Minimising Offloading Latency for Edge-Cloud Systems with Ultra-Reliable and Low-Latency Communications

Dang Van Huynh*, Van-Dinh Nguyen†, Saeed R. Khosravirad‡ and Trung Q. Duong*

*Queen's University Belfast, UK (e-mail: {dhuynh01, trung.q.duong}@qub.ac.uk)
†University of Luxembourg, Luxembourg (e-mail: dinh.nguyen@uni.lu)
‡Nokia Bell Labs, USA (e-mail: saeed.khosravirad@nokia-bell-labs.com)

ICC 2022 - IEEE International Conference on Communications | 978-1-5386-8347-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICC45855.2022.9839148

***Abstract*—We study a joint communication and computation offloading (JCCO) for hierarchical edge-cloud systems with ultra-reliable and low latency communications (URLLC). We aim to minimize the worst-case end-to-end (e2e) latency of computational tasks among multiple industrial Internet of Things (IIoT) devices by jointly optimizing offloading probabilities, processing rates, user association policies and power control subject to their service delay and energy consumption requirements as well as queueing stability conditions. To tackle the problem, we first decompose the original problem into two subproblems and then leverage the alternating optimization (AO) approach to solve them in an iterative fashion by developing newly convex approximate functions. The numerical results are provided to demonstrate the effectiveness of the proposed algorithms in terms of the e2e latency and convergence speed.**

I. Introduction

Task offloading designs for mobile-edge computing (MEC) have been widely investigated in the literature (see [1] and the references therein). In most existing works, energy-efficiency and delay-efficiency are considered as major figure-of-merit in designing task offloading schemes for MEC systems [2]. In particular, an offload forwarding scheme where fog servers (FSs) cooperate with each other to tackle their heterogeneousness in terms of commutation capacity and resources, improving the efficiency of power usage, was proposed in [2]. To guarantee low-latency wireless communication, short packets to convey a small amount of data must be used. This however will pose several challenges to design and optimize the performance of short packet-enabled networks since it demands for more resources (e.g., parity, redundancy) and ultrahigh reliability. Since then, resource allocation in the URLLC-based short block-length regime has recently studied to reduce the required bandwidth, the packet dropping, and maximize the energy efficiency (EE) [3], [4]. Focusing on designing URLLC-aware optimization for task offloading, the exponential-weight algorithm to balance URLLC constraints and energy consumption through online learning was investigated in [5]. The user-server association policy to reduce users' power consumption while trading off the resource allocations for local computation and task offloading was considered in [6].

Although ESs and FSs are often equipped with more powerful computing capability than end users, they are still limited compared to large-scale cloud data centers (CDCs) at the cloud server. Against the above background, this work proposes a novel joint communication and computation for URLLC-enabled hierarchical edge-cloud system, taking into account all the above issues. Our main contributions are summarised as follows. Firstly, we formulate the worst-case end-to-end (e2e) latency minimisation problem with three layers edge-cloud systems, which takes into account various aspects of joint communication and computation offloading. Then, we proposed the OA-IA based solution to deal with the problem. Finally, extensive numerical results are provided to evaluate the effectiveness of the proposed solution compared with other scheme such as random user association (RUA), fixed power, fixed frequency allocation, and without cloud (w/o Cloud).

II. System Model and Problem Formulation

We consider a hierarchical edge-cloud system consisting of the set $\mathcal{M} = \{1, 2, \cdots M\}$ of M UEs (IIoT devices) randomly distributed in a factory automation scenario, and the set $\mathcal{K} = \{1, 2, \cdots, K\}$ of K ESs at the edge layer. Each ES is co-located with an access point (AP) to communicate with UEs over URLLC wireless links. The ESs connect with the cloud server via wired fronthaul links. ***User layer*** includes multiple UEs. A task of UE $m \in \mathcal{M}$ can be executed locally with processing rate f_m^{lo} (cycles/s) or offloaded to a ES with probability of $\alpha_m \in [0, 1]$. We use the indicator vector $\boldsymbol{\pi} \triangleq [\pi_{mk}]_{\forall m,k}$ to denote the association between UEs and ESs. We assume that the tasks of a UE is only offloaded to one ES, i.e., $\sum_{k\in\mathcal{K}} \pi_{mk} = 1, \forall m$. ***Edge layer*** consists of K ESs placed close to UEs, where the processing rate of ES k is denoted as f_k^{es} (cycles/s). To minimize the processing latency while admitting computation tasks from multiple UEs, ES k can offload a portion of $\beta_{mk} \in [0, 1]$ of tasks of UE m to the cloud server through the fronthaul link. ***Cloud layer*** contains large-scale cloud data centers equipped with powerful processors having very high processing rate f^{cs} (cycles/s).

Suppose a task of UE m is characterised by a tuple $I_m \triangleq (D_m, C_m, T_m^{\max})$, in which D_m, C_m and $T_m^{\max}$ are the input

This work was supported in part by the U.K. Royal Academy of Engineering (RAEng) under the RAEng Research Chair and Senior Research Fellowship scheme Grant RCSRF2021\11\41. The work of V.-D. Nguyen was supported in part by the ERC AGNOSTIC project, ref. H2020/ERC2020POC/957570.

data size, the required computation resource (number of CPU cycles) and the maximum delay requirement, respectively. The task arrival rate of UE m is denoted as λ_m^{lo} (tasks/s).

A. Communication Model

Each AP is equipped with $L > 1$ antennas while each UE has single antenna. The channel vector between UE m and AP k, denoted by $\mathbf{h}_{mk} \in \mathbb{C}^{L\times 1}$, can be modeled as $\mathbf{h}_{mk} = \sqrt{g_{mk}}\bar{\mathbf{h}}_{mk}$, where g_{mk} is the large-scale channel coefficient including the pathloss and shadowing which is normalized by the noise power, and $\bar{\mathbf{h}}_{mk}$ is the small-scale fading following the Rayleigh fading model as $\bar{\mathbf{h}}_{mk} \sim \mathcal{CN}(0, \mathbf{I}_L)$. Under a shared wireless medium, the $L \times 1$ received signal vector at AP k can be expressed as $\mathbf{y}_k = \sum_{m\in\mathcal{M}} \mathbf{h}_{mk}\sqrt{p_m}s_m + \mathbf{z}_k$, where p_m and s_m are the transmit power and unit-power data symbom of UE m, respectively; $\mathbf{z}_k \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_L)$ is the additive white Gaussian background noise (AWGN).

Uplink Channel Estimation: The MMSE channel estimate of $\mathbf{h}_{mk}$ is given by [7]:

$$\hat{\mathbf{h}}_{mk} = \frac{g_{mk}Mp_m^p}{g_{mk}Mp_m^p + 1}\mathbf{y}_{mk}^p \tag{1}$$

which follows the distribution of $\mathcal{CN}\left(\mathbf{0}, \sigma_{mk}^2\mathbf{I}\right)$, where σ_{mk}^2 is given as $\sigma_{mk}^2 = g_{mk}^2Mp_m^p/(g_{mk}Mp_m^p + 1)$ and p_m^p is the pilot transmit power of UE m. According to the MMSE estimation property, the channel estimation error $\tilde{\mathbf{h}}_{mk} = \mathbf{h}_{mk} - \hat{\mathbf{h}}_{mk}$ is independent of $\hat{\mathbf{h}}_{mk}$ that follows the distribution of $\mathcal{CN}\left(\mathbf{0}, \delta_{mk}^2\mathbf{I}_L\right)$, where δ_{mk}^2 is given by $\delta_{mk}^2 = g_{mk}/(g_{mk}Mp_m^p + 1)$.

URLLC Uplink Transmission Rate: By employing the successive interference cancellation for signal detection in the uplink, the instantaneous signal-to-interference-plus-noise (SINR) of UE m can be expressed as

$$\gamma_m(\mathbf{p}, \boldsymbol{\pi}_m) = \frac{\sum\limits_{k\in\mathcal{K}} \pi_{mk}p_m\|\hat{\mathbf{h}}_{mk}\|^4}{\Phi_m(\mathbf{p}, \boldsymbol{\pi}_m)} \tag{2}$$

where

$$\Phi_m(\mathbf{p}, \boldsymbol{\pi}_m) \triangleq \sum_{i>m}\sum_{k\in\mathcal{K}} \pi_{mk}p_i|\hat{\mathbf{h}}_{mk}^H\hat{\mathbf{h}}_{ik}|^2 + \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{K}}(1-\pi_{mk})p_i \times |\hat{\mathbf{h}}_{mk}^H\hat{\mathbf{h}}_{ik}|^2 + \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{K}} \pi_{mk}p_i|\hat{\mathbf{h}}_{mk}\tilde{\mathbf{h}}_{ik}|^2 + \|\hat{\mathbf{h}}_{mk}\|^2$$

with $\mathbf{p} = \{p_m\}_{\forall m}$ and $\boldsymbol{\pi}_m = \{\pi_{mk}\}_{\forall k}$. In this paper, we focus the ergodic achievable rate of UE m, where $\bar{\gamma}_m(\mathbf{p}, \boldsymbol{\pi}_m) = \mathbb{E}\{\gamma_m(\mathbf{p}, \boldsymbol{\pi}_m)\}$ can be approximated by

$$\bar{\gamma}_m(\mathbf{p}, \boldsymbol{\pi}_m) = \frac{\sum\limits_{k\in\mathcal{K}} \pi_{mk}\left(L-1\right)p_m\sigma_{mk}^2}{\bar{\Phi}_m(\mathbf{p}, \boldsymbol{\pi}_m)} \tag{3}$$

with $\bar{\Phi}_m(\mathbf{p}, \boldsymbol{\pi}_m) \triangleq \sum\limits_{i>m}\sum\limits_{k\in\mathcal{K}} \pi_{mk}p_i\sigma_{i,k}^2 + \sum\limits_{i\in\mathcal{M}\setminus m}\sum\limits_{k\in\mathcal{K}}\left(1-\pi_{mk}\right)p_i\sigma_{i,k}^2 + \sum\limits_{i\in\mathcal{M}}\sum\limits_{k\in\mathcal{K}}\pi_{mk}p_i\delta_{i,k}^2 + 1$, whose derivation is based on [7]

The uplink achievable data rate of UE m (in bits/s) under URLLC finite blocklength can be approximated as:

$$R_m\left(\mathbf{p}, \boldsymbol{\pi}_m\right) = \left(1-\omega\right)B\log_2\left[1+\bar{\gamma}_m\left(\mathbf{p}, \boldsymbol{\pi}_m\right)\right] - B\sqrt{\frac{\left(1-\omega\right)V_m\left(\mathbf{p}, \boldsymbol{\pi}_m\right)}{N}}\frac{Q^{-1}\left(\epsilon_m\right)}{\ln 2} \tag{4}$$

where $N = \Delta_t B$ denotes the blocklength with Δ_t being the transmission time interval and $\omega \triangleq M/N$; ϵ_m is the decoding error probability, $Q^{-1}(\cdot)$ is the inverse function of $Q(x)$, and $V\left(\mathbf{p}, \boldsymbol{\pi}_m\right)$ is the channel dispersion given by $V_m\left(\mathbf{p}, \boldsymbol{\pi}_m\right) = 1 - \left[1+\bar{\gamma}_m\left(\mathbf{p}, \boldsymbol{\pi}_m\right)\right]^{-2}$.

B. Computation Model

Local Processing Latency: UE m can partially offload with the portion α_m of its task to the ES. The latency to process the remaining task at UE m with the processing rate f_m^{lo} is

$$t_m^{\mathsf{lo}}(\alpha_m, f_m^{\mathsf{lo}}) = \frac{\left(1-\alpha_m\right)C_m}{f_m^{\mathsf{lo}}}. \tag{5}$$

Wireless transmission latency: Given the uplink data rate in (4), the latency to transmit the portion α_m of UE m's task is caculated as

$$t_m^{\mathsf{co}}\left(\alpha_m, \mathbf{p}, \boldsymbol{\pi}_m\right) = \frac{\alpha_m D_m}{R_m\left(\mathbf{p}, \boldsymbol{\pi}_m\right)}. \tag{6}$$

ES Processing Latency: Let λ_k^{es} and λ_m^{lo} be the arrival rates of tasks at ES k and UE m, respectively. We have $\lambda_k^{\mathsf{es}} = \sum\limits_{m\in\mathcal{M}} \pi_{mk}\alpha_m\lambda_m^{\mathsf{lo}}$ [8]. We denote by $\beta_{mk} \in [0,1]$ the offloading portion of task m from ES k to CS. As a result, the task rate to process the remaining tasks offloaded from all UEs at ES k can be computed as $\mu_k^{\mathsf{es}} = f_k^{\mathsf{es}}/\sum\limits_{m\in\mathcal{M}} \pi_{mk}\alpha_m\left(1-\beta_{mk}\right)C_m$. By following the standard queuing model M/M/1 [8], we can compute the worst-case processing latency among ESs as

$$t^{\mathsf{es}}\left(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}\right) = \max_{\forall k\in\mathcal{K}}\left\{\frac{1}{\mu_k^{\mathsf{es}} - \lambda_k^{\mathsf{es}}}\right\} \tag{7}$$

where $\boldsymbol{\alpha} \triangleq \{\alpha_m\}_{\forall m}$ and $\boldsymbol{\beta} \triangleq \{\beta_{mk}\}_{\forall m,k}$.

Fronthaul Transmission Latency: Each ES k transmits the portion β_{mk} of the offloaded task $\pi_{mk}\alpha_m$ of all UEs $m \in \mathcal{M}$ to CS for further processing. The worst-case transmission latency to offload tasks from ESs to CS via fronthaul links can be expressed as

$$t^{\mathsf{fh}}\left(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}\right) = \max_{\forall k\in\mathcal{K}}\left\{\sum_{m\in\mathcal{M}} \pi_{mk}\alpha_m\beta_{mk}\frac{D_m}{R_k^{\mathsf{fh}}}\right\} \tag{8}$$

where R_k^{fh} is the fronthaul capacity between ES k and CS.

Cloud Processing Latency: The latency for the CS to process offloaded tasks ESs can be expressed as

$$t^{\mathsf{cs}}\left(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}\right) = \frac{1}{\mu^{\mathsf{cs}} - \lambda^{\mathsf{cs}}} \tag{9}$$

where $\mu^{\mathsf{cs}} = f^{\mathsf{cs}}/\sum\limits_{m\in\mathcal{M}}\sum\limits_{k\in\mathcal{K}} \pi_{mk}\alpha_m\beta_{mk}C_m$ and $\lambda^{\mathsf{cs}} = \sum\limits_{m\in\mathcal{M}}\sum\limits_{k\in\mathcal{K}} \pi_{mk}\alpha_m\beta_{mk}\lambda_m^{\mathsf{lo}}$ are considered as the task rate and the task arrival rate at CS, respectively.

C. Problem Formulation

From (5)–(9), the overall e2e latency of UE m is given by

$$t_m(f_m^{\mathsf{lo}}, \mathbf{p}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}) = t_m^{\mathsf{lo}}(\alpha_m, f_m^{\mathsf{lo}}) + t_m^{\mathsf{co}}\left(\alpha_m, \mathbf{p}, \boldsymbol{\pi}_m\right) + t^{\mathsf{es}}\left(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}\right) + t^{\mathsf{fh}}\left(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}\right) + t^{\mathsf{cs}}\left(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}\right). \tag{10}$$

The total energy of UE m consumed for the local processing and uplink transmission can be computed as

$$E_m(\alpha_m, f_m^{\mathsf{lo}}, \mathbf{p}, \boldsymbol{\pi}) = (1-\alpha_m)\frac{\theta_m}{2} C_m (f_m^{\mathsf{lo}})^2 + p_m \frac{\alpha_m D_m}{R_m(\mathbf{p}, \boldsymbol{\pi}_m)} \quad (11)$$

where the constant $\theta_m/2$ denotes the average switched capacitance and the average activity factor of UE m [6].

The joint communication and computation offloading (JCCO) problem that aims to minimize the worst-case e2e latency among computational tasks under their service delay and energy consumption requirements is formulated as

$$\textsf{JCCO:} \underset{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi},\mathbf{p},\mathbf{f}}{\text{minimize}} \max_{\forall m \in \mathcal{M}} \left\{ t_m\left(f_m^{\mathsf{lo}}, \mathbf{p}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}\right)\right\} \quad (12a)$$

$$\text{s.t. } t_m\left(f_m^{\mathsf{lo}}, \mathbf{p}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}\right) \le T_m^{\max}, \forall m \quad (12b)$$

$$E_m(\alpha_m, f_m^{\mathsf{lo}}, \mathbf{p}, \boldsymbol{\pi}) \le E_m^{\max}, \forall m \quad (12c)$$

$$R_m(\mathbf{p}, \boldsymbol{\pi}_m) \ge \sum_{k \in \mathcal{K}} \pi_{mk} R_m^{\min}, \forall m \quad (12d)$$

$$\lambda_k^{\mathsf{es}} \le \mu_k^{\mathsf{es}}, \forall k \quad (12e)$$

$$\lambda^{\mathsf{cs}} \le \mu^{\mathsf{cs}}, \quad (12f)$$

$$\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathscr{D}, \boldsymbol{\pi} \in \Pi, \mathbf{p} \in \mathscr{P}, \mathbf{f} \in \mathscr{F} \quad (12g)$$

where $\mathscr{D} \triangleq \{\alpha_m, \beta_{mk}, \forall m,k | 0 \le \alpha_m \le 1, 0 \le \beta_{mk} \le 1, \forall m,k\}$, $\mathscr{P} \triangleq \{p_m, \forall m | 0 \le p_m \le P_m^{\max}, \forall m\}$, $\mathscr{F} \triangleq \{f_m^{\mathsf{lo}}, \forall m | 0 \le f_m^{\mathsf{lo}} \le F_m^{\max}, \forall m\}$, and $\Pi \triangleq \{\pi_{mk}, \forall m,k | \pi_{mk} \in \{0,1\} \,\&\, \sum_{k \in \mathcal{K}} \pi_{mk} = 1, \forall m,k\}$ are the set constraints of offloading decisions, uplink transmission power, processing rates and association policies, respectively; Herein, $P_m^{\max}$ and $F_m^{\max}$ are the maximum power budget and processing rate of UE $m \in \mathcal{M}$, respectively. Constraints (12b) and (12c) are imposed to ensure that the overall e2e latency and energy consumption of UE m are limited by the predetermined thresholds $T_m^{\max}$ and $E_m^{\max}$, respectively. Constraint (12d) guarantees the minimum rate requirement $R_m^{\min}$ for all UEs. Finally, constraints (12e) and (12f) are added to ensure the queue stability at ESs and CS, respectively.

III. Proposed AO-based Algorithms for Solving Problem JCCO

A. Approximate Convex Problems

Let us start by rewriting the JCCO problem (12) equivalently as

$$\underset{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi},\mathbf{p},\mathbf{f},\boldsymbol{\tau}}{\text{minimize}} \max_{\forall m \in \mathcal{M}} \left\{ t_m(f_m^{\mathsf{lo}}, \boldsymbol{\tau})\right\} \quad (13a)$$

$$\text{s.t.} \quad (12c), (12d), (12e), (12f), (12g) \quad (13b)$$

$$t_m\left(f_m^{\mathsf{lo}}, \boldsymbol{\tau}\right) \le T_m^{\max}, \ \forall m \quad (13c)$$

$$\tau^{\mathsf{co}} \ge t_m^{\mathsf{co}}(\alpha_m, \mathbf{p}, \boldsymbol{\pi}_m), \ \forall m \quad (13d)$$

$$\tau^{\mathsf{es}} \ge t^{\mathsf{es}}(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}) \quad (13e)$$

$$\tau^{\mathsf{fh}} \ge t^{\mathsf{fh}}(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}) \quad (13f)$$

$$\tau^{\mathsf{cs}} \ge t^{\mathsf{cs}}(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}) \quad (13g)$$

where $t_m(f_m^{\mathsf{lo}}, \boldsymbol{\tau}) \triangleq \frac{(1-\alpha_m)C_m}{f_m^{\mathsf{lo}}} + \tau^{\mathsf{co}} + \tau^{\mathsf{es}} + \tau^{\mathsf{fh}} + \tau^{\mathsf{cs}}$, and $\boldsymbol{\tau} \triangleq \{\tau^{\mathsf{co}}, \tau^{\mathsf{es}}, \tau^{\mathsf{fh}}, \tau^{\mathsf{cs}}\}$ are newly introduced variables to simplify the objective function. Constraint (13c) is derived from (12b).

It is clear that the objective (13a) is a convex function in $(f_m^{\mathsf{lo}}, \boldsymbol{\tau})$. We also note that a direct application of IA method is still inapplicable due to strong coupling between variables. Considering the fact that the decision variables $(\mathbf{p}, \mathbf{f})$ and $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi})$ can be executed from UEs' and ESs' sides, respectively. Let us denote by $x^{(i)}$ the feasible point of x at the i-th iteration of the proposed iterative algorithm, which is a constant. By leveraging AO method, at iteration i, we decompose problem (13) into two subproblems (SPs) as follows:

$$\textsf{SP-1:} \underset{\mathbf{p},\mathbf{f},\boldsymbol{\tau} | \boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\pi}^{(i)}}{\text{minimize}} \max_{\forall m \in \mathcal{M}} \left\{ t_m(f_m^{\mathsf{lo}}, \boldsymbol{\tau})\right\} \quad (14a)$$

$$\text{s.t. } (12c), (12d), (13c), (13d) \quad (14b)$$

$$\mathbf{p} \in \mathscr{P}, \mathbf{f} \in \mathscr{F} \quad (14c)$$

and

$$\textsf{SP-2:} \underset{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi},\boldsymbol{\tau} | \mathbf{p}^{(i+1)}, \mathbf{f}^{(i+1)}}{\text{minimize}} \max_{\forall m \in \mathcal{M}} \left\{ t_m(f_m^{\mathsf{lo},(i+1)}, \boldsymbol{\tau})\right\} \quad (15a)$$

$$\text{s.t. } (12c) - (12f), (13c) - (13g) \quad (15b)$$

$$\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathscr{D}, \boldsymbol{\pi} \in \Pi. \quad (15c)$$

In an AO-based iterative algorithm, we first solve SP-1 for given $(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\pi}^{(i)})$ to generate the next optimal point of $(\mathbf{p}^{(i+1)}, \mathbf{f}^{(i+1)})$ and then solve SP-2 for updated value of $(\mathbf{p}^{(i+1)}, \mathbf{f}^{(i+1)})$ to generate the next feasible point $(\boldsymbol{\alpha}^{(i+1)}, \boldsymbol{\beta}^{(i+1)}, \boldsymbol{\pi}^{(i+1)})$. This procedure is repeated until convergence.

1) Approximate Convex Program for SP-1: In problem (14), non-convex parts include (12c), (12d) and (13d). Let us handle constraint (12d) first by convexifying $R_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})$ as

$$R_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) \ge \mathcal{R}_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) = \frac{(1-\omega)B}{\ln 2}\left[\mathcal{G}_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) - \kappa_m \mathcal{V}_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})\right], \quad (16)$$

which is fully presented in the Appendix; Here $\kappa_m = Q^{-1}(\epsilon_m)/\sqrt{(1-\omega)N}$ and $G_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) = \ln(1 + \bar{\gamma}_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}))$. As a result, constraint (12d) is iteratively replaced by the following convex constraint

$$\mathcal{R}_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) \ge \sum_{k \in \mathcal{K}} \pi_{mk}^{(i)} R_m^{\min}, \ \forall m \quad (17)$$

Next, we introduce new variables $\mathbf{r} \triangleq \{r_m\}_{\forall m}$ to express constraint (12c) equivalently as

$$\begin{cases} (1-\alpha_m^{(i)})\frac{\theta_m}{2} C_m (f_m^{\mathsf{lo}})^2 + \alpha_m^{(i)} D_m p_m r_m \le E_m^{\max}, & (18a) \\ \frac{1}{r_m} \le \mathcal{R}_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}), \forall m & (18b) \end{cases}$$

where constraint (18a) is non-convex due to the product of $p_m r_m$. We note that $p_m r_m$ is a concave function which can be innerly approximated for $x = p_m$, $y = r_m$, $\bar{x} = p_m^{(i)}$, $\bar{y} = r_m^{(i)}$, yielding

$$(1-\alpha_m^{(i)})\frac{\theta_m}{2} C_m (f_m^{\mathsf{lo}})^2 + \frac{1}{2}\alpha_m^{(i)} D_m \left(\frac{r_m^{(i)}}{p_m^{(i)}} p_m^2 + \frac{p_m^{(i)}}{r_m^{(i)}} r_m^2\right) \le E_m^{\max}, \forall m. \quad (19)$$

Lastly, by (18b), constraint (13d) is iteratively replaced by the following linear constraint

$$\tau^{\mathsf{co}} \geq \alpha_m^{(i)} D_m r_m, \ \forall m. \tag{20}$$

As a result, we obtain the following approximate convex program of SP-1 (14) solved at iteration i:

$$\textsf{SP-1 CVX:} \underset{\mathbf{p},\mathbf{f},\boldsymbol{\tau},\mathbf{r}|\boldsymbol{\alpha}^{(i)},\boldsymbol{\beta}^{(i)},\boldsymbol{\pi}^{(i)}}{\text{minimize}} \ \max_{\forall m\in\mathcal{M}} \left\{t_m(f_m^{\mathsf{lo}},\boldsymbol{\tau})\right\} \tag{21a}$$

$$\text{s.t. } (13c), (14c), (17), (18b), (19), (20). \tag{21b}$$

2) Approximate Convex Program for SP-2: For given $(\mathbf{p}^{(i+1)}, \mathbf{f}^{(i+1)})$ obtained by solving (21), we are now in position to convexify (15). To bypass the binary nature of (15), we first relax $\boldsymbol{\pi}$ to be continuous, i.e., $\boldsymbol{\pi} \in \tilde{\Pi} \triangleq \{\pi_{mk}, \forall m,k | 0 \leq \pi_{mk} \leq 1 \ \& \ \sum_{k\in\mathcal{K}} \pi_{mk} = 1, \forall m,k\}$ and rewrite it as

$$\textsf{SP-2:} \underset{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi},\boldsymbol{\tau}|\mathbf{p}^{(i+1)},\mathbf{f}^{(i+1)}}{\text{minimize}} \ \max_{\forall m\in\mathcal{M}} \left\{t_m(f_m^{\mathsf{lo},(i+1)},\boldsymbol{\tau})\right\} \tag{22a}$$

$$\text{s.t. } (12c)-(12f), (13c), -(13g) \tag{22b}$$

$$\boldsymbol{\alpha},\boldsymbol{\beta} \in \mathcal{D}, \boldsymbol{\pi} \in \tilde{\Pi}. \tag{22c}$$

Constraints (13c) and (22c) are linear while others are non-convex.

Convexity of (12d) *and* (12c)*:* We rewrite $\bar{\gamma}_m(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m) = \frac{\sum_{k\in\mathcal{K}} \pi_{mk}\sigma_{mk}^2}{\tilde{q}_m\left(\mathbf{p}^{(i+1)},\boldsymbol{\pi}_m\right)}$ where $\tilde{q}_m(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m)$ is defined as

$$\tilde{q}_m(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m) \triangleq \frac{\bar{\Phi}_m(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m)}{p_m^{(i+1)}(L-1)}.$$

By processing similarly as in $SP-1$, we have

$$R_m(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m) \geq \tilde{\mathcal{R}}_m^{(i)}(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m) = \frac{(1-\omega)B}{\ln 2} \times \left[\tilde{\mathcal{G}}_m^{(i)}(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m) - \kappa_m \tilde{\mathcal{V}}_m^{(i)}(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m)\right] \tag{23}$$

where $\tilde{\mathcal{G}}_m^{(i)}(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m)$ and $\tilde{\mathcal{V}}_m^{(i)}(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m)$ are conducted similarly as in approximate convex program of SP-1. As a result, we innerly approximate constraint (12d) as

$$\tilde{\mathcal{R}}_m^{(i)}(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m) \geq \sum_{k\in\mathcal{K}} \pi_{mk} R_m^{\min}, \ \forall m. \tag{24}$$

The constraint (12c) is equivalent to

$$\begin{cases} \dfrac{1}{\tilde{\mathcal{R}}_m^{(i)}\left(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m\right)} \leq r_m, & (25a) \\ (1-\alpha_m)\dfrac{\theta_m}{2} C_m (f_m^{\mathsf{lo}})^2 + p_m D_m \alpha_m r_m \leq E_m^{\max}, & (25b) \end{cases}$$

where $\mathbf{r} \triangleq \{r_m\}_{\forall m}$ were defined (18). We can approximate $\alpha_m r_m$ in (25b) as

$$(1-\alpha_m)\frac{\theta_m}{2} C_m (f_m^{(i+1)})^2 + \frac{1}{2} p_m^{(i+1)} D_m \Big(\frac{r_m^{(i+1)}}{\alpha_m^{(i)}}\alpha_m^2 + \frac{\alpha_m^{(i)}}{r_m^{(i+1)}} r_m^2\Big) \leq E_m^{\max}, \ \forall m. \tag{26}$$

Convexity of (12e) *and* (12f)*:* By introducing new variables $\check{\boldsymbol{\phi}} \triangleq \left\{\check{\phi}_{mk}\right\}_{\forall m,k}$, constraint (12e) is expressed as

$$\begin{cases} \sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}} \pi_{mk}\alpha_m \leq \dfrac{f_k^{\mathsf{es}}}{\sum_{m\in\mathcal{M}} C_m \check{\phi}_{mk}^2}, \ \forall k & (27a) \\ \check{\phi}_{mk}^2 \geq \pi_{mk}\alpha_m (1-\beta_{mk}), \ \forall m,k. & (27b) \end{cases}$$

We iteratively replace (27a) by

$$\sum_{m\in\mathcal{M}} \frac{1}{2}\lambda_m^{\mathsf{lo}} \left(\frac{\alpha_m^{(i)}}{\pi_{mk}^{(i)}}\pi_{mk}^2 + \frac{\pi_{mk}^{(i)}}{\alpha_m^{(i)}}\alpha_m^2\right) \leq f_k^{\mathsf{es}} \left(\frac{2}{\sum_{m\in\mathcal{M}} C_m (\check{\phi}_{mk}^{(i)})^2} - \frac{\sum_{m\in\mathcal{M}} C_m \check{\phi}_{mk}^2}{\left(\sum_{m\in\mathcal{M}} C_m (\check{\phi}_{mk}^{(i)})^2\right)^2}\right), \ \forall k \tag{28}$$

which is a convex constraint. To handle constraint (27b), we first rewrite as $\frac{\check{\phi}_{mk}^2}{1-\beta_{mk}} \geq \pi_{mk}\alpha_m$, and approximate both sides as

$$\frac{2\check{\phi}_{mk}^{(i)}\check{\phi}_{mk}}{1-\beta_{mk}^{(i)}} - \frac{(\check{\phi}_{mk}^{(i)})^2(1-\beta_{mk})}{(1-\beta_{mk}^{(i)})^2} \geq \frac{1}{2}\left(\frac{\alpha_m^{(i)}}{\pi_{mk}^{(i)}}\pi_{mk}^2 + \frac{\pi_{mk}^{(i)}}{\alpha_m^{(i)}}\alpha_m^2\right), \tag{29}$$

Similarly, by introducing new variables $\hat{\boldsymbol{\phi}} \triangleq \{\hat{\phi}_m\}_{\forall m}$, (12f) is equivalent to

$$\begin{cases} \sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}} \hat{\phi}_m^2 \leq \dfrac{f^{\mathsf{cs}}}{\sum_{m\in\mathcal{M}} C_m \hat{\phi}_m^2} & (30a) \\ \dfrac{\hat{\phi}_m^2}{\alpha_m} \geq \sum_{k\in\mathcal{K}} \pi_{mk}\beta_{mk}, \ \forall m & (30b) \end{cases}$$

which are approximated (detailed manipulations are omitted here due to the space limit) to yield

$$\sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}} \hat{\phi}_m^2 \leq f^{\mathsf{cs}} \left(\frac{2}{\sum_{m\in\mathcal{M}} C_m (\hat{\phi}_m^{(i)})^2} - \frac{\sum_{m\in\mathcal{M}} C_m \hat{\phi}_m^2}{\left(\sum_{m\in\mathcal{M}} C_m (\hat{\phi}_m^{(i)})^2\right)^2}\right) \tag{31}$$

$$\frac{2\hat{\phi}_m^{(i)}\hat{\phi}_m}{\alpha_m^{(i)}} - \frac{(\hat{\phi}_m^{(i)})^2\alpha_m}{(\alpha_m^{(i)})^2} \geq \sum_{k\in\mathcal{K}} \frac{1}{2}\Big(\frac{\beta_{mk}^{(i)}}{\pi_{mk}^{(i)}}\pi_{mk}^2 + \frac{\pi_{mk}^{(i)}}{\beta_{mk}^{(i)}}\beta_{mk}^2\Big), \tag{32}$$

Convexity of (13d)*:* From (25a), we rewrite (13d) as $\tau^{\mathsf{co}} \geq D_m \alpha_m r_m$ as

$$\tau^{\mathsf{co}} \geq \frac{1}{2} D_m \Big(\frac{r_m^{(i)}}{\alpha_m^{(i)}}\alpha_m^2 + \frac{\alpha_m^{(i)}}{r_m^{(i)}} r_m^2\Big), \ \forall m. \tag{33}$$

Convexity of (13e)*:* It follows from constraint (13e) that

$$\frac{f_k^{\mathsf{es}}}{\sum_{m\in\mathcal{M}} \pi_{mk}\alpha_m (1-\beta_{mk}) C_m} \geq \sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}} \pi_{mk}\alpha_m + \frac{1}{\tau^{\mathsf{es}}}, \ \forall k \tag{34}$$

which can be transformed equivalently as

$$\frac{f_k^{\mathsf{es}}}{\sum_{m\in\mathcal{M}} C_m \check{\phi}_{mk}^2} \geq \sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}} \pi_{mk}\alpha_m + \frac{1}{\tau^{\mathsf{es}}} \tag{35}$$

by (27b). Finally, we have

$$f_k^{\mathsf{es}} \left(\frac{2}{\sum_{m\in\mathcal{M}} C_m (\check{\phi}_{mk}^{(i)})^2} - \frac{\sum_{m\in\mathcal{M}} C_m \check{\phi}_{mk}^2}{\left(\sum_{m\in\mathcal{M}} C_m (\check{\phi}_{mk}^{(i)})^2\right)^2}\right) \geq \sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}} \frac{1}{2}\Big(\frac{\alpha_m^{(i)}}{\pi_{mk}^{(i)}}\pi_{mk}^2 + \frac{\pi_{mk}^{(i)}}{\alpha_m^{(i)}}\alpha_m^2\Big) + \frac{1}{\tau^{\mathsf{es}}}, \ \forall k. \tag{36}$$

Convexify of (13f)*:* We can express constraint (13f) as

$$\begin{cases} \tau^{\mathsf{fh}} \geq \sum\limits_{m\in\mathcal{M}} \varphi_{mk}^2 \dfrac{D_m}{R_k^{\mathsf{fh}}}, \ \forall k & \text{(37a)} \\ \dfrac{\varphi_{mk}^2}{\beta_{mk}} \geq \pi_{mk}\alpha_m, \ \forall m,k & \text{(37b)} \end{cases}$$

where $\boldsymbol{\varphi} \triangleq \{\varphi_{mk}\}_{\forall m,k}$ are new variables to tackle the product of $\pi_{mk}\alpha_m\beta_{mk}$. Constraint (37b) is non-convex. Similar to (27b), we have

$$\frac{2\varphi_{mk}^{(i)}\varphi_{mk}}{\beta_{mk}^{(i)}} - \frac{\varphi_{mk}^{2(i)}\beta_{mk}}{\left(\beta_{mk}^{(i)}\right)^2} \geq \frac{1}{2}\Big(\frac{\alpha_m^{(i)}}{\pi_{mk}^{(i)}}\pi_{mk}^2 + \frac{\pi_{mk}^{(i)}}{\alpha_m^{(i)}}\alpha_m^2\Big). \quad (38)$$

Convexity of (13g)*:* We can rewrite (13g) as

$$\frac{f^{\mathsf{cs}}}{\sum_{m\in\mathcal{M}} C_m\hat{\phi}_m^2} \geq \sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}}\hat{\phi}_m^2 + \frac{1}{\tau^{\mathsf{cs}}} \quad (39)$$

by using (30b). It follows from (31) that

$$f^{\mathsf{cs}}\left(\frac{2}{\sum\limits_{m\in\mathcal{M}} C_m(\hat{\phi}_m^{(i)})^2} - \frac{\sum\limits_{m\in\mathcal{M}} C_m\hat{\phi}_m^2}{\big(\sum\limits_{m\in\mathcal{M}} C_m(\hat{\phi}_m^{(i)})^2\big)^2}\right) \geq \sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}}\hat{\phi}_m^2 + \frac{1}{\tau^{\mathsf{cs}}}. \quad (40)$$

Summing up, we obtain the following approximate convex program of SP-2 solved at iteration i:

$$\textsf{SP-2: CVX} \quad \underset{\substack{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi},\boldsymbol{\tau},\check{\boldsymbol{\phi}},\hat{\boldsymbol{\phi}},\\ \mathbf{r},\boldsymbol{\varphi}|\mathbf{p}^{(i+1)},\mathbf{f}^{(i+1)}}}{\text{minimize}} \ \max_{\forall m\in\mathcal{M}} \left\{t_m(f_m^{\mathsf{lo},(i+1)},\boldsymbol{\tau})\right\} \quad (41a)$$

$$\text{s.t. (13c), (22c), (24), (25a), (26), (28),}$$
$$\text{(29), (31), (32), (33), (36), (37a), (38), (40).} \quad (41b)$$

which requires the per-iteration complexity of $\mathcal{O}\big(\sqrt{9M+4MK+3K}(3MK+3M+4)^2\big)$.

B. Proposed AO-IA based Algorithms

Let denote by $\mathcal{S}_1^{(i)} \triangleq (\mathbf{p}^{(i)}, \mathbf{f}^{(i)}, \mathbf{r}^{(i)})$ and $\mathcal{S}_2^{(i)} \triangleq (\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\pi}^{(i)}, \check{\boldsymbol{\phi}}^{(i)}, \hat{\boldsymbol{\phi}}^{(i)}, \mathbf{r}^{(i)}, \boldsymbol{\varphi}^{(i)})$ the feasible sets of (21) and (41) at iteration i, respectively. The overall algorithm for solving (13) is summarized in Algorithm 1.

Algorithm 1 Proposed AO-IA based Algorithm for Solving the JCCO Problem (13)

Initialization: Set $i = 0$ and randomly generate initial feasible points $\mathcal{S}_1^{(0)}$ and $\mathcal{S}_2^{(i)}$ to constraints in (21) and (41), respectively. Set the tolerance $\varepsilon = 10^{-3}$ and the maximum number of iterations $I^{\max}$.

1: **repeat**
2: Solve problem (21) for given $\mathcal{S}_2^{(i)}$ to obtain the optimal solution denoted by $(\mathbf{p}^\star, \mathbf{f}^\star, \mathbf{r}^\star, \boldsymbol{\tau}^\star)$ and update $\mathcal{S}_1^{(i+1)} := (\mathbf{p}^\star, \mathbf{f}^\star, \mathbf{r}^\star)$;
3: Solve problem (41) for given $\mathcal{S}_1^{(i+1)}$ to obtain the optimal solution denoted by $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \boldsymbol{\pi}^\star, \check{\boldsymbol{\phi}}^\star, \hat{\boldsymbol{\phi}}^\star, \mathbf{r}^\star, \boldsymbol{\varphi}^\star, \boldsymbol{\tau}^\star)$ and update $\mathcal{S}_2^{(i+1)} := (\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \boldsymbol{\pi}^\star, \check{\boldsymbol{\phi}}^\star, \hat{\boldsymbol{\phi}}^\star, \mathbf{r}^\star, \boldsymbol{\varphi}^\star)$;
4: Set $i := i + 1$;
5: **until** Convergence or $i > I^{\max}$
6: Recover binary values of $\boldsymbol{\pi}^\star$: $\pi_{mk}^\star = \lfloor \pi_{mk}^{(i)} + 0.5 \rfloor, \ \forall m,k$;
7: Repeat Steps 1-5 with fixed $\boldsymbol{\pi}^\star$ to refine the optimal solution;
8: **Output**: $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \boldsymbol{\pi}^\star, \mathbf{p}^\star, \mathbf{f}^\star)$.

The main drawback of solving problem (22) is that the exact binary solution of $\boldsymbol{\pi}$ is not guaranteed at optimum, resulting in an infeasible solution to the original problem (12). To overcome this issue, we consider Step 6 in Algorithm 1 using ceiling function to recover binary value of $\boldsymbol{\pi}$ as $\pi_{mk}^\star = \lfloor \pi_{mk}^{(i)} + 0.5 \rfloor, \ \forall m,k$. In Step 7, we repeat Steps 1-5 for given $\boldsymbol{\pi}^\star$ to minimize the performance loss due to Step 6.

IV. Numerical Results

We consider a small-cell scenario where all APs (ESs) and UEs are located within an area of 100×100 m [6]. ESs are located at (50, 33) and (50, 66) for $K = 2$ and $(50, 20)$, $(50, 40)$, $(50, 60)$, $(50, 80)$ for $K = 4$. The large-scale fading of the channel between UE m and AP k is modeled as $g_{mk} = 10^{\mathsf{PL}(d_{mk})/10}$, where $\mathsf{PL}(d_{mk}) = -35.3 - 37.6\log_{10} d_{mk}$ denotes the path loss in dB which is a function of the distance d_{mk} [9]. The number of antennas at each AP is set to $L = 8$. We assume that all UEs have the same power budget, i.e., $P_m^{\max} = 23$ dBm $\forall m$. The URLLC decoding error probability is set to $\epsilon_m = 10^{-9}, \forall m$.

We set the CPU cycles of ESs and CS to 25 and 30 Giga cycles/s, respectively. For UE m, the input data size and the required computation resource are set to $D_m = 100$ KB and $C_m = 800 \times 10^6$ (cycles), respectively. The total e2e latency requirement of each UE is given as $T_m^{\max} = 2$s $\forall m$. The arrival rate of tasks is set to $\lambda_m^{\mathsf{lo}} = 10$ (task/s), $\forall m$ [8]. Unless specifically stated otherwise, other parameters are given as follows: System bandwidth: 10 MHz; Noise power spectral density: -174 dBm/Hz; Maximum blocklength: 100; Fronthaul capacity: 1 Gbps; Maximum energy consumption: 1 Joule; Effective capacitance coefficient: 10^{-27} Watt.s^3/cycle^3.

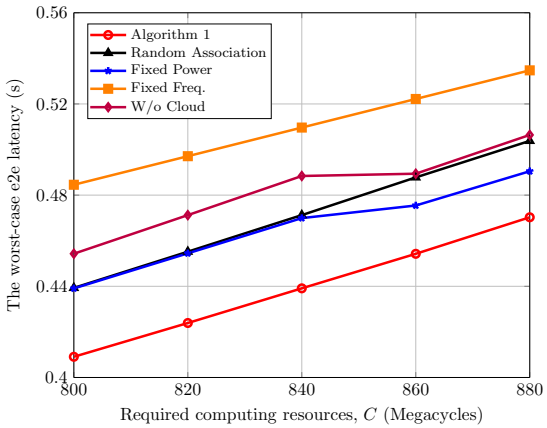

Fig. 1: The worst-case e2e latency of different resource allocation schemes versus the required computation resource $C \equiv C_m, \forall m$, with $M = 10$ UEs and $K = 2$ ESs.

Impact of required computation resource: Fig. 1 illustrates the e2e latency versus the required computation resource $C_m, \forall m$. Unsurprisingly, the e2e latency of all the considered schemes increases when C_m increases. We recall that a higher value of C_m will not only increase processing latency at UEs,

ESs and CS but also force UEs to scale down their processing rate and transmit power to satisfy constraint (12c), resulting in higher total latency. Again, Algorithm 1 still provides the lowest e2e latency amongst all the considered schemes.

Impact of offloading factor and processing rate: In hierarchical edge-cloud systems, the processing rates of ESs and CS have a strong impact on the system performance. In particular, the higher the processing rate, the higher the offloading portion from UEs to ESs. To verify this, we first show the e2e latency and average offloading portion from UEs to ESs versus ESs' processing rate $f^{\text{es}} \equiv f_k^{\text{es}}, \forall k$ in Fig. 2. The offloading portion of UEs slightly increases when ESs have larger computation resource, f^{es}. The e2e latency of Algorithm 1 sharply decreases by approximately 200 ms when the processing rate of ESs increases to 30 gigacycles/s. An important observation is that Algorithm 1 always offloads the higher portion of computation tasks, compared to sub-optimal schemes, thanks to optimal user association policies, leading to lower e2e latency.

V. Conclusion

We investigated the joint communication and computation task offloading in URLLC-based hierarchical edge-cloud systems. To address the practical issues of minimizing the worst-case e2e latency, we proposed an alternating optimization framework to efficiently solve the formulated problem in an iterative manner. Finally, we provided extensive numerical results to demonstrate the significant performance gain achieved by joint optimization of the communication and computation variables in hierarchical edge-cloud systems.

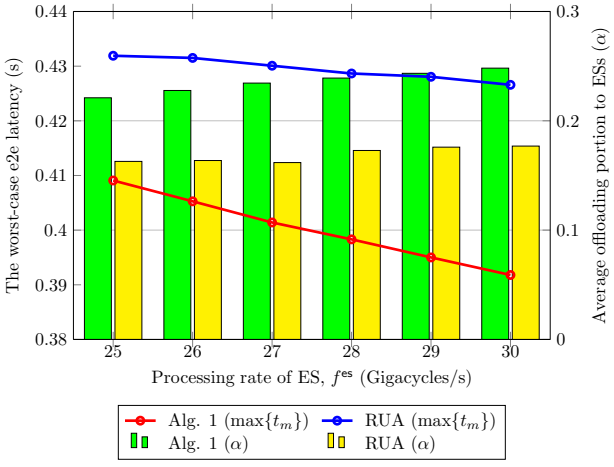

Fig. 2: The worst-case e2e latency and average offloading portion from UEs to ESs versus ESs' processing rate $f^{\text{es}} \equiv f_k^{\text{es}}, \forall k$, with $M = 10$ UEs and $K = 2$ ESs.

Appendix

We first rewrite the SINR of UE m as $\gamma_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) = p_m / q_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})$. By applying [9, Eq. (72)] for $x = p_m$, $y = q_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})$, $\bar{x} = p_m^{(i)}$, and $\bar{y} = q_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)})$, we have

$$G_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) \geq a_m^{(i)} - \frac{b_m^{(i)}}{p_m} - c_m^{(i)} q_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) \triangleq \mathcal{G}_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})$$

where $a_m^{(i)} = \ln\big(1 + \frac{p_m^{(i)}}{q_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})}\big) + 2\frac{p_m^{(i)}}{p_m^{(i)} + q_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})}$, $b_m^{(i)} = \frac{(p_m^{(i)})^2}{p_m^{(i)} + q_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})}$ and $c_m^{(i)} = \frac{p_m^{(i)}}{\big(q_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) + p_m^{(i)}\big) q_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})}$.

To find an upper bounding convex function approximation of $V_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})$, we apply [9, Eq. (75)] for $x = 1 - 1/\big(1 + \bar{\gamma}_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})\big)^2$ and $\bar{x} = 1 - 1/\big(1 + \bar{\gamma}_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)})\big)^2$, yielding

$$V_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) \leq d_m^{(i)} - e_m^{(i)} \frac{q_m^2(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})}{\big(q_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) + p_m\big)^2} \tag{42}$$

where

$$d_m^{(i)} = 0.5\sqrt{V_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)})} + 0.5/\sqrt{V_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)})} \tag{43}$$

$$e_m^{(i)} = 0.5/\sqrt{V_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)})}. \tag{44}$$

The function $\frac{q_m^2(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})}{\big(q_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) + p_m\big)^2}$ in (42) is still not convex [9], which can be further approximated as

$$\frac{q_m^2(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})}{q_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) + p_m} \frac{1}{q_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) + p_m} \geq \frac{2}{q_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)}) + p_m^{(i)}} \left(\frac{2q_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)}) q_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})}{q_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)}) + p_m^{(i)}} - \frac{q_m^2(p_m^{(i)})}{(q_m(p_m^{(i)}) + p_m^{(i)})^2} \right.$$

$$\left. \big(q_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) + p_m\big) \right) - \frac{q_m^2(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})}{(q_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) + p_m^{(i)})^2}$$

By substituting this result to (42), yields

$$V_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) \leq \mathcal{V}_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) \triangleq d_m^{(i)} - \frac{2e_m^{(i)}}{q_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)}) + p_m^{(i)}}$$

$$\times \Big(2f_m^{(i)} q_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) - (f_m^{(i)})^2 \big(q_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) + p_m\big) \Big)$$

$$+ \frac{(f_m^{(i)})^2}{q_m^2(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)})} q_m^2(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) \text{where } f_m^{(i)} \triangleq \frac{q_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)})}{q_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)}) + p_m^{(i)}}.$$

References

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

[2] Y. Xiao and M. Krunz, "Distributed optimization for energy-efficient fog computing in the tactile Internet," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 11, pp. 2390–2400, Nov. 2018.

[3] C. She, C. Yang, and T. Q. S. Quek, "Joint uplink and downlink resource configuration for ultra-reliable and low-latency communications," *IEEE Trans. Commun.*, vol. 66, no. 5, pp. 2266–2280, 2018.

[4] C. Sun, *et al.*, "Optimizing resource allocation in the short blocklength regime for ultra-reliable and low-latency communications," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 402–415, 2019.

[5] Z. Zhou, Z. Wang, H. Yu, H. Liao, S. Mumtaz, L. Oliveira, and V. Frascolla, "Learning-based URLLC-aware task offloading for Internet of health things," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 2, pp. 396–410, Feb. 2021.

[6] C.-F. Liu, *et al.*, "Dynamic task offloading and resource allocation for ultra-reliable low-latency edge computing," *IEEE Trans. Commun.*, vol. 67, no. 6, pp. 4132–4150, 2019.

[7] H. Ren *et al.*, "Joint pilot and payload power allocation for massive-MIMO-enabled URLLC IIoT networks," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 5, pp. 816–830, 2020.

[8] J. Tang, W. P. Tay, T. Q. S. Quek, and B. Liang, "System cost minimization in cloud ran with limited fronthaul capacity," *IEEE Trans. Wireless Commun.*, vol. 16, no. 5, pp. 3371–3384, May 2017.

[9] A. A. Nasir *et al.*, "Resource allocation and beamforming design in the short blocklength regime for URLLC," *IEEE Trans. Wireless Commun.*, vol. 20, no. 2, pp. 1321–1335, 2021.