have developed a long-term self-adjusting learning model, which differs from standard training-then-testing learning in that the intelligent agent is capable of discovering the optimal action to take for each state through exploration and exploita-

Y. Yang and M. C. Gursoy are with the Department of Electrical Engineering and Computer Science, Syracuse University, Syracuse, NY, 13244. (Email: yyang82@syr.edu, mcgursoy@syr.edu)

Y. Hu is with the School of Electronic Information, Wuhan University, 430072 Wuhan, China. (Email: yulin.hu@whu.edu.cn)

tion. These works demonstrate the interests in combining RIS with NOMA to improve the transmission either in uplink or downlink. We further notice that a study in [21] reveals an attracting scheme that can endow the system with resiliency and robustness to satisfy the stringent requirements in the URLLC scenario by deploying an IRS.

In this paper, we combine the FBL regime with the RIS-aided MEC network aiming to maximize the energy efficiency under both coding length and maximum decoding error rate constraints in a URLLC scenario, where the offloading decisions at the UEs, the RIS reflecting coefficients, the length of offloading phase and the computational resource allocations at the MEC server are critically important in achieving the highest efficiency levels. We iteratively optimize those parameters to attain an optimal energy efficiency. A major departure in this paper from aforementioned prior work is the practical considerations of the FBL regime and RIS-aided NOMA transmissions in decision-making in MEC networks, motivated by low-latency scenarios and applications such as augmented/virtual reality. Additionally, different from [17] and [18], we investigate the offloaded data bits rather than the time allocation in hybrid NOMA and OMA. Unlike [19], we aim at maximizing the energy efficiency instead of the sum rate. Our work focus on the uplink of the MEC network, while in [20] the authors analyze the downlink transmission. Finally, in contrast to our recent conference papers [22]-[23], in this paper we provide extensions to the more general scenario with multiple UEs, and conduct a detailed convergence analysis for the proposed algorithms. In particular, our main contributions are summarized as follows:

1) We describe and analyze the model when FBL coding, RIS and NOMA transmission are utilized in an MEC network.
2) We investigate the maximization of the energy efficiency under both coding blocklength and maximum decoding error rate constraints.
3) We develop three-step optimization algorithms to solve the proposed problems in both single-UE and multiple-UE cases, and analyze the convergence of the proposed approach.
4) We introduce a UE grouping method and develop a dynamic CPU frequency allocation algorithm to improve the optimization by reducing the complexity as the number of UEs increases.

The remainder of this paper is organized as follows. We describe the system model, characterize the decoding error rate as well as the signal-to-interference-plus-noise ratio (SINR) in NOMA, and provide the energy efficiency formula in Section II. In Section III, we state the optimization problem in the single-UE case and subsequently provide a three-step algorithm to address it. In Section IV, we investigate the energy efficiency maximization in the case with multiple UEs and NOMA, and propose a solution approach and analyze the convergence of the proposed algorithm. In Section V, we consider hybrid NOMA-TDMA, introduce a UE grouping method to improve the efficacy of the optimization algorithm when the number of UEs increases, and then we develop a dynamic CPU frequency allocation algorithm to better take advantage of hybrid NOMA-TDMA. Simulation results are provided in Section VI. Finally, in Section VII, we draw conclusions.

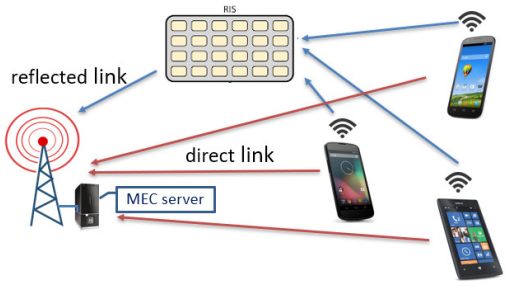


Fig. 1: An illustration of the considered MEC network.

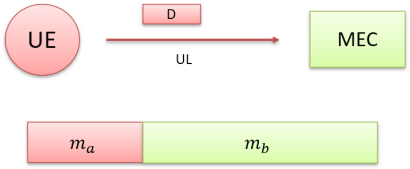


Fig. 2: System topology and frame structure.

II. System Model

An RIS-aided MEC network where the BS is equipped with an MEC server is considered in this paper. There are N UEs in the network and each UE, i.e., UE $n, \forall n \in \mathcal{N} = \{1, 2, ..., N\}$, has a single antenna. The BS has B antennas and the RIS has K reflecting components. A wireless controller is used by the BS to operate the RIS so that it is capable of dynamically adjusting the RIS phase shift matrix (i.e., the phase shift of each reflecting element). Fig. 1 depicts the network.

In this paper, each task requested by the UE requires a certain computational resource (e.g., CPU frequency) and has a latency constraint. The UE can either fully or partially offload the delay-critical task to the MEC server located at the BS by using FBL codes to complete the compute-intensive application. We consider service as an abstraction of a requested task and a three-parameter notation $S(I, T, X)$ is introduced to represent the service [2]. In this notation, I is the task-input size (in bits), T is the completion latency constraint (in seconds) and X is the computation intensity (in CPU cycles per bit).

A. FBL Transmission

In this paper, all the channels between the UE and the BS as well as the RIS are assumed to experience block fading, and thus the channels remain constant within a transmission block.

The duration of a transmission symbol is defined as T_{syb}, indicating that the delay limitation of T in seconds corresponds to $M = T/T_{\text{syb}}$ symbol periods. More specifically, T seconds or equivalently M symbol durations serve as a bound on the frame length of the service completion of the requested task. An uplink (UL) offloading phase with a length of m_a symbols and a computation phase with a length of m_b symbols (equivalent to $m_b T_{\text{syb}}$ seconds) are the two phases in a frame, as depicted in Fig. 2. The n-th UE transmits D_n bits to the MEC server via a wireless link in the UL phase, and the MEC server at the BS processes all the received requests in the computation phase. The BS will then provide the results

back to the UEs[1]. It is obvious that the total service time of the application is constrained by $m_a + m_b = M$. Following [13], the coding rate R in the FBL regime is approximated as

$$R \approx \log_2(1+\gamma) - \sqrt{\frac{V}{m_a}}\frac{Q^{-1}(\varepsilon)}{\ln 2}, \tag{1}$$

where ε is the decoding error probability, γ is the signal-to-noise ratio (SNR) or signal-to-interference-plus-noise ratio (SINR) at the receiver, Q^{-1} is the inverse function of $Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty e^{-\frac{t^2}{2}}dt$ and V is the channel dispersion defined as $V = 1-(1+\gamma)^{-2}$.

Considering $R = \frac{D}{m_a}$ as the target achievable coding rate, the decoding error probability of the transmission in the UL phase can be expressed as

$$\varepsilon \approx Q\left(\sqrt{\frac{m_a}{V}}\left(\log_2(1+\gamma) - \frac{D}{m_a}\right)\log_e 2\right). \tag{2}$$

Note that since we operate in the FBL regime, the blocklength of each frame is limited by M and the decoding error probability at the receiver is non-negligible.

B. Transmission Model with RIS

With the introduction of the RIS, there are two links in the channel from the UE to the BS: the direct link and the reflected link (UE to RIS to BS), as also depicted in Fig. 1. Consequently, we can express the channel fading vector $\boldsymbol{h}_n$ from the n-th UE to the BS as [16]

$$\boldsymbol{h}_n = \boldsymbol{h}_{d,n}^H + \boldsymbol{h}_{r,n}^H \Theta \boldsymbol{\mathcal{G}}, \tag{3}$$

where $\boldsymbol{h}_{d,n}^H \in \mathbb{C}^{1\times B}$ is the channel vector from the UE to the BS (where H in the superscript indicates *conjugate transpose*), $\boldsymbol{h}_{r,n}^H \in \mathbb{C}^{1\times K}$ is the channel vector from the UE to the RIS and $\boldsymbol{\mathcal{G}} \in \mathbb{C}^{K\times B}$ is the matrix from the RIS to the BS. Θ denotes the phase shift matrix of the RIS which is defined as $\Theta = \beta \text{diag}(e^{i\theta_1}, ..., e^{i\theta_K}) \in \mathbb{C}^{K\times K}$ where $\theta_k \in [0, 2\pi]$, $k \in \mathcal{K} = \{1, 2, ..., K\}$ and $\beta \in [0, 1]$ is the amplitude reflection coefficient and we have $\beta = 1$ in this article. Consequently, at the MEC server the received signal $\mathbf{y}$ can be written as

$$\mathbf{y} = \sum_{n=1}^{N} \boldsymbol{h}_n x_n + \boldsymbol{\eta}, \tag{4}$$

where x_n is the signal that the n-th UE transmits, with an average power of $\mathbb{E}[||x_n||^2] = P_n$, and $\boldsymbol{\eta}$ is the additive white Gaussian noise (AWGN) at the BS, e.g., $\boldsymbol{\eta} \sim \mathcal{CN}(0, \sigma^2 \mathrm{I}_N)$.

C. SINR in NOMA

In NOMA transmission, since the reflected link depends on the unknown RIS phase shift matrix Θ, according to [19], we replace Θ by an identity matrix I so that all the N UEs can be sorted in an increasing order, i.e.,

$$||\boldsymbol{h}_{d,1}^H + \boldsymbol{h}_{r,1}^H \mathrm{I}\boldsymbol{\mathcal{G}}|| \leq ||\boldsymbol{h}_{d,2}^H + \boldsymbol{h}_{r,2}^H \mathrm{I}\boldsymbol{\mathcal{G}}|| \leq ..., \leq ||\boldsymbol{h}_{d,N}^H + \boldsymbol{h}_{r,N}^H \mathrm{I}\boldsymbol{\mathcal{G}}||. \tag{5}$$

By considering the signals from all other UEs as interference, the BS will first decode the signal from the last UE N in this order. Consequently, the n-th UE's SINR in NOMA transmission is expressed as

$$\gamma_n = \frac{P_n||\boldsymbol{h}_{d,n}^H + \boldsymbol{h}_{r,n}^H \Theta \boldsymbol{\mathcal{G}}||^2}{\sum_{t=1}^{n-1} P_t||\boldsymbol{h}_{d,t}^H + \boldsymbol{h}_{r,t}^H \Theta \boldsymbol{\mathcal{G}}||^2 + \sigma^2} \tag{6}$$

for $n = 2, \ldots, N$.

By employing the SIC technique in NOMA, the SINR of UE 1 can be expressed as

$$\gamma_1 = \frac{P_1||\boldsymbol{h}_{d,1}^H + \boldsymbol{h}_{r,1}^H \Theta \boldsymbol{\mathcal{G}}||^2}{\sigma^2}. \tag{7}$$

Note that this is actually the SNR since UE 1 does not experience interference.

D. Energy Efficiency

Within the considered the system model, energy consumption for offloading transmission E_n^O, energy consumption for local processing E_n^L, and energy consumption for MEC processing E_n^M are the three components in the energy consumption for the n-th UE in a frame.

For the n-th UE, E_n^O is formulated as the product of the offloading transmission power P_n and the offloading time $m_a T_{\text{syb}}$ of the UE:

$$E_n^O = P_n m_a T_{\text{syb}}. \tag{8}$$

Note that since NOMA transmission is adopted, all UEs share the same offloading time of $m_a T_{\text{syb}}$ in the UL phase.

According to [2], the local CPU frequency f_n and the locally processed data size I_n–D_n of the n-th UE determine the energy consumption for local processing, which is expressed as follows:

$$E_n^L = (I_n - D_n) X_n \Gamma_n {f_n}^2, \tag{9}$$

where Γ_n, which varies depending on the processor's architecture, is the n-th UE's effective capacitance coefficient.

Similar to E_n^L, the allocated computational resource (CPU frequency) F_n and the D_n data bits processed at the MEC server determine the MEC processing energy consumption E_n^M for the n-th UE at the BS:

$$E_n^M = D_n X_n \Gamma_M {F_n}^2, \tag{10}$$

where Γ_M is the MEC server's effective capacitance coefficient, which depends on the processor's design. In this paper, we presume $\Gamma_M \ll \Gamma_n, \forall n \in \mathcal{N}$.

Accordingly, the overall energy consumption for the n-th UE in a frame is $E_n = E_n^O + E_n^L + E_n^M$. In this paper, our objective is to maximize the energy efficiency, which is defined as the ratio of total processed data bits over total energy consumption in a frame, i.e.,

$$EE = \frac{\sum_{n=1}^{N} I_n}{\sum_{n=1}^{N} E_n}. \tag{11}$$

E. Notations

The key parameters of the system and their notations are summarized in Table I.

III. Energy Efficiency with a Single UE

To analyze the RIS-aided MEC network in the finite blocklength regime, we first study the scenario in which there

[1]Due to the small sizes of computation results, the time needed for downloading from the BS is typically negligible compared to the time required for offloading and computing.

N	Number of UEs
K	Number of reflecting components at the RIS
B	Number of antennas at the BS
I	Task-input size
T	Completion latency constraint
X	Computation intensity
M	Blocklength constraint of a frame
T_{syb}	Duration of a transmission symbol
m_a	Symbol lengths of the uplink (UL) offloading phase
m_b	Symbol lengths of the computation phase
ε	Decoding error probability in the UL phase
ε_{max}	Maximum decoding error rate constraint
D_n	Data bits offloaded to the MEC server from UE n
$\boldsymbol{h}_n$	Fading vector from UE n to the BS
$\boldsymbol{h}_{d,n}^H$	Fading vector of the direct link from UE n to the BS
$\boldsymbol{h}_{r,n}^H$	Fading vector from UE n to the RIS
$\boldsymbol{\mathcal{G}}$	Fading matrix from the RIS to the BS
Θ	Phase shift matrix of the RIS
γ	Signal-to-noise ratio (SNR)/signal-to-interference-plus-noise ratio (SINR)
P_n	Average power of the transmitted signal of UE n
$\boldsymbol{\eta}$	Additive white Gaussian noise (AWGN)
σ^2	Noise power of the AWGN
Γ_n	Effective capacitance coefficient of UE n
Γ_M	Effective capacitance coefficient of MEC server
F_n	Allocated computational resource (CPU frequency) of UE n at the MEC server
$F_{\max}$	Maximal CPU frequency at the BS
$\mathrm{I}_{\max}$	Maximal number of iterations in alternating optimization algorithm

TABLE I: Summary of symbols and notations.

is only one UE in the system model. We hereby formulate the problem of energy efficiency maximization and propose a solution method for optimizing the RIS reflecting coefficients, offloaded data bits, and offloading duration. The consideration of the single UE case enables us to elucidate the main approach and proposed techniques in a simpler setting. Subsequently extend the analysis to the more challenging and higher-dimensional multiuser cases with NOMA and hybrid NOMA-TDMA in Sections IV and V.

A. Problem Formulation for the Single UE Case

Our goal is to maximize the overall energy efficiency by optimally determining the length of the UL phase as well as the UEs' offloaded data bits and RIS phase shift matrix subject to coding blocklength and maximum decoding error rate constraints. Hence, the overall optimization problem for the single UE case is formulated as follows:

$$\textbf{P1:} \quad \underset{\{D,m_a,\boldsymbol{\theta}\}}{\textbf{Maximize}} \quad \frac{I}{E} \tag{12}$$

$$\textbf{s. t.} \quad \varepsilon \leq \varepsilon_{max}, \tag{12a}$$

$$0 \leq D \leq I, \tag{12b}$$

$$\frac{(I-D)X}{f} \leq MT_{\text{syb}}, \tag{12c}$$

$$\frac{DX}{F} \leq m_b T_{\text{syb}}, \tag{12d}$$

$$m_a, m_b \in \mathbb{Z}, \tag{12e}$$

where ε_{max} is the maximum decoding error rate constraint. Moreover, (12b) is the range of D. (12c) and (12d) are the local computing delay constraint and the MEC computing delay constraint, respectively.

P1 is a non-convex optimization problem due to the non-convex constraints and the strongly coupled optimization variables $D, m_a, \boldsymbol{\theta}$. Hence, finding the globally optimal solution is challenging. To address this, we propose a three-step alternating optimization method to decouple the optimization variables and solve this problem iteratively.

B. Three-step Alternating Optimization

To decouple the optimization variables, in the i-th iteration, we first fix D, m_a as $D_{i-1}, m_{a,i-1}$ (by adopting the optimization results in the $i-1$-th iteration) to design the RIS reflecting coefficients $\boldsymbol{\theta}_i$. Then, with fixed $\boldsymbol{\theta}_i, m_{a,i-1}$, we can optimally obtain D_i. We further optimize $m_{a,i}$ with the fixed $D_i, \boldsymbol{\theta}_i$ and use it in the $i+1$-th iteration.

1) Optimization of the RIS Reflecting Coefficients: In the i-th iteration, it is obvious that $\frac{I}{E}$ is fixed when $D = D_{i-1}$ and $m_a = m_{a,i-1}$, and hence we now seek to find a proper $\boldsymbol{\theta}$ under the maximum error rate constraint. Since the Q function is monotonically decreasing, we know based on (2) that increasing the value of $L = \sqrt{\frac{m_a}{V}}(\log_2(1+\gamma) - \frac{D}{m_a})\log_e 2$ decreases ε. We further observe that L (defined above) is monotonically increasing with the SNR γ. Therefore, when the RIS coefficients are being optimized, **P1** reduces to **P1A**:

$$\textbf{P1A:} \quad \text{Find} \quad \boldsymbol{\theta} \tag{13}$$

$$\textbf{s. t.} \quad |e^{i\theta_k}| = 1, \quad \forall k \in \mathcal{K} \tag{13a}$$

$$\gamma \geq \gamma_{th}, \tag{13b}$$

where γ_{th} is the minimum SNR needed to satisfy the error rate constraint ε_{max}. Even though one feasible solution for **P1A** is sufficient for continuing the algorithm, we can improve the convergence speed by finding the maximum SNR we can obtain by adjusting $\boldsymbol{\theta}$ since larger SNR enables us to offload more data bits with a given uploading blocklength and hence improves the energy efficiency. So instead of solving **P1A**, we equivalently solve **P1B**:

$$\textbf{P1B:} \quad \underset{\boldsymbol{\theta}}{\textbf{Maximize}} \quad \rho||\boldsymbol{h}_d^H + \boldsymbol{h}_r^H \Theta \boldsymbol{\mathcal{G}}||^2 \tag{14}$$

$$\textbf{s. t.} \quad |e^{i\theta_k}| = 1, \quad \forall k \in \mathcal{K}, \tag{14a}$$

where $\rho = \frac{P}{\sigma^2}$. **P1B** is still a non-convex optimization problem in general. According to [16], we define a vector $\boldsymbol{\phi} = [\phi_1, \phi_2, ..., \phi_K]^H$, where $\phi_k = e^{i\theta_k}$. We further define $\boldsymbol{\Phi} = \text{diag}(\boldsymbol{h}_r^H)\boldsymbol{\mathcal{G}} \in \mathbb{C}^{K \times V}$ so that $\boldsymbol{h}_r^H \Theta \boldsymbol{\mathcal{G}} = \boldsymbol{\phi}^H \boldsymbol{\Phi}$, and hence we have $\rho||\boldsymbol{h}_d^H + \boldsymbol{h}_r^H \Theta \boldsymbol{\mathcal{G}}||^2 = \rho||\boldsymbol{h}_d^H + \boldsymbol{\phi}^H \boldsymbol{\Phi}||^2$.

Expanding $||\boldsymbol{h}_d^H + \boldsymbol{\phi}^H \boldsymbol{\Phi}||^2$ we have $\gamma = \rho(||\boldsymbol{h}_d^H||^2 + \boldsymbol{h}_d^H \boldsymbol{\Phi}^H \boldsymbol{\phi} + \boldsymbol{\phi}^H \boldsymbol{\Phi} \boldsymbol{h}_d + \boldsymbol{\phi}^H \boldsymbol{\Phi} \boldsymbol{\Phi}^H \boldsymbol{\phi})$. Similar to [16], we now introduce an auxiliary variable χ and define

$$\boldsymbol{W} = \begin{bmatrix} \boldsymbol{\Phi}\boldsymbol{\Phi}^H & \boldsymbol{\Phi}\boldsymbol{h}_d \\ \boldsymbol{h}_d^H \boldsymbol{\Phi}^H & 0 \end{bmatrix}, \widetilde{\boldsymbol{\phi}} = \begin{bmatrix} \boldsymbol{\phi} \\ \chi \end{bmatrix}.$$

Hence, the SNR γ can be further expressed as $\gamma = \rho(||\boldsymbol{h}_d^H||^2 + \widetilde{\boldsymbol{\phi}}^H \boldsymbol{W} \widetilde{\boldsymbol{\phi}})$. We then construct a positive semidefinite matrix (PSD) $\boldsymbol{\Psi}$ related to the RIS reflecting coefficients and define $\boldsymbol{\Psi} = \widetilde{\boldsymbol{\phi}} \widetilde{\boldsymbol{\phi}}^H$ with the constraints $\boldsymbol{\Psi} \succeq \mathbf{0}$ and $\text{rank}(\boldsymbol{\Psi}) = 1$. With this, we have $\widetilde{\boldsymbol{\phi}}^H \boldsymbol{W} \widetilde{\boldsymbol{\phi}} = \text{Tr}(\boldsymbol{W} \widetilde{\boldsymbol{\phi}} \widetilde{\boldsymbol{\phi}}^H) = \text{Tr}(\boldsymbol{W}\boldsymbol{\Psi})$. Note that $\text{rank}(\boldsymbol{\Psi}) = 1$ is a non-convex constraint, and we relax this constraint and adopt semidefinite relaxation (SDR) method to solve **P1B-1** formulated below:

$$\textbf{P1B-1:} \quad \underset{\boldsymbol{\Psi}}{\textbf{Maximize}} \ \rho(||\boldsymbol{h}_d^H||^2 + \text{Tr}(\boldsymbol{W}\boldsymbol{\Psi})) \tag{15}$$

$$\textbf{s. t.} \quad \boldsymbol{\Psi}_{k,k} = 1, \quad k = 1, 2, ..., K+1, \tag{15a}$$

$$\boldsymbol{\Psi} \succeq \mathbf{0}. \tag{15b}$$

P1B-1 is a convex semidefinite program (SDP) that can be solved optimally by readily available software packages like CVX. Note that solving **P1B-1** will not necessarily give us a rank-one solution. If $\text{rank}(\boldsymbol{\Psi}) = 1$, we can obtain an optimal solution $\widetilde{\boldsymbol{\phi}}^*$ from $\boldsymbol{\Psi} = \widetilde{\boldsymbol{\phi}} \widetilde{\boldsymbol{\phi}}^H$, otherwise the Gaussian randomization can be used to recover a sub-optimal $\widetilde{\boldsymbol{\phi}}^*$, as discussed in [8].

Now we can obtain $\boldsymbol{\theta}^*$ from $\widetilde{\boldsymbol{\phi}}^*$ resulting in the maximum γ as being the solution of **P1B-1**. $\boldsymbol{\theta}^*$ is a solution of **P1A** unless **P1A** is not feasible. We denote the obtained $\boldsymbol{\theta}^*$ in the i-th iteration as $\boldsymbol{\theta}_i$.

2) Optimization of Offloaded Data Bits: By fixing $\boldsymbol{\theta} = \boldsymbol{\theta}_i$ and $m_a = m_{a,i-1}$, we formulate **P1C** below to obtain the optimal number of data bits to be offloaded:

$$\textbf{P1C:} \quad \underset{D}{\textbf{Maximize}} \quad \frac{I}{E} \tag{16}$$

$$\textbf{s. t.} \quad \varepsilon \leq \varepsilon_{max}, \tag{16a}$$

$$0 \leq D \leq I, \tag{16b}$$

$$\frac{(I-D)X}{f} \leq MT_{\text{syb}}, \tag{16c}$$

$$\frac{DX}{F} \leq m_b T_{\text{syb}}. \tag{16d}$$

From (2), we know that increasing D will also increase the ε, and hence (16a) will give us a upper bound on D. (16c) and (16d) will provide a lower bound and an upper bound on D, respectively. Combining all the bounds with (16b), we can obtain a range of D. Based on $\Gamma_M \ll \Gamma_L$ and (9), (10), the total energy efficiency $\frac{I}{E}$ is a monotonic function of D, and therefore the optimal D^* in **P1C** can be easily determined from the feasible range, and we set $D_i = D^*$.

3) Optimization of the Offloading Blocklength: With fixed $\boldsymbol{\theta} = \boldsymbol{\theta}_i$ and $D = D_i$, problem **P1** now becomes **P1D**:

$$\textbf{P1D:} \quad \underset{m_a}{\textbf{Maximize}} \quad \frac{I}{E} \tag{17}$$

$$\textbf{s. t.} \quad \varepsilon \leq \varepsilon_{max}, \tag{17a}$$

$$\frac{DX}{F} \leq m_b T_{\text{syb}}. \tag{17b}$$

$$m_a, m_b \in \mathbb{Z}. \tag{17c}$$

According to (2), decreasing m_a will increase the ε, and therefore (17a) leads to a lower bound on m_a. Moreover, (17b) will give us an upper bound on m_a since $m_a = M - m_b$. With these, we can obtain a feasible range of m_a. Additionally, due to (8), the total energy efficiency $\frac{I}{E}$ is a monotonic function of m_a in this case. Hence the optimal m_a^* in **P1D** can be identified from the feasible range, and we set $m_{a,i} = m_a^*$.

By iteratively solving **P1B-1**, **P1C** and **P1D**, we can obtain the solutions for **P1** once they converge. The proposed three-step algorithm for the single UE case is described in Algorithm 1 below.

Algorithm 1 Three-step alternating optimization for the single UE case

Initialization:

1) Initialize D_0, $m_{a,0}$.

Actions:

1) **For** $i = 1 : \mathrm{I}_{max}$
2) Obtain $\boldsymbol{\theta}_i$ by solving **P1B-1** with D_{i-1}, $m_{a,i-1}$.
3) Obtain D_i by solving **P1C** with $\boldsymbol{\theta}_i$, $m_{a,i-1}$.
4) Obtain $m_{a,i}$ by solving **P1D** with $\boldsymbol{\theta}_i$, D_i.
5) **End** If converge.

C. Convergence Analysis with a Single UE

Since in the i-th iteration in **P1B**, $\gamma_{th,i-1}$ is the threshold SNR with which ε_{max} is attained, we have

$$\varepsilon_{max} \approx Q\left(\sqrt{\frac{m_{a,i-1}}{V_{th,i}}}(\log_2(1+\gamma_{th,i}) - \frac{D_{i-1}}{m_{a,i-1}})\log_e 2\right), \tag{18}$$

where $V_{th,i} = 1 - (1+\gamma_{th,i})^{-2}$.

We further have in the $i-1$-th iteration

$$\varepsilon_{i-1} \approx Q\left(\sqrt{\frac{m_{a,i-1}}{V_{i-1}}}(\log_2(1+\gamma_{i-1}) - \frac{D_{i-1}}{m_{a,i-1}})\log_e 2\right). \tag{19}$$

In the $i-1$-th iteration, by solving **P1D**, we can obtain $m_{a,i-1}$ based on the constraint (17a) which leads to a lower bound on m_a, and $m_{a,i-1}$ will be determined by this lower bound. In other words, $m_{a,i-1}$ is achieved only when the equality in (17a) holds, so we have $\varepsilon_{i-1} = \varepsilon_{max}$. Compared with (18) and (19), this equality leads to $\gamma_{i-1} = \gamma_{th,i}$. In addition, in the i-th iteration (13b) ensures $\gamma_i \geq \gamma_{th,i}$, which results in $\gamma_i \geq \gamma_{i-1}$. This indicates that γ grows with the increase in the iteration index. Note that γ has an upper bound due to ρ being finite. Hence, the proposed Algorithm 1 converges.

D. Stopping Point Analysis with a Single UE

We first present the key characterization as a remark and subsequently provide arguments to establish the result.

Remark 1: *Algorithm 1 will stop in the $j+1$-th iteration when the offloaded data D_j is determined via (16a), i.e., $\varepsilon = \varepsilon_{max}$ in solving problem **P1C** in the j-th iteration.*

This remark establishes that the optimal number of bits to be offloaded is determined when the decoding error constraint is satisfied with equality. When this occurs, the other variables, e.g., RIS coefficient matrix and offloading blocklength, remain the same in the following iteration and are optimized as well. This can be shown as follows. Assume in the j-th iteration that problem **P1C** is solved and D_j is determined by having $\varepsilon = \varepsilon_{max}$. In solving the problem **P1D** in the same iteration, $\boldsymbol{\theta}_j$ and D_j will be used, and hence with the same m_a we still have $\varepsilon = \varepsilon_{max}$ in **P1D**. Furthermore, the optimal m_a should take the minimum value in its feasible range, which is simply decided by (17a), and thereby the same $m_{a,j} = m_{a,j-1}$ will be obtained due to $\varepsilon = \varepsilon_{max}$.

In the $j+1$-th iteration, $\boldsymbol{\theta}_{j+1} = \boldsymbol{\theta}_j$ since nothing changes in solving problem **P1B-1** and we should have the same results. In addition, we should also have $D_{j+1} = D_j$ due to $\boldsymbol{\theta}_{j+1} = \boldsymbol{\theta}_j$ and $m_{a,j} = m_{a,j-1}$, which are the same fixed values as in the j-th iteration. Moreover, $m_{a,j+1} = m_{a,j}$ can be obtained due to the same reason as in the j-th iteration.

Finally, according to $\boldsymbol{\theta}_{j+1} = \boldsymbol{\theta}_j$, $D_{j+1} = D_j$ and $m_{a,j+1} = m_{a,j}$ we can claim that Algorithm 1 will stop at the j-th iteration when D_j is determined via (16a), i.e., $\varepsilon = \varepsilon_{max}$ in solving problem **P1C** in the j-th iteration. □

IV. Energy Efficiency with Multiple UEs and NOMA

In this section, we first formulate and analyze the global optimization problem with N UEs (that utilize NOMA for their transmissions) and then propose a three-step alternating optimization method to solve the problem. Note that, compared to the single UE case, we now have a more challenging and higher-dimensional optimization problem. In particular, as major differences from the previous case, we need to optimize the offloaded data of multiple UEs, address interference and consider SINR rather than SNR, and determine the optimal computational resource allocation at the MEC.

A. Problem Formulation

By jointly determining the UEs' offloaded data bits $\{D_n\}$, the length of the UL NOMA phase m_a, CPU frequencies allocated to the tasks of different UEs $\{F_n\}$, and the RIS reflecting coefficients $\boldsymbol{\theta}$ subject to both coding blocklength and maximum decoding error rate constraints, we aim to maximize the overall energy efficiency. Consequently, the global energy efficiency maximization problem with N UEs is formulated as follows:

$$\textbf{P2:} \quad \underset{\{D_n, m_a, F_n, \boldsymbol{\theta}\}}{\textbf{Maximize}} \ \frac{\sum_{n=1}^{N} I_n}{\sum_{n=1}^{N} E_n} \tag{20}$$

$$\textbf{s. t.} \quad \varepsilon_n \leq \varepsilon_{max,n}, \quad \forall n \in \mathcal{N} \tag{20a}$$

$$0 \leq D_n \leq I_n, \quad \forall n \in \mathcal{N} \tag{20b}$$

$$\frac{(I_n - D_n)X_n}{f_n} \leq MT_{\text{syb}}, \quad \forall n \in \mathcal{N} \tag{20c}$$

$$\frac{D_n X_n}{F_n} \leq m_b T_{\text{syb}}, \quad \forall n \in \mathcal{N} \tag{20d}$$

$$\sum_{n=1}^{N} F_n \leq F_{\max}, \tag{20e}$$

$$m_a, m_b \in \mathbb{Z}, \tag{20f}$$

where $F_{\max}$ is the maximal CPU frequency at the BS, $\varepsilon_{max,n}$ is the maximum decoding error rate constraint for the n-th UE. Moreover, (20b) is the range of offloaded data bits. (20c) and (20d) are the delay constraints in local computing and the MEC processing, respectively.

Note that the BS will decode the signal from the last UE N first. If $\hat{\varepsilon}_N$ is our desired decoding error rate for the N-th UE, then we let $\varepsilon_{max,N} = \hat{\varepsilon}_N$ in (20a). However, the signal from the n-th UE can be successfully decoded only when the signals from all the previous $N-n$ UEs are decoded without error, otherwise the interference cannot be canceled via the SIC technique. Based on iterative analysis, if the signal from the $n+1$-th UE is decoded successfully, then all the signals from the previous $N-n-1$ UEs are decoded perfectly. In that case, the overall error rate of the n-th UE is given by $\varepsilon_n + \varepsilon_{n+1} - \varepsilon_n\varepsilon_{n+1}$, where ε_n and ε_{n+1} are the error rates of the n-th UE and the $n+1$-th UE (in the offloading phase) that can be calculated by using (2). Note that $\varepsilon_n\varepsilon_{n+1}$ is neglected in this paper due to both ε_n and ε_{n+1} having typically small values in a URLLC setting. If $\hat{\varepsilon}_n$ is the desired decoding error rate for UE n, we should have $\varepsilon_n + \varepsilon_{n+1} \leq \hat{\varepsilon}_n$. Considering that we have $\varepsilon_{n+1} \leq \hat{\varepsilon}_{n+1}$, if $\varepsilon_n \leq \hat{\varepsilon}_n - \hat{\varepsilon}_{n+1}$ is satisfied, then the n-th UE's error rate requirement will also be satisfied by setting $\varepsilon_{max,n} = \hat{\varepsilon}_n - \hat{\varepsilon}_{n+1}$ in (20a).

As a result of the non-convex constraints and the strongly coupled optimization variables $\{D_n\}, m_a, \{F_n\}, \boldsymbol{\theta}$, **P2** is a non-convex optimization problem, and obtaining the globally optimal solution is challenging. In order to address this, we again propose a three-step alternating optimization method that decouples the optimization variables and iteratively solves the problem.

B. Three-step Alternating Optimization for Multiple UEs

In the i-th iteration, we first fix D_n, F_n, m_a as $D_{n,i-1}, F_{n,i-1}, m_{a,i-1}$ by adopting the optimization results in the $i-1$-th iteration and design the RIS reflecting coefficients $\boldsymbol{\theta}_i$ in order to decouple the optimization variables. Then, with given $\boldsymbol{\theta}_i, F_{n,i-1}, m_{a,i-1}$, we can optimally obtain $D_{n,i}$ in the second step. Furthermore, we optimize $m_{a,i}$ and $F_{n,i}$ with the fixed $D_{n,i}, \boldsymbol{\theta}_i$ in the third step and use them in the $i+1$-th iteration.

1) Optimization of RIS Reflecting Coefficient: In the i-th iteration, when $\{D_n\}, \{F_n\}, m_a$ are fixed it is obvious that $\frac{\sum_{n=1}^{N} I_n}{\sum_{n=1}^{N} E_n}$ is also fixed, and hence we now seek to find a proper $\boldsymbol{\theta}$ to satisfy the maximum error rate constraints. Similar to the single UE case, we increase the SINR γ to decrease the decoding error rate ϵ, and hence **P2** is transformed into **P2A** when $\{D_n\}, \{F_n\}, m_a$ are fixed:

$$\textbf{P2A:} \quad \text{Find} \quad \boldsymbol{\theta} \tag{21}$$

$$\textbf{s. t.} \quad |e^{i\theta_k}| = 1, \quad \forall k \in \mathcal{K} \tag{21a}$$

$$\gamma_n \geq \gamma_{th,n}, \quad \forall n \in \mathcal{N} \tag{21b}$$

where $\gamma_{th,n}$ is the minimum SINR required to comply with the n-th UE's error rate constraint $\varepsilon_{max,n}$. Even though one viable solution of $\boldsymbol{\theta}$ suffices to continue the iterations, we can speed up convergence by maximizing the SINR of UE 1, as that signal will be the last to be decoded. As a result, we solve **P2B** instead of solving **P2A**:

$$\textbf{P2B:} \quad \underset{\boldsymbol{\theta}}{\textbf{Maximize}} \quad \rho_1 ||\boldsymbol{h}_{d,1}^H + \boldsymbol{h}_{r,1}^H \Theta \boldsymbol{\mathcal{G}}||^2 \tag{22}$$

$$\textbf{s. t.} \quad |e^{i\theta_k}| = 1, \quad \forall k \in \mathcal{K} \tag{22a}$$

$$\gamma_n \geq \gamma_{th,n}, \quad \forall n \in \{2, ..., N\}. \tag{22b}$$

In general, **P2B** is still a non-convex optimization problem. Following the same analysis of **P1B** in Section III(B), we can express the SINR of UE 1 as $\gamma_1 = \frac{P_1(||\boldsymbol{h}_{d,1}^H||^2 + \text{Tr}(\boldsymbol{W}_1\boldsymbol{\Psi}))}{\sigma^2}$, and the SINR of the n-th UE can thus be expressed as

$$\gamma_n = \frac{P_n(||\boldsymbol{h}_{d,n}^H||^2 + \text{Tr}(\boldsymbol{W}_n\boldsymbol{\Psi}))}{\sum_{t=1}^{n-1} P_t(||\boldsymbol{h}_{d,t}^H||^2 + \text{Tr}(\boldsymbol{W}_t\boldsymbol{\Psi})) + \sigma^2}. \tag{23}$$

In **P2B**, we now know that (22b) requires us to satisfy $\gamma_n = \frac{P_n(||\boldsymbol{h}_{d,n}^H||^2 + \text{Tr}(\boldsymbol{W}_n\boldsymbol{\Psi}))}{\sum_{t=1}^{n-1} P_t(||\boldsymbol{h}_{d,t}^H||^2 + \text{Tr}(\boldsymbol{W}_t\boldsymbol{\Psi})) + \sigma^2} \geq \gamma_{th,n}, \forall n \in \{2, ..., N\}$, which is equivalent to satisfying the following inequalities:

$$P_n \text{Tr}(\boldsymbol{W}_n\boldsymbol{\Psi}) - \gamma_{th,n} \sum_{t=1}^{n-1} P_t \text{Tr}(\boldsymbol{W}_t\boldsymbol{\Psi}) \geq \gamma_{th,n} \sum_{t=1}^{n-1} P_t ||\boldsymbol{h}_{d,t}^H||^2 + \gamma_{th,n}\sigma^2 - P_n ||\boldsymbol{h}_{d,n}^H||^2, \quad \forall n \in \{2, ..., N\} \tag{24}$$

Therefore, instead of solving **P2B**, we equivalently solve **P2B-1**:

$$\textbf{P2B-1:} \quad \underset{\boldsymbol{\Psi}}{\textbf{Maximize}} \ \frac{P_1(||\boldsymbol{h}_{d,1}^H||^2 + \text{Tr}(\boldsymbol{W_1}\boldsymbol{\Psi}))}{\sigma^2} \tag{25}$$

$$\textbf{s. t.} \quad \boldsymbol{\Psi}_{k,k} = 1, \quad k = 1, 2, ..., K+1, \tag{25a}$$

$$P_n \text{Tr}(\boldsymbol{W}_n\boldsymbol{\Psi}) - \gamma_{th,n} \sum_{t=1}^{n-1} P_t \text{Tr}(\boldsymbol{W}_t\boldsymbol{\Psi}) \geq \gamma_{th,n} \sum_{t=1}^{n-1} P_t ||\boldsymbol{h}_{d,t}^H||^2 + \gamma_{th,n}\sigma^2 - P_n ||\boldsymbol{h}_{d,n}^H||^2, \quad \forall n \in \{2, ..., N\}, \tag{25b}$$

$$\boldsymbol{\Psi} \succeq \mathbf{0}. \tag{25c}$$

When solving **P2B-1**, we relax the non-convex constraint $\text{rank}(\boldsymbol{\Psi}) = 1$ and adopt the semidefinite relaxation (SDR) method to address it. Therefore, **P2B-1** becomes a convex semidefinite program (SDP) that can be solved optimally by using a conventional convex optimization tool. As also discussed before, solving **P2B-1** will not necessarily give us a rank-one solution. The optimal solution $\widetilde{\boldsymbol{\phi}}^*$ can be obtained from the equation $\boldsymbol{\Psi} = \widetilde{\boldsymbol{\phi}}\widetilde{\boldsymbol{\phi}}^H$ if $\text{rank}(\boldsymbol{\Psi}) = 1$. Otherwise,

the Gaussian randomization [8] can be utilized to retrieve a sub-optimal $\widetilde{\phi}^*$. Then, we may derive $\boldsymbol{\theta}^*$ from $\widetilde{\phi}^*$ providing the maximum γ_1 as being the solution of **P2B-1**. Hence, this $\boldsymbol{\theta}^*$ is a viable solution of **P2A**. Such obtained $\boldsymbol{\theta}^*$ in the i-th iteration is denoted as $\boldsymbol{\theta}_i$.

2) Optimization of Offloaded Data: In the second step of the three-step optimization algorithm, we fix $\boldsymbol{\theta} = \boldsymbol{\theta}_i$, $m_a = m_{a,i-1}$ and $F_n = F_{n,i-1}$, and then **P2C** is formulated to obtain the optimal number of data bits to be offloaded:

$$\textbf{P2C:} \quad \underset{\{D_n\}}{\textbf{Maximize}} \quad \frac{\sum_{n=1}^{N} I_n}{\sum_{n=1}^{N} E_n} \tag{26}$$

$$\textbf{s. t.} \quad \varepsilon_n \leq \varepsilon_{max,n}, \quad \forall n \in \mathcal{N} \tag{26a}$$

$$0 \leq D_n \leq I_n, \quad \forall n \in \mathcal{N} \tag{26b}$$

$$\frac{(I_n - D_n)X_n}{f_n} \leq MT_{\text{syb}}, \quad \forall n \in \mathcal{N} \tag{26c}$$

$$\frac{D_n X_n}{F_n} \leq m_b T_{\text{syb}}, \quad \forall n \in \mathcal{N} \tag{26d}$$

Based on (2), it is obvious that increasing D_n will result in a larger ε_n, so each inequality in (26a) will give us an upper bound on D_n for $n = 1, \ldots, N$. Besides, (26c) and (26d) will provide a lower bound and an upper bound on D_n, respectively. Considering (26b), all the bounds can be combined to create a range of feasible D_n. Since we have $\Gamma_M \ll \Gamma_n, \forall n \in \mathcal{N}$ and (9), (10), the total energy efficiency $\frac{\sum_{n=1}^{N} I_n}{\sum_{n=1}^{N} E_n}$ is a monotonic function of D_n, and thus the optimal D_n^* in **P2C** can be directly determined from the feasible range, and we set $D_{n,i} = D_n^*$ for $n = 1, \ldots, N$ in the i-th iteration.

3) Optimization of Offloading Blocklength and CPU Frequency Allocation: In the third step of the proposed algorithm, we first fix $\boldsymbol{\theta} = \boldsymbol{\theta}_i$, $D_n = D_{n,i}$, and **P2** becomes **P2D**:

$$\textbf{P2D:} \quad \underset{m_a, \{F_n\}}{\textbf{Maximize}} \quad \frac{\sum_{n=1}^{N} I_n}{\sum_{n=1}^{N} E_n} \tag{27}$$

$$\textbf{s. t.} \quad \varepsilon_n \leq \varepsilon_{max,n}, \quad \forall n \in \mathcal{N} \tag{27a}$$

$$\frac{D_n X_n}{F_n} \leq m_b T_{\text{syb}}, \quad \forall n \in \mathcal{N} \tag{27b}$$

$$\sum_{n=1}^{N} F_n \leq F_{\max}, \tag{27c}$$

$$m_a, m_b \in \mathbb{Z}. \tag{27d}$$

From (2), we know that decreasing m_a will increase ε_n, and hence (27a) will give us a lower bound on m_a. Moreover, based on (8), it is obvious that decreasing m_a leads to a higher value for $\frac{\sum_{n=1}^{N} I_n}{\sum_{n=1}^{N} E_n}$. In addition, (27b) will give us an upper bound on m_a, and therefore ${m_a}^*$ can be obtained simply based on (27a). According to (10), when m_a^* is determined, we can transform the CPU frequency allocation into following problem **P2E**:

$$\textbf{P2E:} \quad \underset{\{F_n\}}{\textbf{Minimize}} \quad \sum_{n=1}^{N} E_n \tag{28}$$

$$\textbf{s. t.} \quad \frac{D_n X_n}{F_n} \leq m_b T_{\text{syb}}, \quad \forall n \in \mathcal{N} \tag{28a}$$

$$\sum_{n=1}^{N} F_n \leq F_{\max}. \tag{28b}$$

P2E is a convex problem and the optimal ${F_n}^*$ in **P2E** can be readily obtained. In the i-th iteration, we set $m_{a,i} = m_a^*$, $F_{n,i} = F_n^*$.

By iteratively solving **P2B-1**, **P2C** and **P2D** as well as **P2E**, we can obtain the solution of **P2** once they converge. Algorithm 2 below provides a description of the proposed three-step optimization algorithm.

Algorithm 2 Three-step alternating optimization for multiple UEs

Initialization:

1) Initialize $\{D_{n,0}\}$, $m_{a,0}$, $\{F_{n,0}\}$.

Actions:

1) **For** $i = 1 : \mathrm{I}_{\max}$
2) Obtain $\boldsymbol{\theta}_i$ by solving **P2B-1** with $\{D_{n,i-1}\}$, $m_{a,i-1}$, $\{F_{n,i-1}\}$.
3) Obtain $\{D_{n,i}\}$ by solving **P2C** with $\boldsymbol{\theta}_i$, $m_{a,i-1}$, $\{F_{n,i-1}\}$.
4) First obtain $m_{a,i}$ from (27a) in **P2D** with $\boldsymbol{\theta}_i$, $\{D_{n,i}\}$ and then obtain $\{F_{n,i}\}$ by solving **P2E** with $\boldsymbol{\theta}_i$, $m_{a,i}$, $\{D_{n,i}\}$.
5) **End** If converge.

C. Convergence Analysis with Multiple UEs and NOMA

We first have the following proposition:

Proposition 1: *γ_1 is non-decreasing as the iteration index i in the proposed three-step optimization in Algorithm 2 increases.*

Proof: Let $\gamma_1(\boldsymbol{\theta}_{i+1}|\boldsymbol{D}_i, m_{a,i}, \boldsymbol{F}_i)$ to be the SNR of UE 1 after solving **P2B-1** in the $i+1$-th iteration with given parameters $\boldsymbol{D}_i, m_{a,i}, \boldsymbol{F}_i$, and hence we have

$$\begin{aligned}\gamma_1(\boldsymbol{\theta}_{i+1}|\boldsymbol{D}_i, m_{a,i}, \boldsymbol{F}_i) &\overset{(a)}{\geq} \gamma_1(\boldsymbol{\theta}_i|\boldsymbol{D}_i, m_{a,i}, \boldsymbol{F}_i) \\ &\overset{(b)}{=} \gamma_1(\boldsymbol{\theta}_i|\boldsymbol{D}_{i-1}, m_{a,i-1}, \boldsymbol{F}_{i-1}).\end{aligned} \tag{29}$$

Inequality in (a) above is due to the fact that **P2B** is a maximization problem, and hence in the $i+1$-th iteration the optimal solution $\boldsymbol{\theta}_{i+1}$ should provide us a higher SNR for UE 1. Equality in (b) holds due to γ_1 being only determined by $\boldsymbol{\theta}$. Therefore, γ_1 is non-decreasing. □

Based on the **Proposition 1**, since we have finite transmit power P_1, γ_1 should converge. Note that

$$\frac{\partial \gamma_1}{\partial \boldsymbol{\Psi}} = \frac{\partial P_1(||\boldsymbol{h}_{d,1}^H||^2 + \text{Tr}(\boldsymbol{\Psi}\boldsymbol{W}_1))}{\sigma^2 \partial \boldsymbol{\Psi}} = \frac{P_1 \boldsymbol{W}_1^T}{\sigma^2} \neq \mathbf{0}. \tag{30}$$

When γ_1 converges, $\boldsymbol{\Psi}$ should converge simultaneously. Note that since the derivative above is nonzero, any variation in $\boldsymbol{\Psi}$ will lead to a variation in γ_1, contradicting its convergence. Based on (23), we know γ_n is only decided by $\boldsymbol{\Psi}$. Consequently, $\{\gamma_n\}$ should also converge.

To illustrate the convergence of $\{D_n\}$, we have the following proposition:

Proposition 2: *D_n is non-increasing as the iteration index i in Algorithm 2 increases*[2].

Proof: **P2E** is a convex problem when $\boldsymbol{\theta}$, $\{D_n\}$ and m_a are fixed. According to (8), (9), (10) and (11), we know that smaller $\{F_n\}$ leads to a higher energy efficiency. Considering the constraint in (28a), we can claim that

$$F_n^* = \frac{D_{n,i} X_n}{m_{b,i} T_{\text{syb}}} = F_{n,i}, \tag{31}$$

[2]In order to achieve the optimal solutions, we set $D_n = I_n$ for all UEs in the initialization of Algorithm 2.

for all UEs in the i-th iteration. Hence, in the $i+1$-th iteration, based on (26d) in **P2C**, we have

$$D_{n,i+1} \leq \frac{m_{b,i} T_{\text{syb}} F_{n,i}}{X_n} = D_{n,i}. \tag{32}$$

Therefore, D_n is non-increasing for all UEs. □

We further have $D_n \geq I_n - \frac{M T_{\text{syb}} f_n}{X_n}$ from (26c) in **P2C**. Since the right side of the inequality is a constant, combining with **Proposition 2**, $\{D_n\}$ will converge.

With converged $\{\gamma_n\}$ and $\{D_n\}$, m_a will converge since m_a is determined by (27a) in **P2D**, which is independent of $\{F_n\}$.

Finally, with converged $\{\gamma_n\}$, $\{D_n\}$ and m_a, $\{F_n\}$ should also converge since **P2E** is a convex problem.

Therefore, Algorithm 2 will converge for all UEs.

D. Stopping Point Analysis with Multiple UEs

In order to identify the optimal operating points, We again first present the characterization as a remark and then provide arguments that lead to this characterization.

Remark 2: *Algorithm 2 will stop in the $j+1$-th iteration when there is at least one UE (and assume, without loss of generality, that it is the p-th UE) whose offloaded data $D_{p,j}$ is determined via (26a), i.e., $\varepsilon_{p,j} = \varepsilon_{max,p}$ in solving problem **P2C** in the j-th iteration.*

Assume that the condition introduced in Remark 2 holds, i.e., in the j-th iteration, the p-th UE's offloaded data $D_{p,j}$ is determined by satisfying $\varepsilon_{p,j} = \varepsilon_{max,p}$ in solving problem **P2C**. In the same iteration, $\boldsymbol{\theta}_j$ and $D_{p,j}$ will be used in solving problem **P2D**, and hence with the same m_a we still have $\varepsilon_{p,j} = \varepsilon_{max,p}$ in **P2D**. Therefore, the optimal m_a in **P2D** in the j-th iteration is simply decided by $\varepsilon_{p,j} = \varepsilon_{max,p}$. This is because any decrease in m_a will lead to a violation in $\varepsilon_{p,j} = \varepsilon_{max,p}$, and smaller m_a results in a better energy efficiency. Consequently, the same $m_{a,j} = m_{a,j-1}$ will be obtained since this is the minimum value that m_a can take to satisfy $\varepsilon_{p,j} = \varepsilon_{max,p}$, which is one of the constraints in (27a).

Additionally, in solving problem **P2E** in the j-th iteration, in the results we should have $\frac{D_{n,j} X_n}{F_{n,j}} = m_{b,j} T_{\text{syb}}$ due to $\Gamma_M \ll \Gamma_n, \forall n \in \mathcal{N}$ which indicates that all $\{F_n\}$ should be as small as possible and (28a) provides the lower bounds of $\{F_n\}$. Therefore, for the p-th UE in solving problem **P2E** in the j-th iteration, we have $\frac{D_{p,j} X_p}{F_{p,j}} = m_{b,j} T_{\text{syb}}$.

For any other UE $n \in \mathcal{N} \setminus p$, in the j-th iteration, $D_{n,j}$ should be obtained from (26d), and thereby we have $\frac{D_{n,j} X_n}{F_{n,j-1}} = m_{b,j-1} T_{\text{syb}}, \forall n \in \mathcal{N} \setminus p$. Based on our previous analysis and $m_{a,j} = m_{a,j-1}$, we also have $\frac{D_{n,j} X_n}{F_{n,j}} = m_{b,j} T_{\text{syb}}$, and thus $F_{n,j} = F_{n,j-1}, \forall n \in \mathcal{N} \setminus p$.

In the $j+1$-th iteration, in solving **P2C**, based on the definition of $\gamma_{th,p,j+1}$ we have $\varepsilon_{max,p} = Q\left(\sqrt{\frac{m_{a,j}}{V_{th,p,j+1}}}(\log_2(1+\gamma_{th,p,j+1}) - \frac{D_{p,j}}{m_{a,j}})\log_e 2\right) = \varepsilon_{p,j} = Q\left(\sqrt{\frac{m_{a,j}}{V_{p,j}}}(\log_2(1+\gamma_{p,j}) - \frac{D_{p,j}}{m_{a,j}})\log_e 2\right)$[3], which indicates $\gamma_{p,j} = \gamma_{th,p,j+1}$, resulting in $\gamma_{p,j} \leq \gamma_{p,j+1}$ since $\gamma_{th,p,j+1} \leq \gamma_{p,j+1}$ (see in (27b)). We also have $\varepsilon_{max,p} = Q\left(\sqrt{\frac{m_{a,j}}{V_{p,j+1}}}(\log_2(1+\gamma_{p,j+1}) - \frac{\hat{D}_{p,j+1}}{m_{a,j}})\log_e 2\right) = \varepsilon_{p,j} = Q\left(\sqrt{\frac{m_{a,j}}{V_{p,j}}}(\log_2(1+\gamma_{p,j}) - \frac{D_{p,j}}{m_{a,j}})\log_e 2\right)$, where $\hat{D}_{p,j+1}$ denotes the upper bound of D_p provided by (26a) in the $j+1$-th iteration, and due to $\gamma_{p,j} \leq \gamma_{p,j+1}$ we have $D_{p,j} \leq \hat{D}_{p,j+1}$, combining with $\frac{D_{p,j} X_p}{F_{p,j}} = m_{b,j} T_{\text{syb}}$, $D_{p,j+1}$ is then determined by (26d) in the $j+1$-th iteration, and we have $D_{p,j+1} = D_{p,j}$.

Furthermore, for any other UE $n \in \mathcal{N} \setminus p$, when $\hat{D}_{n,j+1}$ denotes the upper bound of $\{D_n\}$ provided by (26a) in the $j+1$-th iteration, we have $\varepsilon_{max,n} = Q\left(\sqrt{\frac{m_{a,j}}{V_{n,j+1}}}(\log_2(1+\gamma_{n,j+1}) - \frac{\hat{D}_{n,j+1}}{m_{a,j}})\log_e 2\right) = Q\left(\sqrt{\frac{m_{a,j}}{V_{th,n,j+1}}}(\log_2(1+\gamma_{th,n,j+1}) - \frac{D_{n,j}}{m_{a,j}})\log_e 2\right)$, and thereby $D_{n,j} \leq \hat{D}_{n,j+1}$ according to $\gamma_{th,n,j+1} \leq \gamma_{n,j+1}$ (see in (27b)). Consequently, for any UE $n, n \in \mathcal{N} \setminus p$, its $D_{n,j+1}$ is still determined by (26d) in the $j+1$-th iteration. Therefore, $D_{n,j+1} = D_{n,j}, \forall n \in \mathcal{N} \setminus p$.

Moreover, since $\{F_{n,j+1}\}$ are obtained simply from (28a), $\{F_{n,j+1}\}$ should maintain the same values as in the j-th iteration, e.g., $F_{n,j+1} = F_{n,j}$.

Finally, when $\{F_{n,j+1}\}$, $m_{a,j+1}$ and $\{D_{n,j+1}\}$ all keep the same values as in the j-th iteration, $\boldsymbol{\theta}_{j+2} = \boldsymbol{\theta}_{j+1}$ is assured since nothing changes in solving problem **P2B-1** in the $j+2$-th iteration, and we should attain the same results, e.g., $\boldsymbol{\theta}_{j+2} = \boldsymbol{\theta}_{j+1}$. At last, we can claim that Algorithm 2 will stop at the $j+1$-th iteration when there is at least one UE (without loss of generality, p-th UE) whose offloaded data $D_{p,j}$ is determined via (26a), i.e., $\varepsilon_{p,j} = \varepsilon_{max,p}$ in solving problem **P2C** in the j-th iteration. □

V. Energy Efficiency with Hybrid NOMA-TDMA

We note that the algorithm runtime can become a bottleneck in the proposed Algorithm 2 due to the iterative optimization structure. Especially, as the number of UEs increases, the number of constraints in the SDP also grows, as expressed in (25b), and hence increasing the execution time of the entire optimization algorithm.

One approach to reduce the complexity is to execute the optimization algorithm for each UE sequentially. In this case, each UE offloads its own data via OMA and the offloading phase duration $m_a T_{\text{syb}}$ is allocated equally to all UEs. Even though such sequential offloading structure is capable of addressing scenarios with large number of UEs, we note that with the increase in the number of UEs, the offloading time allocated to each UE becomes smaller. In that case, it is difficult for UEs to offload their data on time, which correspondingly deteriorates the energy efficiency.

Hence, in order to strike a balance between complexity reduction and performance improvement, we consider hybrid NOMA-TDMA scheme in this section by dividing the UEs into groups.

A. UE Grouping and Hybrid NOMA-TDMA Transmissions

We consider that UEs are grouped such that the task data bits of UEs in the same group are offloaded simultaneously via NOMA transmission, and OMA transmission (i.e., TDMA) is adopted among different groups. We first propose the UE grouping method, and then we develop a dynamic CPU frequency allocation algorithm at the BS to better take advantage of the UE grouping method and hybrid NOMA-TDMA transmissions. In the following, we initially describe the global energy efficiency optimization problem when UE grouping is adopted. Subsequently, a four-step algorithm is introduced to solve the problem. As also noted above, when UE grouping is

[3]Here we use $m_{a,j} = m_{a,j-1}$.

adopted, we utilize TDMA in the UL phase among different groups. And if we divide the N UEs into G groups, the $\frac{N}{G}$ UEs within each group perform NOMA transmission.

B. Utility Metric for UE Grouping

The criterion on how to group the UEs is of vital importance. Traditional UE grouping methods in NOMA are typically based on the channel vectors [24]. However, such a grouping method does not take the latency requirements/constraints into account. In this paper, we construct a utility metric for UE grouping that balances the weight of channel vectors and the latency constraints.

We first construct a latency-related parameter for UE n: $L_n = \frac{I_n X_n}{f_n}$ that represents the required time if the entire task/service is processed locally. Larger L_n indicates a higher urgency for UE n to offload its data to the MEC server in the UL phase. One important fact that should be noticed is that the transmission order of groups significantly affects the performance. The UEs in the first group will complete their offloading transmissions ahead of other UEs and hence can utilize all the CPU computational resources at the BS (MEC server) until the UEs in the second group complete their transmissions (if dynamic CPU frequency allocation at the BS is adopted). Therefore, it is intuitively better to place the UE with higher L_n in the front groups.

However, channel vector $\boldsymbol{h}$ is still an important factor in grouping especially in NOMA transmissions. From (6), we note that UEs with larger difference between their channel gains can reduce the transmission power requirement to achieve the desired SINR and hence improve the energy efficiency. Now, taking into account both the channel strengths and latency factors, we construct the following utility metric for UE grouping to indicate the importance/urgency of UE n in the g-th group:

$$S_{g,n} = \alpha \frac{Z_g - Z_n}{Z_g} + (1-\alpha)\frac{L_n - L_0}{L_n}, \tag{33}$$

where we define $Z_g = ||\boldsymbol{h}_{d,g}^H + \boldsymbol{h}_{r,g}^H \mathbf{I}\boldsymbol{\mathcal{G}}||$ as the largest channel gain in the g-th group and $Z_n = ||\boldsymbol{h}_{d,n}^H + \boldsymbol{h}_{r,n}^H \mathbf{I}\boldsymbol{\mathcal{G}}||$. L_0 is a constant satisfying $L_0 \leq L_n, \forall n \in \mathcal{N}$. Note that the above metric is used if there is at least one other UE in the group. If there is no UE in the g-the group yet, the UE with the largest gain among the remaining UEs is selected as the first UE. Accordingly, we can describe the UE grouping method based on UE grouping utility in Algorithm 3 below.

Algorithm 3 UE Grouping

Initialization:

1) Calculate L_n, Z_n for UE n, $n \in \mathcal{N}$.

Actions:

1) **For** $g = 1 : G$
2) For all the UEs in the UE set $\mathcal{N}$, place UE p having the largest Z_p (among all remaining unplaced UEs) into group g, set $Z_g = Z_p$.
3) Remove UE p from the UE set $\mathcal{N}$.
4) **While** the g-th group is not full
5) Calculate $S_{g,n}$ for all the UEs in the UE set $\mathcal{N}$.
6) Place UE q having the largest $S_{g,q}$ into group g.
7) Set $Z_g = Z_q$.
8) Remove UE q from the UE set $\mathcal{N}$.
9) **end while**
10) **end for**

According to Algorithm 3, we can divide N UEs into G groups where each group includes $\frac{N}{G}$ UEs and each group is allocated $\tau = \frac{m_a T_{\text{syb}}}{G}$ seconds in the offloading phase.

C. Dynamic CPU Frequency Allocation at the BS

In [22], CPU frequencies are allocated once all transmissions are completed and MEC server has all the data task bits. However, different from [22], we develop a dynamic CPU frequency (computational resource) allocation to better take advantage of UE grouping and scheduling. For instance, as also noted before, the UEs in the first group will complete their offloading transmissions ahead of other UEs and hence it is possible for them to utilize all the CPU computational resources at the BS until the UEs in the second group finish their transmissions.

Before we introduce the dynamic CPU frequency allocation algorithm, we first introduce two lemmas.

Lemma 1: *To minimize the MEC processing energy consumption, for each UE, the optimal approach is to utilize all the available processing time at the MEC server while satisfying the task deadline.*

Proof: From (10), we see that the MEC processing energy consumption is proportional to the square of the allocated frequency. Due to the fact that the result of the MEC processing time multiplied with the allocated frequency should be equal or greater than the data bits offloaded from each UE times the required computational intensity, in the optimal situation the UE should take advantage of all the possible processing time at the MEC server since the square of the frequency grows much faster, leading to higher energy consumption. □

Lemma 2: *In the optimal scenario, for each UE, the allocated frequency within each time slot of duration τ should be equally distributed among all the possible processing time slots while ensuring the timely completion of the processing of all offloaded task data bits.*

Proof: From the grouping method, we know the available number of time slots that can be utilized for each UE. Note also that the slot length is a constant $\tau = \frac{m_a T_{\text{syb}}}{G}$. Once the allocated data bits D_n is fixed, the only parameter that will influence the MEC processing energy consumption is the square of the allocated frequency among all the possible processing time slots of each UE. In addition, by timely completing the processing of all offloaded task data bits, we have a sum constraint for the allocated frequencies across all the possible processing time slots for each UE. Under a sum constraint, for each UE, the frequency should be equally allocated among all the possible processing time slots in order to minimize the MEC processing energy consumption. □

Based on **Lemma 1** and **Lemma 2**, we propose Algorithm 4 below for the dynamic CPU frequency allocation. Note that a UE group is now dynamically allocated CPU frequencies at the MEC/BS once its transmission is completed (instead of waiting for the transmissions of all groups to be completed). This provides more flexibility in resource allocation, leading to improved energy efficiency. In the algorithm below, we first initialize the frequency allocation for all groups, depending on their transmission completion times in the offloading phase. Subsequently, we check whether the maximum CPU frequency limit $F_{\max}$ is exceeded in any interval with such initialization. If exceeded, we determine how to optimally reallocate the CPU frequencies to minimize the MEC energy consumption.

Algorithm 4 Dynamic CPU Frequency Allocation

Initialization:

1) Reorder all the UEs according to the UE grouping results. The first and second UE in the g-th group should be the $\frac{U(g-1)}{G}+1$-th and $\frac{U(g-1)}{G}+2$-th UE.
2) **For** $n=1:N$
   for $g=1:G+1$
   if $g \leq G_n$
   $F_{n,g}=0$
   else calculate $F_{n,g}=\frac{D_n X_n}{m_b T_{\text{syb}}+(G-G_n)\tau}$.
   end if
   end for
3) **end for**

Actions:

1) **For** $g=1:G$
2) **If** $\sum_{n=1}^{N} F_{n,G+2-g} \leq F_{\max}$
   $g=g+1$. **else** calculate $\Delta=\sum_{n=1}^{N} F_{n,G+2-g}-F_{\max}$
   Solve: **P0**: $_{N-\frac{Ng}{G}}$

$$\underset{\mathbf{Y}}{\textbf{Minimize}} \quad \sum_{n=1}^{} \{D_n X_n (F_{n,G+2-g}-Y_n)^2+ \sum_{v=G_n+1}^{G+1-g} D_n X_n (F_{n,v}+\frac{\mathbb{I}_{g=1}(g) m_b T_{\text{syb}} Y_n}{(G-G_n)\tau}+\frac{\mathbb{I}_{g\geq 2}(g) Y_n}{G+1-g-G_n})^2\} \tag{34}$$

$$\textbf{Subject to} \quad \sum_{j=1}^{N-\frac{Ng}{G}} Y_n \geq \Delta,$$

   where $\mathbb{I}(\bullet)$ is the indicator function.
   for $n=1:N-\frac{Ng}{G}$
   $F_{n,G+2-g}=F_{n,G+2-g}-Y_n$
   for $v=G_n+1:G+1-g$
   $F_{n,v}=F_{n,v}+\frac{\mathbb{I}_{g=1}(g) m_b T_{\text{syb}} Y_n}{(G-G_n)\tau}+\frac{\mathbb{I}_{g\geq 2}(g) Y_n}{G+1-g-G_n}$
   end for
   end for
3) **end if**
4) **end for**

In Algorithm 4, G_n is the allocated group index of UE n from Algorithm 3. $F_{n,g}$ denotes the allocated frequency at the BS to the n-th UE when the g-th group is in transmitting and $F_{n,G+1}$ is the allocated frequency to the n-th UE in the computation phase. In Algorithm 4, we first allocate the required CPU resources/frequencies (for remote processing at the MEC server) equally to all available time slots among all UEs. Then, we check each time slot to make sure whether there is a violation at the MEC server (i.e., $\sum_{n=1}^{N} F_{n,G+2-g} > F_{\max}$, for the g-th time slot). If there is no violation, we keep the previous CPU frequency allocation as the optimal strategy (following the characterizations in Lemmas 1 and 2). If there is any violation in the currently checked time slot, we then spread the overflowed required CPU frequencies to the unchecked time slots so that the allocated CPU frequencies in each time slot become as small as possible, thereby reducing the energy consumption and hence improving the energy efficiency, as shown in Fig. 3. Note that in Fig. 3, $l \in \{1,...,N/G\}$ is the UE index within each group. With the introduction of the dynamic computational resource allocation, it is possible for us to dynamically update/allocate the remaining computational resources at the BS.

D. Problem Formulation in UE Grouping

When UE grouping is adopted, by jointly determining the UEs' offloading data bits $\{D_n\}$, the length of the UL phase m_a, CPU frequencies allocated to the tasks of different UEs among different group transmission durations as well as the computation phase $\{F_{n,g}\}$, and the RIS reflecting coefficients $\boldsymbol{\theta}$, we aim to achieve the optimal energy efficiency subject to both coding blocklength and maximum decoding error rate constraints. With the introduction of UE grouping method and dynamic CPU frequency allocation, the global energy efficiency maximization problem is formulated as follows:

$$\textbf{P3:} \quad \underset{\{D_n, m_a, F_{n,g}, \boldsymbol{\theta}\}}{\textbf{Maximize}} \ \frac{\sum_{n=1}^{N} I_n}{\sum_{n=1}^{N} E_n} \tag{35}$$

$$\textbf{s. t.} \quad \varepsilon_n \leq \varepsilon_{max,n}, \quad \forall n \in \mathcal{N} \tag{35a}$$

$$0 \leq D_n \leq I_n, \quad \forall n \in \mathcal{N} \tag{35b}$$

$$\frac{(I_n-D_n)X_n}{f_n} \leq M T_{\text{syb}}, \quad \forall n \in \mathcal{N} \tag{35c}$$

$$\left(\sum_{g=1}^{G} \frac{F_{n,g} m_a}{G}+F_{n,G+1} m_b\right) T_{\text{syb}} \geq D_n X_n, \quad \forall n \in \mathcal{N} \tag{35d}$$

$$\sum_{n=1}^{N} F_{n,g} \leq F_{\max}, \quad \forall g \in \{1,2,...,G+1\} \tag{35e}$$

$$m_a, m_b \in \mathbb{Z}. \tag{35f}$$

where $\varepsilon_{max,n}$ is the maximum decoding error rate constraint for the n-th UE. Moreover, (35b) is the range of offloaded data bits and (35c) provides the local computing delay constraint. (35d) ensures that all of UEs' offloaded tasks/services can be completed on time. (35e) is the maximum CPU frequency constraint at the BS (MEC server).

E. Four-step Alternating Optimization

Due to the non-convex constraints and the strongly coupled optimization variables $\{D_n, m_a, F_{n,g}, \boldsymbol{\theta}\}$, **P3** is a non-convex optimization problem, and a four-step alternating optimization method is introduced to decouple the optimization variables and solve the problem iteratively.

1) Optimization of the RIS Reflecting Coefficients: Our first step is to optimize the RIS reflecting coefficients. Note that the RIS will adjust its reflecting coefficients during different group transmissions, which means we need to obtain $\boldsymbol{\theta_g}$ for the g-th group by solving **P3A** below individually by fixing $D_n, F_{n,g}$ and m_a:

$$\textbf{P3A:} \quad \text{Find} \ \ \boldsymbol{\theta_g} \tag{36}$$

$$\textbf{s. t.} \quad |e^{i\theta_k}|=1, \quad \forall k \in \mathcal{K} \tag{36a}$$

$$\gamma_{l,g} \geq \gamma_{th,l,g}, \forall l \in \{1,...,N/G\} \tag{36b}$$

where $\gamma_{th,l,g}$ is the threshold SINR value with which the error rate constraint $\varepsilon_{max,l}$ for the l-th UE in the g-th group is attained with equality.

Following a similar analysis as in Section IV-B, we equivalently solve **P3A-1** instead of solving **P3A**:

$$\textbf{P3A-1:} \quad \underset{\boldsymbol{\Psi}}{\textbf{Maximize}} \ \frac{P_{1,g}(||\boldsymbol{h}_{d,1,g}^H||^2+\text{Tr}(\boldsymbol{W}_{1,g}\boldsymbol{\Psi}))}{\sigma^2} \tag{37}$$

$$\textbf{s. t.} \quad \boldsymbol{\Psi}_{k,k}=1, \quad k=1,2,...,K+1, \tag{37a}$$

$$P_{l,g}\text{Tr}(\boldsymbol{W}_{l,g}\boldsymbol{\Psi})-\gamma_{th,l,g}P_{1,g}\text{Tr}(\boldsymbol{W}_{1,g}\boldsymbol{\Psi}) \geq$$

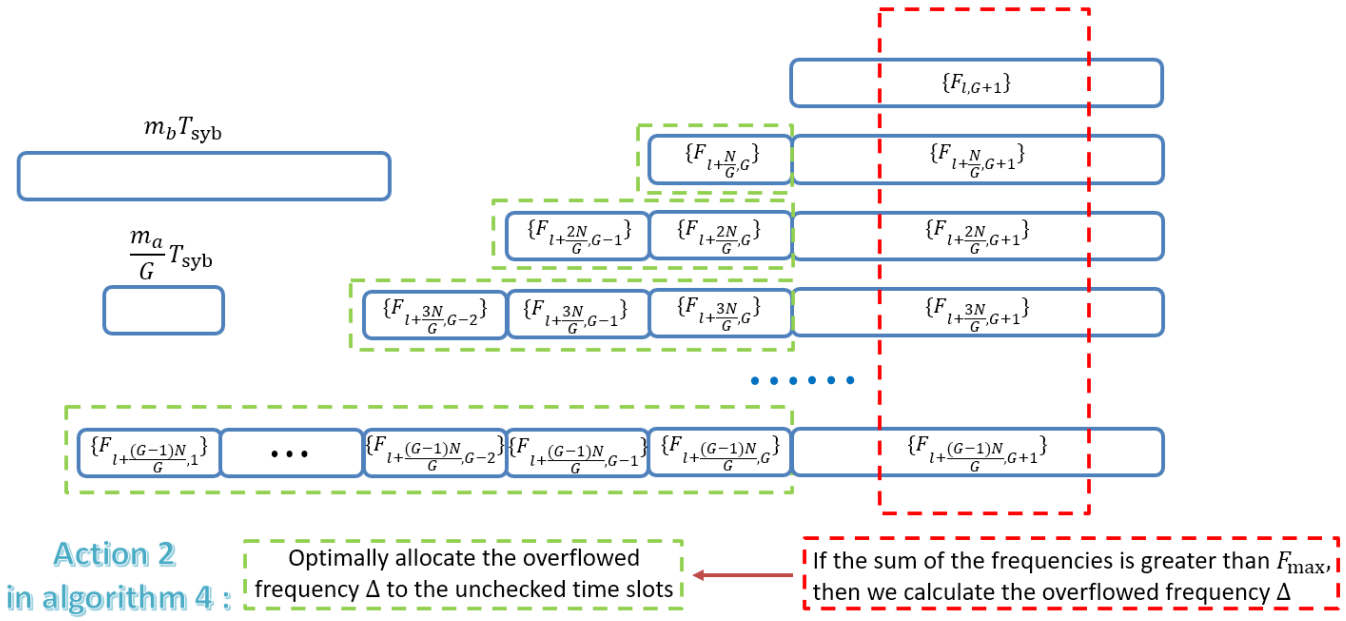


Fig. 3: Demonstration of action 2 in Algorithm 4.

$$\gamma_{th,l,g} P_{1,g} ||\boldsymbol{h}_{d,1}^H||^2 + \gamma_{th,l,g}\sigma^2 - P_{l,g}||\boldsymbol{h}_{d,l,g}^H||^2, \quad \forall l \in \{1, ..., N/G\} \tag{37b}$$

$$\boldsymbol{\Psi} \succeq \mathbf{0}. \tag{37c}$$

P3A-1 is a convex problem and with a standard convex optimization tool, it can be efficiently solved. If $\text{rank}(\boldsymbol{\Psi}) = 1$, an optimal solution $\widetilde{\phi}^*$ can be obtained from $\boldsymbol{\Psi} = \widetilde{\phi}\widetilde{\phi}^H$, otherwise we need to utilize the Gaussian randomization to recover a sub-optimal $\widetilde{\phi}^*$ [16][22]. **P3A** is processed individually for all G groups and we combine all the $\boldsymbol{\theta_g}^*$ to construct $\boldsymbol{\theta}^*$, which is denoted as $\boldsymbol{\theta}_i$ in the i-th iteration.

2) Optimization of the Offloaded Data Bits: With fixed $\boldsymbol{\theta} = \boldsymbol{\theta}_i$ and $m_a = m_{a,i-1}, F_{n,g} = F_{n,g,i-1}$, we have **P3B**:

$$\textbf{P3B:} \quad \underset{\{D_n\}}{\textbf{Maximize}} \quad \frac{\sum_{n=1}^N I_n}{\sum_{n=1}^N E_n} \tag{38}$$

$$\textbf{s. t.} \quad \varepsilon_n \le \varepsilon_{max,n}, \quad \forall n \in \mathcal{N} \tag{38a}$$

$$0 \le D_n \le I_n, \quad \forall n \in \mathcal{N} \tag{38b}$$

$$\frac{(I_n - D_n)X_n}{f_n} \le MT_{\text{syb}}, \quad \forall n \in \mathcal{N} \tag{38c}$$

$$\left(\sum_{g=1}^{G} \frac{F_{n,g} m_a}{G} + F_{n,G+1} m_b\right) T_{\text{syb}} \ge D_n X_n, \quad \forall n \in \mathcal{N} \tag{38d}$$

Here D_n^* can be obtained easily since **P3** becomes a convex problem in this case and we define $D_{n,i} = D_n^*$.

3) Optimization of the Offloading Blocklength: By fixing $\boldsymbol{\theta} = \boldsymbol{\theta}_i$, $D_n = D_{n,i}$ and $F_{n,g} = F_{n,g,i-1}$, **P3** now becomes **P3C**:

$$\textbf{P3C:} \quad \underset{m_a}{\textbf{Maximize}} \quad \frac{\sum_{n=1}^N I_n}{\sum_{n=1}^N E_n} \tag{39}$$

$$\textbf{s. t.} \quad \varepsilon_n \le \varepsilon_{max,n}, \quad \forall n \in \mathcal{N} \tag{39a}$$

$$\left(\sum_{g=1}^{G} \frac{F_{n,g} m_a}{G} + F_{n,G+1} m_b\right) T_{\text{syb}} \ge D_n X_n, \quad \forall n \in \mathcal{N} \tag{39b}$$

$$m_a, m_b \in \mathbb{Z}. \tag{39c}$$

The optimal $m_a{}^*$ can be determined either based on (39a) or (39b)[4]. We set $m_{a,i} = m_a^*$.

4) Optimization of the Allocated Frequency: In the forth step, once $m_{a,i}$ is obtained in the previous step, we fix $\boldsymbol{\theta} = \boldsymbol{\theta}_i$ and $D_n = D_{n,i}$ and $m_a = m_{a,i}$ to construct **P3D**:

$$\textbf{P3D:} \quad \underset{\{F_{n,g}\}}{\textbf{Maximize}} \quad \frac{\sum_{n=1}^N I_n}{\sum_{n=1}^N E_n} \tag{40}$$

$$\textbf{s. t.} \quad F_{n,g} = 0, \quad \forall g \in \{1, 2, ..., G_n\}, \forall n \in \mathcal{N} \tag{40a}$$

$$\left(\sum_{g=1}^{G} \frac{F_{n,g} m_a}{G} + F_{n,G+1} m_b\right) T_{\text{syb}} \ge D_n X_n, \quad \forall n \in \mathcal{N} \tag{40b}$$

$$\sum_{n=1}^{N} F_{n,g} \le F_{\max}, \quad \forall g \in \{1, 2, ..., G+1\} \tag{40c}$$

In **P3D**, (40a) ensures that no CPU frequency will be allocated to a user until the group that includes this user completes its offloading transmission. **P3D** can be solved via Algorithm 4.

By iteratively solving **P3A-1**, **P3B**, **P3C** and **P3D**, the solution of **P3** can be obtained once they converge. The four-step algorithm is described in Algorithm 5 below.

Algorithm 5 Four-step Alternating Optimization for **P3**

Initialization:

1) Initialize $\{D_{n,0}\}$, $m_{a,0}$, $\{F_{n,g,0}\}$.

Actions:

1) **For** $i = 1 : \text{I}_{\max}$
2) Obtain $\boldsymbol{\theta}_i$ by solving **P3A-1** with $\{D_{n,i-1}\}$, $m_{a,i-1}$, $\{F_{n,g,i-1}\}$.
3) Obtain $\{D_{n,i}\}$ by solving **P3B** with $\boldsymbol{\theta}_i$, $m_{a,i-1}$, $\{F_{n,g,i-1}\}$.
4) Obtain $m_{a,i}$ from (39a) in **P3C** with $\boldsymbol{\theta}_i$, $\{D_{n,i}\}$, $\{F_{n,g,i-1}\}$.

[4]Both (39a) and (39b) provide lower bounds on m_a, and the optimal blocklength will be determined by the larger of these lower bounds.

5) Obtain $\{F_{n,g,i}\}$ by solving **P3D** with $\boldsymbol{\theta}_i$, $m_{a,i}$, $\{D_{n,i}\}$ via Algorithm 4.
6) **End** if converges.

F. Convergence Analysis with UE Grouping, Hybrid NOMA-TDMA, and Dynamic Frequency Allocation

The convergence of Algorithm 5 is ensured by the following propositions.

In the UE grouping method, since the RIS will adjust its reflecting coefficients during different group transmissions, solving **P3A-1** to obtain $\boldsymbol{\theta_g}$ for the g-th group is exactly the same as solving **P2B-1** in Algorithm 2, and therefore **Proposition 1** still holds for different groups, i.e., $\gamma_{1,g}$ is non-decreasing as the iteration index i in the proposed Algorithm 5 increases.

Similarly as in the case with multiple UEs and NOMA, due to finite transmit power $P_{1,g}$, $\gamma_{1,g}$ should converge. Such conclusion can be verified for all G groups, and hence all $\gamma_{1,g}, \forall g \in \{1,2,...,G\}$ converge. Also note that

$$\frac{\partial \gamma_{1,g}}{\partial \boldsymbol{\Psi}} = \frac{\partial P_{1,g}(||\boldsymbol{h}_{d,1,g}^H||^2 + \text{Tr}(\boldsymbol{\Psi}\boldsymbol{W}_{1,g}))}{\sigma^2 \partial \boldsymbol{\Psi}} = \frac{P_1 \boldsymbol{W}_{1,g}^T}{\sigma^2} \neq \mathbf{0}. \tag{41}$$

When $\gamma_{1,g}$ converges, $\boldsymbol{\Psi}$ should converge simultaneously in the g-th group transmission. Note that since the derivative above is nonzero, any variation in $\boldsymbol{\Psi}$ will lead to a variation in $\gamma_{1,g}$, contradicting its convergence. Based on (23), we know $\gamma_{l,g}$ is only decided by $\boldsymbol{\Psi}$. Consequently, $\{\gamma_{l,g}\}, \forall l \in \{1,...,N/G\}$ should also converge.

To illustrate the convergence of $\{D_n\}$, we first have the following proposition:

Proposition 3: *The allocation $\{F_{n,g}\}$ obtained from dynamic CPU frequency allocation can support/process the same amount of CPU cycles at the MEC server compared to the non-dynamic frequency allocation considered in Section IV.*

Proof: In step 2 of the action phase of Algorithm 4, due to the indicator function, we prove Proposition 3 considering the following two cases:

I. When $g = 1$, $\quad \forall n \in \{1,2,...,N - \frac{N}{G}\}$:

$$\begin{aligned}
&m_b T_{\text{syb}}(F_{n,G+1} - Y_n) + \sum_{v=G_n+1}^{G} (F_{n,v} + \tfrac{m_b T_{\text{syb}} Y_n}{(G-G_n)\tau})\tau \\
&= m_b T_{\text{syb}} F_{n,G+1} - m_b T_{\text{syb}} Y_n + \sum_{v=G_n+1}^{G} (F_{n,v}\tau + \tfrac{m_b T_{\text{syb}} Y_n}{G-G_n}) \\
&= m_b T_{\text{syb}} F_{n,G+1} + \sum_{v=G_n+1}^{G} F_{n,v}\tau - m_b T_{\text{syb}} Y_n + \tfrac{G-G_n-1+1}{G-G_n} m_b T_{\text{syb}} Y_n \\
&= m_b T_{\text{syb}} F_{n,G+1} + \sum_{v=G_n+1}^{G} F_{n,v}\tau \\
&= D_n X_n.
\end{aligned}$$

II. When $g \geq 2$, $\quad \forall n \in \{1,2,...,N - \frac{Ng}{G}\}$:

$$\begin{aligned}
&(F_{n,G+2-g} - Y_n)\tau + \sum_{v=G_n+1}^{G+1-g} (F_{n,v} + \tfrac{Y_n}{G+1-g-G_n})\tau \\
&= \tau F_{n,G+2-g} - Y_n\tau + \sum_{v=G_n+1}^{G+1-g} F_{n,v}\tau + \sum_{v=G_n+1}^{G+1-g} \tfrac{Y_n\tau}{G+1-g-G_n} \\
&= \tau F_{n,G+2-g} - Y_n\tau + \sum_{v=G_n+1}^{G+1-g} F_{n,v}\tau + \tfrac{G+1-g-G_n-1+1}{G+1-g-G_n} Y_n\tau \\
&= \tau F_{n,G+2-g} + \sum_{v=G_n+1}^{G+1-g} F_{n,v}\tau \\
&= D_n X_n - m_b T_{\text{syb}} F_{n,G+1} - \sum_{u=1}^{g-2} F_{n,G+2-g+u}\tau.
\end{aligned}$$

In both cases, we observe that the total number of supported/processed CPU cycles at the MEC server after adopting dynamic CPU frequency allocation via Algorithm 4 remains the same, i.e., it is independent of Y_j, which is the optimal change in the allocated CPU frequency in dynamic allocation in Algorithm 4. With this, **Proposition 3** has been proved. □

We consequently have the following **Proposition 4**:

Proposition 4: *D_n is non-increasing as the iteration index i in Algorithm 5 increases*[5].

Proof: **P3D** is a convex problem when $\boldsymbol{\theta}$, $\{D_n\}$ and m_a are fixed. According to (8), (9), (10) and (11), we know that smaller $\{F_{n,g}\}$ leads to a higher energy efficiency. Considering the constraint in (40b), via Algorithm 4 in the i-th iteration we have

$$\left(\sum_{g=1}^{G} \frac{F_{n,g,i} m_a}{G} + F_{n,G+1,i} m_b\right) T_{\text{syb}} = D_{n,i} X_n. \quad \forall n \in \mathcal{N} \tag{42}$$

$D_{n,i}X_n$ is the amount of total processed CPU cycles at the MEC server in the i-th iteration. According to **Proposition 3**, $D_{n,i}X_n$ remains the same regardless of whether the dynamic CPU frequency allocation is performed or not. Therefore, in the $i+1$-th iteration, based on (38d) in **P3B**, we have

$$D_{n,i+1} \leq \left(\sum_{g=1}^{G} \frac{F_{n,g,i} m_a}{G} + F_{n,G+1,i} m_b\right) \frac{T_{\text{syb}}}{X_n} = D_{n,i}. \tag{43}$$

Therefore, D_n is non-increasing for all UEs. □

We further have $D_n \geq I_n - \frac{M T_{\text{syb}} f_n}{X_n}$ from (38C) in **P3B**. Since the right side of this inequality is a constant, combining with **Proposition 4**, $\{D_n\}$ will converge.

With converged $\{\gamma_{l,g}\}$ and $\{D_n\}$, when m_a is determined by (39a) in **P3C**, which is independent of $\{F_{n,g}\}$, m_a will converge.

When m_a is determined by (39b), in this case the equality of (39b) must hold, and thereby in the $i+1$-th iteration we have

$$\left(\sum_{g=1}^{G} \frac{F_{n,g,i} m_{a,i+1}}{G} + F_{n,G+1,i} m_{b,i+1}\right) T_{\text{syb}} = D_{n,i+1} X_n. \tag{44}$$

In the i-th iteration, Algorithm 4 assures the equality of (40b), and hence we have

$$\left(\sum_{g=1}^{G} \frac{F_{n,g,i} m_{a,i}}{G} + F_{n,G+1,i} m_{b,i}\right) T_{\text{syb}} = D_{n,i} X_n \tag{45}$$

When $\{D_n\}$ converges, $D_{n,i+1} = D_{n,i}$. Based on (44) and (45), we further have:

$$\begin{aligned}
&\left(\sum_{g=1}^{G} \tfrac{F_{n,g,i} m_{a,i+1}}{G} + F_{n,G+1,i} m_{b,i+1}\right) T_{\text{syb}} \\
&= \left(\sum_{g=1}^{G} \tfrac{F_{n,g,i} m_{a,i}}{G} + F_{n,G+1,i} m_{b,i}\right) T_{\text{syb}},
\end{aligned}$$

and thereby the optimal m_a^* in the $i+1$-th iteration is exactly the same as the optimal m_a^* in the i-th iteration, i.e., $m_{a,i+1} = m_{a,i}$. Consequently, m_a will converge in all cases.

Finally, with converged $\{\gamma_{l,g}\}$, $\{D_n\}$ and m_a, $\{F_{n,g}\}$ should also converge since **P3D** is a convex problem.

[5]In order to achieve the optimal solutions, we set $D_n = I_n$ for all UEs in the initialization of Algorithm 5.

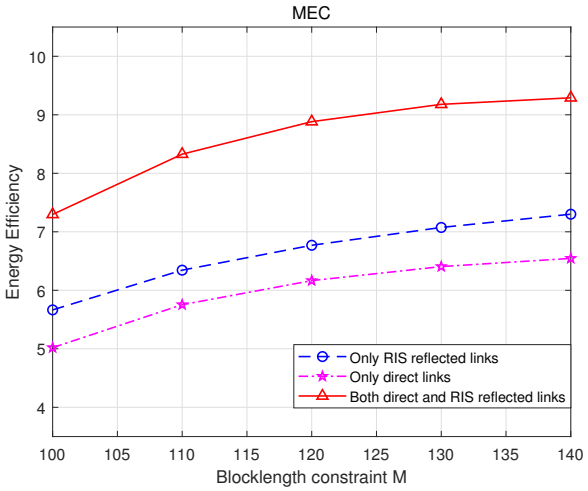


Fig. 4: Baseline comparisons for single UE under different channel conditions.

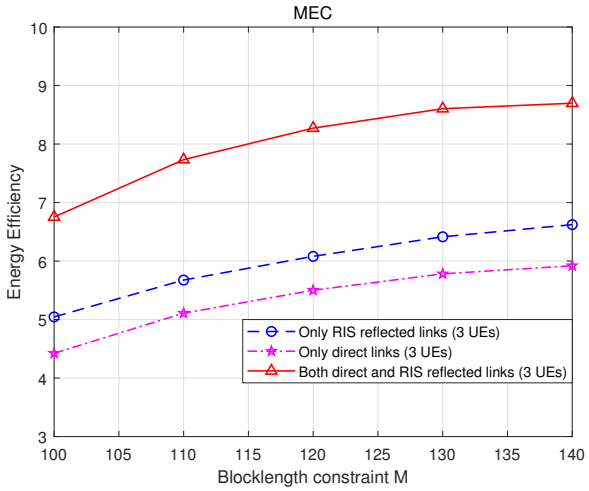


Fig. 5: Baseline comparisons for 3 UEs under different channel conditions.

Therefore, Algorithm 5 will converge for all UEs.

G. Stopping Point Analysis with UE Grouping in Hyrid NOMA-TDMA

Below, we first identify the stopping condition of Algorithm 5 in the remark and subsequently discuss how this result is established.

Remark 3: *Algorithm 5 will stop in the $j+1$-th iteration when in every NOMA group there is at least one UE (assume, without loss of generality, that it is the (p,g)-th UE in the g-th group) whose offloaded data $D_{(p,g),j}$ is determined via (38a), i.e., $\varepsilon_{(p,g),j} = \varepsilon_{max,(p,g)}$ in solving problem **P3B** in the j-th iteration.*

This result can be established in the same way as in the multiple-UE case in Section IV(D), which can be considered as the special case when $G = 1$. With TDMA, there is only one group transmitting at one time, and during its transmission period the decision-making is exactly the same as in the multiple-UE case in Section IV(D). Therefore, in the j-th iteration, if every NOMA group has at least one UE (e.g., UE (p,g)) whose $D_{(p,g),j}$ is determined via (38a), i.e., $\varepsilon_{(p,g),j} = \varepsilon_{max,(p,g)}$ in solving problem **P3B**, Algorithm 5 will stop in the $j+1$-th iteration. □

VI. Numerical Results

In this section, we conduct a numerical analysis and determine the maximum energy efficiency in the network under different scenarios. In the simulations, the channels are generated by $\boldsymbol{h}_{l,n} = \sqrt{\xi_0 d_{l,n}^{-\alpha_{l,n}}}\widetilde{\boldsymbol{g}}_{l,n}$, $l \in \{d,r\}$ and $\boldsymbol{\mathcal{G}} = \sqrt{\xi_0 d_B^{-\alpha_B}}\widetilde{\boldsymbol{g}}_B$. $d_{l,n}$, $\alpha_{l,n}$ and $\widetilde{\boldsymbol{g}}_{l,n}$ denote the distance to the RIS/BS, path loss exponent, and complex Gaussian distributed fading components for the n-th UE, respectively. Similarly, d_B, α_B, $\widetilde{\boldsymbol{g}}_B$ are the distance from the RIS to the BS, path loss exponent, and complex Gaussian distributed fading components of such links. The channel simulation parameters setting is listed in Table. II below.

Parameter	Definition	Value
$\alpha_{d,n}$	Path loss exponent for the n-th UE to the BS	5
$\alpha_{r,n}$	Path loss exponent for the n-th UE to the RIS	2
α_B	Path loss exponent from the RIS to the BS	3.5
ξ_0	Path loss at the reference point $d_0 = 1$ m	-30 dB
σ^2	Noise power	-95 dBm
X_n	Task intensity for the n-th UE	500 cycles/bit

TABLE II: Summary of channel parameters.

In the simulation results, we first provide Fig. 4 and Fig. 5 as baseline comparisons in which energy efficiency curves are plotted as a function of the blocklength. In these figures, we compare the performances when only direct links, only RIS reflected links, or both direct and RIS reflected links are available in the cases of single UE and 3 UEs, respectively. We observe in these figures that the highest energy efficiency levels are attained when both direct and RIS links are present. We also notice that having only RIS-reflected links leads to higher energy efficiency than that with only direct links, highlighting the benefits of deploying RIS in the environment.

We then analyze the performance of the proposed alternating optimization algorithm for a single UE in Fig. 6 for different number of RIS elements. We immediately notice that the energy efficiency is improved when the blocklength constraint M increases, which is expected since increasing M is the same as loosening the latency constraint, resulting in more time being left for the MEC server to process the offloaded task. We further observe that enlarging the number of RIS elements improves the performance as well. By increasing the number of RIS elements, we are more likely to obtain our desired SNR and hence improve the energy efficiency.

Next, we analyze the performance of the proposed optimization algorithm for 3 UEs in Fig. 7 and Fig. 8, where the curves of energy efficiency versus blocklength constraint M are plotted. In Fig. 7, different curves are for different number of RIS elements. From Fig. 7, we observe that larger blocklength constraint M leads to a better energy efficiency, which is again expected since increasing M results in more time that can be used by the MEC server to process the offloaded tasks. We further observe that the performance is improved when the number of RIS elements increases. The increase in the number of RIS elements provides us with higher degrees of freedom to achieve the desired SINR levels, thus improving the energy efficiency.

In Fig. 8, different curves stand for different maximum CPU frequency ($F_{\max}$) constraints at the MEC server. Similar to Fig. 7, the energy efficiency again improves with the increase

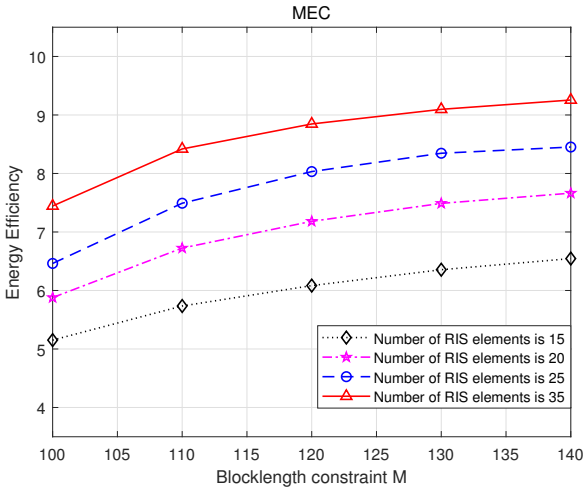


Fig. 6: Optimized energy efficiency for single UE with different number of RIS elements.

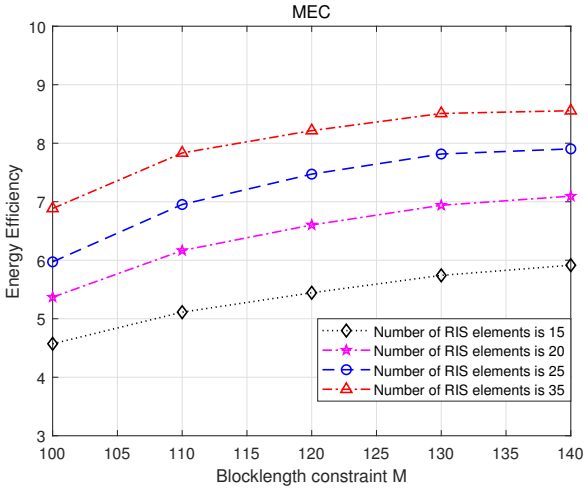


Fig. 7: Optimized energy efficiency for 3 UEs with different number of RIS elements.

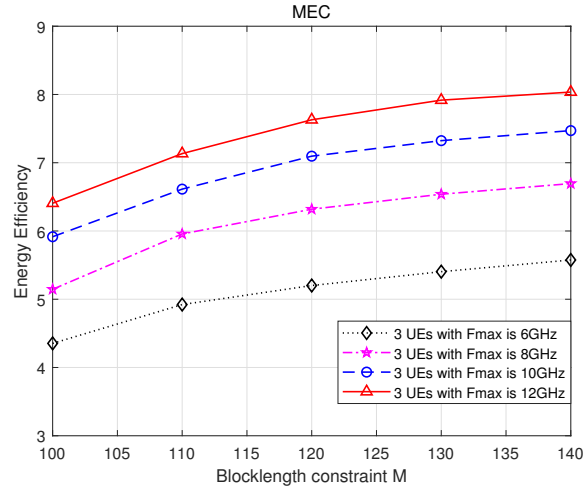


Fig. 8: Optimized energy efficiency for 3 UEs with different maximum CPU frequency at the BS.

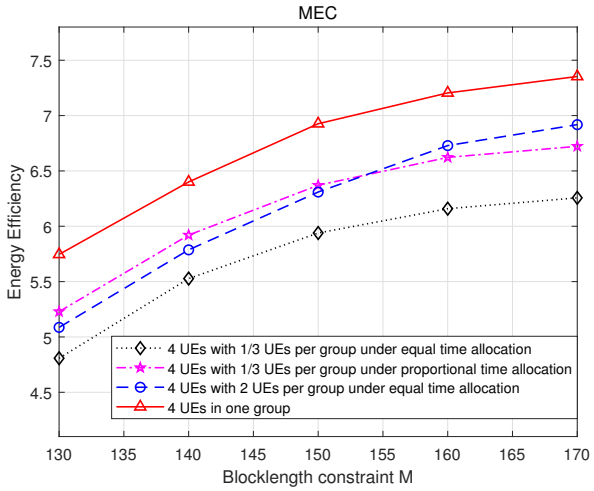


Fig. 9: Optimized energy efficiency for 4 UEs with different grouping method.

in M. Additionally, increasing the maximum CPU frequency $F_{\max}$ at the MEC server also enhances the energy efficiency. This is due to the fact that with larger $F_{\max}$, more data can be processed at the MEC server, and based on our assumption that $\Gamma_M \ll \Gamma_n, \forall n \in \mathcal{N}$, less energy will be consumed for processing the same amount of data, resulting in a better energy efficiency.

In Fig. 9, we compare different grouping methods in the case of 4 UEs. We have employed 4 grouping methods in the numerical results. The first one is the case with all 4 UEs in a group (indicating that NOMA is utilized). In the other cases, hybrid NOMA-TDMA is employed. In particular, second scheme is 2 UEs per group and we have 2 groups in total, and the offloading time is allocated equally to each group as $\frac{m_a T_{\text{syb}}}{2}$ seconds. In the third and fourth methods, we still have 2 groups while one group has only 1 UE and the other includes 3 UEs. The difference between the last two methods is in the offloading time allocation. In the third grouping method, we equally allocate the total offloading time of $m_a T_{\text{syb}}$ seconds to the 2 groups, and each group has $\frac{m_a T_{\text{syb}}}{2}$ seconds in the offloading phase, which is similar to the second grouping method. In the fourth method, we proportionally allocate the offloading time, which means that for the group with only 1 UE, the offloading time is $\frac{m_a T_{\text{syb}}}{4}$ seconds, and for the group with 3 UEs, the offloading time is $\frac{3m_a T_{\text{syb}}}{4}$ seconds. From Fig. 6, we can observe that the more UEs we have in one group, the better energy efficiency we can obtain. This is because if more UEs are in one group, we can better take advantage of NOMA transmissions. Furthermore, proportional offloading time allocation outperforms equal offloading time allocation, which is due to the fact that the more offloading time we allocate to the group with more UEs, the more benefits we can obtain by utilizing NOMA transmissions. Note that even though all 4 UEs in one group attains the best performance, this requires a much higher runtime compared with other grouping methods.

Furthermore, considering UE grouping and dynamic CPU frequency allocations, we initially demonstrate the performance with 6 UEs and compare the energy efficiency with and without dynamic CPU frequency allocation, and then we move to the case in which we adjust the UE grouping utility

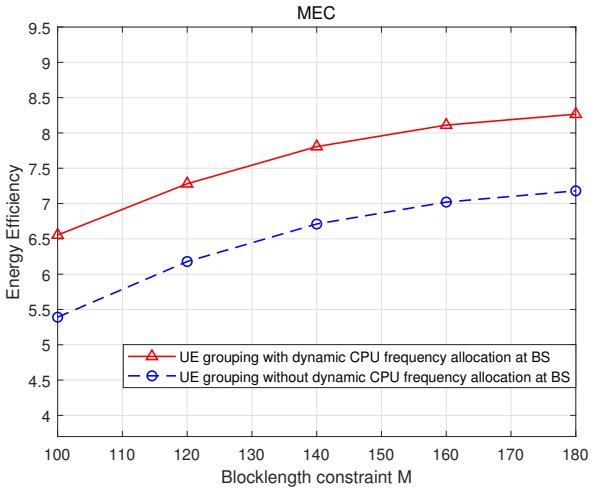


Fig. 10: Optimized energy efficiency for the case of 6 UEs with and without dynamic CPU frequency allocation.

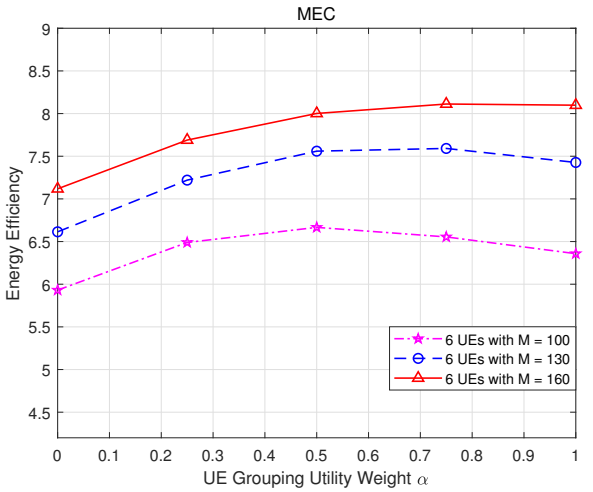


Fig. 11: Optimized energy efficiency for the case of 6 UEs with different UE grouping utility weights.

weight. In this case, we set $F_{\max} = 10$ GHz at the MEC server.

Specifically, we analyze the performance of the proposed UE grouping algorithm for 6 UEs in Fig. 10 and Fig. 11. In Fig. 10, the curves of energy efficiency versus blocklength constraint M at the MEC server are plotted with and without dynamic CPU frequency allocation at the BS. From Fig. 10, we observe that the energy efficiency with dynamic CPU frequency allocation always exceeds the one that does not adopt the dynamic CPU frequency allocation, which is expected since utilizing dynamic CPU frequency allocation is equivalent to extending the MEC processing time duration, resulting in more time that can be used by the MEC server to process the offloaded tasks, thereby improving the energy efficiency.

In Fig. 11, different curves are obtained under different blocklength constraints. Specifically, we considered three different values for the blocklength constraint, i.e., $M = 100, 130, 160$. The curves are plotted as a function of the weight α in the grouping utility metric defined in (32). We first observe that there is an optimal weight value that maximizes the energy efficiency. Such optimal α balances the importance between the channel gain and latency constraint. We further observe that the optimal value of α becomes larger as the blocklength constraint M is relaxed and M becomes larger. This is mainly because with larger M, more time can be allocated to the offloading transmission (in the UL phase) and the UE with relatively favorable channel conditions benefits more from a larger m_a since it can offload more data bits to the BS (MEC server), and therefore reduces the energy consumption, leading to improved energy efficiency.

VII. Conclusion

In this work, we have analyzed an RIS-assisted MEC network aiming to maximize the energy efficiency under both coding blocklength and maximum decoding error rate constraints in a low-latency scenario. We have initially investigated the single UE case and proposed an alternating optimization method to solve the problem. Extending the system model, we subsequently investigated an MEC network with multiple UEs in which NOMA transmission is adopted. We constructed a three-step alternating optimization algorithm to tackle the problem, and conducted a convergence analysis. Furthermore, we have proposed a UE grouping method (and hybrid NOMA-TDMA transmissions) to alleviate the required runtime when the number of UEs increases. We have also developed a dynamic CPU frequency allocation algorithm to better take advantage of the UE grouping method. Numerical results demonstrate that the proposed alternating optimization algorithms can solve the optimization problems efficiently. We have observed that with larger blocklength value M and CPU frequency $F_{\max}$ at the MEC server, the energy efficiency is improved. It is also noted that adjusting the RIS phase shift matrix is equivalent to improving the SINR at the BS and such an enhanced SINR leads to a higher energy efficiency. Furthermore, the proposed dynamic CPU frequency allocation algorithm can improve the performance substantially. It is also noted that increasing M leads to a larger optimal value for the UE grouping utility weight α and such an optimal α leads to a higher energy efficiency. Our future work will address the impact of having multiple MEC servers in the network.

References

[1] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge AI: Intelligentizing mobile edge computing, caching and communication by federated learning," *IEEE Netw.*, vol. 33, no. 5, pp. 156–165, 2019.

[2] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 2017.

[3] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, 2016.

[4] Y. Mao, J. Zhang, and K. B. Letaief, "Dynamic computation offloading for mobile-edge computing with energy harvesting devices," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3590–3605, 2016.

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

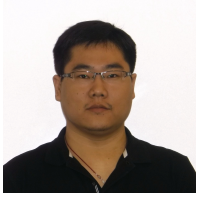
[6] Q. Wu, S. Zhang, B. Zheng, C. You, and R. Zhang, "Intelligent reflecting surface aided wireless communications: A tutorial," *IEEE Trans. Commun.*, 2021.

[7] Q. Wu and R. Zhang, "Towards smart and reconfigurable environment: Intelligent reflecting surface aided wireless network," *IEEE Commun. Mag.*, vol. 58, no. 1, pp. 106–112, 2019.

[8] Q. Wu and R. Zhang, "Intelligent reflecting surface enhanced wireless network: Joint active and passive beamforming design," in *IEEE Global Commun. Conf. (GLOBECOM)*, pp. 1–6, IEEE, 2018.

[9] Q. Wang, F. Zhou, R. Q. Hu, and Y. Qian, "Energy efficient robust beamforming and cooperative jamming design for IRS-assisted MISO

networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 4, pp. 2592–2607, 2020.
[10] Q. Wang, R. Q. Hu, Y. Qian, *et al.*, "Hierarchical Energy-Efficient Mobile-Edge Computing in IoT Networks," *IEEE Internet Things J.*, vol. 7, no. 12, pp. 11626–11639, 2020.
[11] P. Popovski, "Ultra-reliable communication in 5G wireless systems," in *1st International Conference on 5G for Ubiquitous Connectivity*, pp. 146–151, IEEE, 2014.
[12] H. Chen, R. Abbas, P. Cheng, M. Shirvanimoghaddam, W. Hardjawana, W. Bao, Y. Li, and B. Vucetic, "Ultra-reliable low latency cellular networks: Use cases, challenges and approaches," *IEEE Commun. Mag.*, vol. 56, no. 12, pp. 119–125, 2018.
[13] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Transactions on Information Theory*, vol. 56, no. 5, pp. 2307–2359, 2010.
[14] R. Hashemi, S. Ali, N. H. Mahmood, and M. Latva-aho, "Average Rate and Error Probability Analysis in Short Packet Communications over RIS-aided URLLC Systems," *arXiv preprint arXiv:2102.13363*, 2021.
[15] C. Huang, A. Zappone, G. C. Alexandropoulos, M. Debbah, and C. Yuen, "Reconfigurable intelligent surfaces for energy efficiency in wireless communication," *IEEE Trans. Wireless Commun.*, vol. 18, no. 8, pp. 4157–4170, 2019.
[16] Q. Wu and R. Zhang, "Intelligent reflecting surface enhanced wireless network via joint active and passive beamforming," *IEEE Trans. Wireless Commun.*, vol. 18, no. 11, pp. 5394–5409, 2019.
[17] D. Zhang, Q. Wu, M. Cui, G. Zhang, and D. Niyato, "Throughput Maximization for IRS-Assisted Wireless Powered Hybrid NOMA and TDMA," *IEEE Wireless Commun. Lett.*, 2021.
[18] G. Chen, Q. Wu, W. Chen, D. W. K. Ng, and L. Hanzo, "IRS-aided wireless powered MEC systems: TDMA or NOMA for computation offloading?," *IEEE Trans. Wireless Commun.*, 2022.
[19] M. Zeng, X. Li, G. Li, W. Hao, and O. A. Dobre, "Sum rate maximization for IRS-assisted uplink NOMA," *IEEE Commun. Lett.*, vol. 25, no. 1, pp. 234–238, 2020.
[20] Z. Yang, Y. Liu, Y. Chen, and N. Al-Dhahir, "Machine learning for user partitioning and phase shifters design in RIS-aided NOMA networks," *IEEE Trans. Commun.*, vol. 69, no. 11, pp. 7414–7428, 2021.
[21] M. NaseriTehrani and S. Farahmand, "Resource Allocation for IRS-Enabled Secure Multiuser Multi-Carrier Downlink URLLC Systems," in *2022 IEEE 23rd International Workshop on Signal Processing Advances in Wireless Communication (SPAWC)*, pp. 1–5, IEEE, 2022.
[22] Y. Yang, Y. Hu, and M. C. Gursoy, "Energy Efficiency Analysis in RIS-aided MEC Networks with Finite Blocklength Codes," in *2022 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 423–428, IEEE, 2022.
[23] Y. Yang and M. C. Gursoy, "Energy-Efficient Scheduling in RIS-aided MEC Networks with NOMA and Finite Blocklength Codes," in *2022 International Symposium on Wireless Communication Systems (ISWCS)*, pp. 1–6, 2022.
[24] M. S. Ali, H. Tabassum, and E. Hossain, "Dynamic user clustering and power allocation for uplink and downlink non-orthogonal multiple access (NOMA) systems," *IEEE Access*, vol. 4, pp. 6325–6343, 2016.



Yang Yang received his B.S. Degree in information engineering from Southeast University (SEU), China in 2013, the Master's Degree in electrical and computer engineering from Worcester Polytechnic Institute (WPI) in Massachusetts, US in 2016. Since 2016, he is a Ph.D. student at Syracuse University (SU). His research interests include unmanned aerial vehicle (UAV), mobile edge computing (MEC), machine learning, reconfigurable intelligent surface (RIS) and finite blocklength (FBL) regime.

Yulin Hu (Senior member, IEEE) received his M.Sc.E.E. degree from USTC, China, in 2011. He successfully defended his dissertation of a joint Ph.D. program supervised by Prof. Anke Schmeink at RWTH Aachen University and Prof. James Gross at KTH Royal Institute of Technology in Dec. 2015 and received his Ph.D.E.E. degree (Hons.) from RWTH Aachen University where he was a postdoctoral Research Fellow since Jan. to Dec. in 2016. He was a senior researcher and team leader with Prof. Anke Schmeink in ISEK research Area at RWTH Aachen University. From May to July in 2017, he was a visiting scholar with Prof. M. Cenk Gursoy in Syracuse University, USA. He is currently a professor with School of Electronic Information, Wuhan University, and a visiting professor with INDA Institute, RWTH Aachen University. His research interests are in information theory, optimal design of wireless communication systems. He has been invited to contribute submissions to multiple conferences. He was a recipient of the IFIP/IEEE Wireless Days Student Travel Awards in 2012. He received the Best Paper Awards at IEEE ISWCS 2017 and IEEE PIMRC 2017, respectively. He served as a TPC member for many conferences. He served as the WS&SS Chair for IEEE SmartData 2022, the track co-chair for ICCCN 2023, and the organizer and chair of special sessions in IEEE ISWCS 2018, 2021 and 2023. He is currently serving as an editor for IEEE Transactions on Vehicular Technology, Physical Communication (Elsevier), EURASIP Journal on Wireless Communications and Networking, and Frontiers in Communications and Networks.

M. Cenk Gursoy received the B.S. degree with high distinction in electrical and electronics engineering from Bogazici University, Istanbul, Turkey, in 1999 and the Ph.D. degree in electrical engineering from Princeton University, Princeton, NJ, in 2004. He was a recipient of the Gordon Wu Graduate Fellowship from Princeton University between 1999 and 2003. He is currently a Professor in the Department of Electrical Engineering and Computer Science at Syracuse University. His research interests are in the general areas of wireless communications, information theory, communication networks, signal processing, optimization and machine learning. He is an Editor for IEEE Transactions on Communications, and an Area Editor for IEEE Transactions on Vehicular Technology. He is on the Executive Editorial Committee of IEEE Transactions on Wireless Communications. He also served as an Editor for IEEE Transactions on Green Communications and Networking between 2016–2021, IEEE Transactions on Wireless Communications between 2010–2015 and 2017–2022, IEEE Communications Letters between 2012–2014, IEEE Journal on Selected Areas in Communications - Series on Green Communications and Networking (JSAC-SGCN) between 2015-2016, Physical Communication (Elsevier) between 2010–2017, and IEEE Transactions on Communications between 2013–2018. He has been the co-chair of the 2017 International Conference on Computing, Networking and Communications (ICNC) - Communication QoS and System Modeling Symposium, the co-chair of 2019 IEEE Global Communications Conference (Globecom) - Wireless Communications Symposium, the co-chair of 2019 IEEE Vehicular Technology Conference Fall - Green Communications and Networks Track, and the co-chair of 2021 IEEE Global Communications Conference (Globecom), Signal Processing for Communications Symposium. He received an NSF CAREER Award in 2006. More recently, he received the EURASIP Journal of Wireless Communications and Networking Best Paper Award, 2020 IEEE Region 1 Technological Innovation (Academic) Award, 2019 The 38th AIAA/IEEE Digital Avionics Systems Conference Best of Session (UTM-4) Award, 2017 IEEE PIMRC Best Paper Award, 2017 IEEE Green Communications & Computing Technical Committee Best Journal Paper Award, UNL College Distinguished Teaching Award, and the Maude Hammond Fling Faculty Research Fellowship. He is a Senior Member of IEEE, and is the Aerospace/Communications/Signal Processing Chapter Co-Chair of IEEE Syracuse Section.