Computation Offloading in MIMO Based Mobile Edge Computing Systems Under Perfect and Imperfect CSI Estimation

Ti Ti Nguyen, *Student Member, IEEE*, Long Bao Le, *Senior Member, IEEE*, and Quan Le-Trung

Abstract—Intelligent offloading of computation-intensive tasks to a mobile cloud server provides an effective mean to expand the usability of wireless devices and prolong their battery life, especially for low-cost internet-of-things (IoT) devices. However, realization of this technology in multiple-input multiple-output (MIMO) systems requires sophisticated design of joint computation offloading and other network functions such as channel state information (CSI) estimation, beamforming, and resource allocation. In this paper, we study the computation task offloading and resource allocation optimization in MIMO based mobile edge computing systems considering perfect/imperfect-CSI estimation. Our design aims to minimize the maximum weighted energy consumption subject to practical constraints on available computing and radio resources and allowable latency. The optimal and low-complexity algorithms are proposed to solve the underlying mixed integer non-linear problems (MINLP). For the perfect-CSI, we employ bisection search to find the optimal solution. The low-complexity algorithms are developed by decomposing the original optimization problem into the offloading optimization (OP) and power allocation (PA) subproblems and solve them iteratively. Moreover, the difference of convex functions (DC) method is employed to deal with non-convex structure of (PA) subproblems in the imperfect-CSI scenario. Numerical results confirm the advantages of proposed designs over conventional local computation strategies in energy saving and fairness.

Index Terms—Mobile edge-cloud computing, resource allocation, power allocation, computation offloading, MIMO

1 Introduction

The number of novel and sophisticated wireless applications has increased drastically in recent years such as object recognition, social network, e-health, natural language processing, and virtual reality gaming thanks to the appearance of artificial intelligence-based services and high-speed communications [1]. It is expected that these desktop-level applications can be run on mobile platforms equipped with powerful processors [2]. However, this is challenging due to limitation of energy and computation capacity on mobile devices, especially for low-cost IoT devices. In fact, deployment of higher clock frequency central processing units (CPU) to process computation-intensive applications leads to the increase in the energy consumption [3]. Unfortunately, the advancement in mobile battery technology can improve the energy density only sixfold since the 1900s [4] while the computing capacity of the mobile chipset increases exponentially following Moorse's law. This means that the improvement in the battery capacity is not sufficiently fast to keep up with the practical applications' requirements.

Thus, the battery can become the bottleneck to realize many emerging mobile services. One potential solution for enhancing the mobile usability and extending the mobile battery life is to offload the computation-intensive tasks from mobile users to the central cloud or edge-cloud using the so-called mobile cloud computing (MCC) or mobile edge computing (MEC) technologies [5]. The computation offloading in both paradigms is quite similar and the difference between them is related to the available computing resources and the relative distance from the cloud to mobile users. For the MCC, the enormous computing resource is provided in the core network, whereas the MEC system is equipped with more limited computing resource at the network edges. Moreover, the MCC architecture usually has limited backhaul capacity, thus it may not be suitable for applications with strict latency constraints. The MEC can address this challenge but the limited computing resource at the edge-cloud must be carefully allocated to efficiently support the computation offloading services.

1.1 Related Works

Literature survey of recent computation offloading designs can be found in [6]. A mobile cloud middleware is proposed in [7] to enable the interoperability across multiple clouds, asynchronous delegation of mobile tasks, and dynamic allocation of the cloud resource. Moreover, the MAUI platform in [8] enables fine-grained code offload by automatically extracting the program state needed in the offloading process and creating the replicated version of the user's application execution file in the cloud. The offloading decision

- *T. T. Nguyen and L.B. Le are with the INRS, University of Quebec, Montréal, Québec H5A 1K6, Canada. E-mail: {titi.nguyen, long.le}@emt.inrs.ca.*
- *Q. Le-Trung is with the University of Information Technology, Vietnam National University, Ho Chi Minh, Vietnam. E-mail: quanlt@uit.edu.vn.*

Manuscript received 11 Apr. 2018; revised 26 Nov. 2018; accepted 6 Jan. 2019. Date of publication 13 Jan. 2019; date of current version 9 Dec. 2021. (Corresponding author: Nguyen Ti Ti.)
Digital Object Identifier no. 10.1109/TSC.2019.2892428

optimization is recently studied in [9], [10]. In these existing works, the offloading decision is not jointly optimized with constrained wireless resource allocation. However, this is very important for the setting where multiple users request services at the same time, which may create strong interference if not be well managed and low transmission rates, which may lead to failure of the offloading process.

In order to guarantee the service latency requirements and successful transmission of the involved data to the cloud in an energy-efficient manner, it is important to jointly optimize the allocation of computation and radio resources [11]. Along this line, Zhang et al. in [12] consider probabilistic computation offloading and study an optimal strategy to minimize the energy consumption considering data transmission over a Gilbert-Elliott channel. In [13], the authors formulate the link selection and data transmission scheduling as a discrete-time stochastic dynamic program to optimize the energy efficiency for the single-task single-user system. Computation offloading for the multi-task single-user system is considered in [14], [15]. In particular, the task offloading is dynamically decided by using the Lyapunov optimization method in [14] or by solving a constrained shortest path problem in [15].

For the single-task multi-user system, the computation offloading and resource allocation optimization becomes more complicated [16], [17], [18], [19], [20]. Partial computation offloading to minimize the weighted sum of energy consumption is studied in [16], where CPU partitioning decision variables are assumed to be continuous. In [17], [18], [21], the authors consider the single-task offloading and resource allocation for the interference-free transmission scenario. In [17], the computing and communication resources are shared among users with the control assistance of the network operator. This proposed framework exploits available network resources with small communication delay to achieve high energy-efficiency. In [18], the authors consider the mixed fog/cloud architecture to leverage the computing resources both at the fog and the cloud while the work [21] employs the Lyapunov-based optimization technique to minimize the average weighted sum power consumption where the computation tasks of mobile users are assumed to be fine-grained.

Moreover, iterative algorithms are developed in [19], [20] to improve energy-efficiency considering radio interference. Specifically, the decomposition framework and heuristic algorithm are respectively proposed in [19] and [20] to deal with the complicated intra- and inter-cell interference. However, making offloading decisions based on the priority in [20] may not guarantee fairness among mobile users. Some recent works consider the integration of the massive MIMO technology into the MCC/MEC system [22], [23], [24]. In particular, they tackle the joint backhaul, computing and radio resource allocation problem for a single-task offloading problem which aims to minimize the total energy consumption. These designs, however, do not consider the CSI estimation performance, which is a very important factor directly affecting the wireless communication quality and transmission rate of the underlying MIMO based wireless system [25].

While the above papers consider three different MEC systems, namely the single-task single-user system, single-task multi-user system, and multi-task single-user system, the more complex multi-task multi-user system has been studied in some recent works [26], [27], [28]. Specifically, [26] proposes a heuristic algorithm to tackle the offloading and task scheduling problems for multicore-based mobile devices where data transmission is not optimized. Assuming the interference-free wireless network, binary task offloading design is conducted in [27] by employing semi-definite relaxation and the probability based rounding technique. However, both transmit energy and spectral efficiency are assumed to be unchanged in optimization of the allocated bandwidth in this work, which may not hold true in practice. This optimization framework is further extended in [28] to consider co-channel interference in the heterogeneous network based MEC system; however, independent power allocation for different users adopted by this work may not be efficient in managing the interference. Even though there have been some efforts in tackling the computation offloading design for the multi-task multi-user MEC system, consideration of advanced communication aspects such as MIMO communications for such design requires much more further research.

1.2 Contributions and Organization of the Paper

Existing computation offloading designs for the multi-task multi-user scenario have not considered the important MIMO communication technology and its related issues such as the imperfect CSI estimation. Our current paper aims to fill this gap in the literature by proposing general offloading and resource allocation algorithms which can provide fairness and consider the cutting-edge MIMO technology. In particular, the main contributions of this paper can be summarized as follows:

- We formulate the joint computation offloading and resource allocation problem that minimizes the maximum weighted consumed energy (Min-max W.C.E) for mobile users considering the latency and resource limitation constraints. The problem formulation captures the general partial offloading for the multi-task multi-user setting where the computation tasks can be either processed locally at the mobile user or offloaded and processed in the cloud. We consider two important scenarios with perfect-CSI (P-CSI) and imperfect-CSI (IP-CSI) estimation for the MIMO-based MEC system. To the best of our knowledge, the study of P-CSI and IP-CSI for MIMO communications in the MEC system has not been conducted in the literature.
- We propose different efficient algorithms to solve the underlying MINLP problems. For the P-CSI scenario, we propose an optimal algorithm achieving the global optimal solution by employing the bisection search method in which the optimization problem is decomposed into independent convex subproblems for individual users. We also propose a low-complexity algorithm which iteratively solves the offloading subproblem (OP) and power allocation subproblem (PA) until convergence. For the IP-CSI setting, the decomposition of the original problem into the (OP) and (PA) subproblems is also

TABLE 1
Important Notations

Notations	Description
$K/\mathcal{K}$	Number/set of users
$\mathcal{K}_1$	Set of users having at least 1 offloaded task
$\mathcal{L}_k$	Set of tasks of user k
W	Transmission bandwidth (Hz)
M	Number of antennas at BS
$\boldsymbol{H}/\hat{\boldsymbol{H}}$	Perfect/imperfect estimated channel matrix
$\boldsymbol{A}/\hat{\boldsymbol{A}}$	Beamforming matrix with P/IP- CSI
$c_{k,l_k}/b_{k,l_k}$	Number of CPU cycles/upload bits of task l_k of UE k
b^{dl}_{k,l_k}	Number of download bits of task l_k of UE k (bits)
Γ_{dpu}	Ratio of download bits per upload bits
s_{k,l_k}	Offloading decision of task l_k of UE k
f^{c}_k	Allocated CPU clock speed for UE k from cloud (Hz)
F^{c}	Maximum CPU clock speed of cloud (Hz)
f_k/F_k	CPU clock speed of UE k and its maximum (Hz)
$p_k/p_{k,c}$	Uplink transmit power/circuit power of UE k (Watts)
p^{dl}_k	Downlink transmit power allocated for UE k (Watts)
$P_k/P^{\mathsf{dl}}_{\mathsf{max}}$	Maximum transmit power of UE k/BS (Watts)
ξ	Min-max weighted consumed energy
$\xi^{\mathsf{lo}}_k/\xi^{\mathsf{t}}_k$	Computation/transmit energy of UE k (Joule)
$t^{\mathsf{lo}}_k/t^{\mathsf{t}}_k$	Computation/transmit time of UE k (second)
α_k	Computation energy coefficient of UE k
β_k	Large-scale fading coefficient of channel $\boldsymbol{h}_k$
β^{a}_k	Defined in (13)
η_k	Maximum latency requirement of UE k (second)
σ_{bs}	Noise power received at BS (Watts)
σ^{dl}_k	Noise power received at UE k (Watts)
T/τ	Channel coherence interval / pilot sequence length
$r^{\mathsf{lb}}_k/\hat{r}^{\mathsf{lb}}_k$	Lower-bound of uplink rate with P/IP-CSI (bits/s)
$\hat{r}^{\mathsf{dl,lb}}_k$	Lower-bound of downlink rate (bits/s)
λ_k/σ_k	Coefficients defined in (16)
$b^{\mathsf{a}}_k/c^{\mathsf{a}}_k$	Offloaded bits/ CPU cycles of UE k
ξ^{a}_k	Relation of transmission bits, energy defined in $(\mathcal{P}_3)'_k$
$\mathbb{E}(x)$	Expected value of x

performed and the DC optimization approach is then employed to convexify and tackle the non-convex constraints in the (PA) subproblem.

- We prove the convergence of different proposed iterative algorithms. Moreover, we discuss the extension of the proposed design to consider both uplink data transmission and downlink feedback of the computation outcome in the computation offloading design. We show how our proposed algorithm can be extended to address this more general problem. Moreover, we analyze the complexity of the proposed algorithms.
- Numerical studies show that the low-complexity algorithm achieves close-to-optimal performance in the P-CSI scenario. Moreover, we show that our proposed design can achieve good fairness for different users. Finally, we investigate the impacts of different parameters including the maximum allowable delay, the energy coefficient of mobile devices, the number of computation tasks to the achievable performance.

The remaining of this paper is organized as follows. Section 2 presents the system model and the task scheduling and computation-resource allocation problems. Section 3 describes the joint radio resource and computing resource algorithms for the P-CSI scenario. Section 4 discusses the algorithm design for the IP-CSI scenario. Section 5 presents the extension of our design for applications that require downlink transmission of the computation results and analyzes the complexity of the proposed algorithms. Section 6 evaluates the performance of the proposed algorithms and Section 7 concludes the work. In Table 1, we summarize important notations in this paper.

2 System Model and Problem Formulation

Considering an MEC system comprising K single-antenna users or user equipments (UEs) and one base station (BS) equipped with M antennas. For convenience, we denote the set of UEs as $\mathcal{K} = \{1, 2, \ldots, K\}$. We assume that $M > K$, the cloud server is located at the edge of the cellular network and the high-speed fiber link is used to connect the network operator and the cloud server. Then, the cloud can serve offloaded computation demands from multiple UEs simultaneously. It is assumed that the cloud has received in advance the UEs' task images (i.e., the replicated versions of the execution files corresponding to the offloading tasks), which can, therefore, be executed in the cloud in the offloading case [3], [19].

2.1 Computation Offloading Model

We assume that UE k has the set of $\mathcal{L}_k$ independent computation tasks from his/her application and these tasks can be executed locally at the mobile device or offloaded and executed in the cloud independently with the maximum allowable delay η_k [6], [29]. For example, the action recognition application for videos can be decomposed into two main tasks, the first one for capturing the spacial information and the second one for analyzing the temporal information [30]. Moreover, each task $l_k \in \mathcal{L}_k$ has the number of required CPU cycles c_{k,l_k} and the number of data bits b_{k,l_k} (e.g., to transmit the involved programming states to the BS). For instance, according to [8] the number of bits and CPU cycles of the task of detecting and extracting faces are about 0.26 Gcycles and 14 kbits, respectively.

We now introduce a binary offloading decision variable s_{k,l_k} for task $l_k \in \mathcal{L}_k$ as follows:

$$s_{k,l_k} = \begin{cases} 1, & \text{if task } l_k \text{ is executed at the mobile device} \\ 0, & \text{if task } l_k \text{ is offloaded to the cloud.} \end{cases} \quad (1)$$

The local computation energy and time due to user k can be expressed, respectively, as

$$\xi^{\mathsf{lo}}_k = \alpha_k f_k^2 \sum_{l_k \in \mathcal{L}_k} s_{k,l_k} c_{k,l_k}, \quad (2)$$

$$t^{\mathsf{lo}}_k = f_k^{-1} \sum_{l_k \in \mathcal{L}_k} s_{k,l_k} c_{k,l_k}, \quad (3)$$

where α_k denotes the energy coefficient specified in the CPU model and f_k denotes the CPU clock speed of UE k (CPU cycles/second or Hz), which is assumed to be smaller than the UE's maximum clock speed F_k. Each individual UE k can partially or totally offload its computation tasks to the cloud if the underlying program can be more efficiently executed in the cloud or requires more computing resource at the mobile device to execute within the delay η_k.

2.2 Cloud Computation Model

Upon receiving the offloading demand from UE k, the cloud server will assign the computing resource measured in the

CPU clock speed f_k^{c} to execute the UE's application. The required execution time to finish the computation demand from UE k in the cloud server can be expressed as

$$t_k^{\mathsf{c}} = (f_k^{\mathsf{c}})^{-1} \sum_{l_k \in \mathcal{L}_k} (1 - s_{k,l_k}) c_{k,l_k}. \tag{4}$$

In addition, we assume that the total computing resources allocated by the cloud server must be within its available computing budget F^{c}. This constraint can be expressed as $\sum_{k \in \mathcal{K}_1} f_k^{\mathsf{c}} \leq F^{\mathsf{c}}$, where $\mathcal{K}_1 = \{k \in \mathcal{K} | \sum_{l \in \mathcal{L}_k} s_{k,l_k} < |\mathcal{L}_k|\}$, denotes the set of UEs that cannot execute all their tasks locally.

2.3 Wireless Transmission Model

Let $\boldsymbol{h}_k \in \mathbb{C}^{M \times 1}$ denote the uplink channel gain vector between UE k and the BS's antennas where elements of the uplink channel gain vector $\boldsymbol{h}_k$ are modeled as $h_{mk} = \varphi_{mk}\sqrt{\beta_k}, m \in \{1, 2, \ldots, M\}$, where φ_{mk} and β_k represent the small-scale and large-scale fading coefficients, respectively. Assuming that the large-scale channel coefficient does not change during the required application execution latency (i.e., slow fading) and the small-scale channel coefficients $\varphi_{mk}, \forall m$ are independently and identically distributed (i.i.d.) $\mathcal{CN}(0, 1)$ random variables. Let p_k denote the uplink transmit power of UE k and $\boldsymbol{n}$ denote the noise vector whose components are i.i.d. $\mathcal{CN}(0, \sigma_{bs})$ variables.

We derive the transmission rate, time, and energy for P-CSI and IP-CSI scenarios and ZF beamforming in the following.

2.3.1 P-CSI Scenario

The received baseband signal at the BS after passing through a linear detector is

$$y_k = \sqrt{p_k}\boldsymbol{a}_k^H \boldsymbol{h}_k x_k + \sum_{i \neq k, i \in \mathcal{K}_1} \sqrt{p_i}\boldsymbol{a}_k^H \boldsymbol{h}_i x_i + \boldsymbol{a}_k^H \boldsymbol{n}, \tag{5}$$

where x_k represents the transmitted symbol from UE k, which satisfies $\mathbb{E}(|x_k|^2) = 1$, and $\boldsymbol{a}_k$ denotes the beamforming vector of UE k.

The uplink rate achieved by UE k is given by

$$r_k = W \log_2(1 + \gamma_k), \tag{6}$$

where W is the communication bandwidth and γ_k is the signal-to-noise-plus-interference ratio (SINR) of UE k, which can be written as

$$\gamma_k = \frac{p_k |\boldsymbol{a}_k^H \boldsymbol{h}_k|^2}{\sum_{i \neq k, i \in \mathcal{K}_1} p_i |\boldsymbol{a}_k^H \boldsymbol{h}_i|^2 + \sigma_{bs} ||\boldsymbol{a}_k||^2}. \tag{7}$$

Then, the average energy required for data transmission of UE k can be computed as

$$\begin{aligned} \bar{\xi}_k^{\mathsf{t}} &= \mathbb{E}\big((p_k + p_{k,c}) t_k\big) \\ &= (p_k + p_{k,c}) \sum_{l_k \in \mathcal{L}_k} (1 - s_{k,l_k}) b_{k,l_k} \mathbb{E}\left(\frac{1}{r_k}\right), \end{aligned} \tag{8}$$

where $p_{k,c}$ denotes the circuit power. The exact expression for $\mathbb{E}(1/r_k)$ is quite complicated for analysis; therefore, we introduce the upper bound of this term as follows. Toward this end, we have following result for function $\psi(x) = 1/\log(1 + x^{-1})$:

$$\nabla^2\big(\psi(x)\big) = \frac{2 - (2x+1)\log\big(1 + \frac{1}{x}\big)}{\log^3\big(1 + \frac{1}{x}\big)(x^2 + x)^2} = \begin{cases} \ll 0, & \text{if } x \approx 0 \\ \approx 0, & \text{if } x \gg 0. \end{cases} \tag{9}$$

When UE k decides to offload its computation tasks to the cloud for energy saving, intuitively its wireless transmission condition in terms of SINR must be sufficiently good so that one can maintain the required application execution latency. Therefore, a tight approximation for the upper bound of the ergodic transmission time and energy can be obtained by applying the Jensen's inequality for a concave function as follows:

$$\mathbb{E}\left(\frac{1}{r_k}\right) \leq \left[W \log_2\left(1 + \frac{1}{\mathbb{E}\left(\gamma_k^{-1}\right)}\right)\right]^{-1} \stackrel{\text{def}}{=} \frac{1}{r_k^{\mathsf{lb}}}. \tag{10}$$

Then, the upper bound of the average transmission time and energy can be written, respectively, as

$$t_{k,\mathsf{P}}^{\mathsf{t,ub}} = (r_k^{\mathsf{lb}})^{-1} \sum_{l_k \in \mathcal{L}_k} (1 - s_{k,l_k}) b_{k,l_k}, \tag{11}$$

$$\xi_{k,\mathsf{P}}^{\mathsf{t,ub}} = (p_k + p_{k,c}) t_{k,\mathsf{P}}^{\mathsf{t,ub}}. \tag{12}$$

Assuming that the zero-forcing (ZF) beamforming method, which is a popular and widely accepted technique for MIMO communication, is employed to recover the user's signal at the receiver. Then, under the P-CSI scenario, the linear detector matrix corresponding to the ZF beamforming method can be written as $\boldsymbol{A} = \boldsymbol{H}(\boldsymbol{H}^H \boldsymbol{H})^{-1}$ where $\boldsymbol{H}$ is a channel matrix whose columns are channel vectors of the UEs and $\boldsymbol{A}$ is the beamforming matrix whose columns are beamforming vectors of the UEs. Then, the lower bound on the achievable rate with P-CSI is given by [31]

$$r_k^{\mathsf{lb}} = W \log_2\left(1 + \frac{p_k/\sigma_{bs}}{\mathbb{E}\{[(\boldsymbol{H}^H \boldsymbol{H})^{-1}]_{kk}\}}\right) = W \log_2(1 + p_k \beta_k^{\mathsf{a}}), \tag{13}$$

where $\beta_k^{\mathsf{a}} = \frac{(M - |\mathcal{K}_1|)\beta_k}{\sigma_{bs}}$.

2.3.2 IP-CSI Scenario

Assuming that each UE employs a pilot signal to estimate its CSI once in each channel coherence interval. Let T denote the number of symbol periods corresponding to the channel coherence interval and let τ denote the number of symbols in the pilot. Let $\sqrt{\tau p^{\mathsf{tr}}}\boldsymbol{\phi}_k \in \mathbb{C}^{\tau \times 1}$ be the pilot sequence assigned for UE k where p^{tr} denotes the pilot power and $||\boldsymbol{\phi}_k||^2 = 1$. Assuming that $\tau \geq |\mathcal{K}_1|$ and the pilot sequences are designed pair-wise orthogonally, i.e., $\boldsymbol{\phi}_k^H \boldsymbol{\phi}_j = 0, \forall k \neq j$. Suppose that we employ the minimum mean square error (MMSE) CSI estimation approach [31], the covariance of the estimated CSI $\hat{h}_{m,k}$ is $\mathbb{E}(|\hat{h}_{mk}|^2) = \frac{\tau p^{\mathsf{tr}} \beta_k^2}{\tau p^{\mathsf{tr}} \beta_k + \sigma_{bs}}$.

Let $\boldsymbol{\epsilon} = \hat{\boldsymbol{H}} - \boldsymbol{H}$ be the difference between the estimated channel matrix $\hat{\boldsymbol{H}}$ and true channel matrix $\boldsymbol{H}$. Following [31], its element can be modeled as $\epsilon_{ik} \sim \mathcal{CN}(0, \frac{\sigma_{bs}\beta_k}{\tau p^{\mathsf{tr}} \beta_k + \sigma_{bs}})$. The received signal associated with the UE k after passing

through the ZF-based detector $\hat{\boldsymbol{A}} = \hat{\boldsymbol{H}}(\hat{\boldsymbol{H}}^H\hat{\boldsymbol{H}})^{-1}$ in this scenario can be re-written as

$$y_k = \sqrt{p}_k\hat{\boldsymbol{a}}_k^H\hat{\boldsymbol{h}}_k x_k + \sum_{i\neq k, i\in\mathcal{K}_1}\sqrt{p_i}\hat{\boldsymbol{a}}_k^H\hat{\boldsymbol{h}}_i x_i - \sum_{i\in\mathcal{K}_1}\sqrt{p_i}\hat{\boldsymbol{a}}_k^H\hat{\boldsymbol{\epsilon}}_i x_i + \hat{\boldsymbol{a}}_k^H\boldsymbol{n}. \tag{14}$$

The SINR for this IP-CSI scenario by treating the estimated channel as the true channel is expressed as [31]

$$\hat{\gamma}_k = \frac{p_k|\hat{\boldsymbol{h}}_k^H\hat{\boldsymbol{a}}_k|^2}{\sum_{i\neq k, i\in\mathcal{K}_1} p_i|\hat{\boldsymbol{h}}_i^H\hat{\boldsymbol{a}}_k|^2 + \sum_{i\in\mathcal{K}_1} p_i|\hat{\boldsymbol{a}}_k^H\epsilon_i|^2 + \sigma_{bs}||\hat{\boldsymbol{a}}_k||^2}. \tag{15}$$

Then, the ergodic rate achieved by UE k can be computed as $\hat{r}_k = \mathbb{E}(W\log_2(1+\hat{\gamma}_k))$. With ZF based detection, the lower bound of rate $\hat{r}_k^{\mathsf{lb}}$ using the Jensen's inequality can be written as follows [31]:

$$\begin{aligned}\hat{r}_k^{\mathsf{lb}} &= W\log_2\left(1+\frac{1}{\mathbb{E}(\hat{\gamma}_k^{-1})}\right)\\ &= W\log_2\left(1+\frac{p_k}{\sum_{i\in\mathcal{K}_1}p_i\lambda_{k,i}+\sigma_k}\right)\\ &= W\log_2\left(\boldsymbol{p}^{\mathsf{t}}\boldsymbol{\lambda}_k+\sigma_k+p_k\right) - W\log_2\left(\boldsymbol{p}^{\mathsf{t}}\boldsymbol{\lambda}_k+\sigma_k\right),\end{aligned} \tag{16}$$

where $\lambda_{k,i} = \frac{(\tau p^{\mathsf{tr}}\beta_k+\sigma_{bs})\sigma_{bs}\beta_i}{(\tau p^{\mathsf{tr}}\beta_i+\sigma_{bs})\tau p^{\mathsf{tr}}\beta_k^2(M-K)}$, $\sigma_k = \frac{(\tau p^{\mathsf{tr}}\beta_k+\sigma_{bs})\sigma_{bs}}{\tau p^{\mathsf{tr}}\beta_k^2(M-K)}$.

The training is performed for each coherence bandwidth chunk B_c in the total available bandwidth of W. For each channel coherence interval of T symbol periods, UEs will spend τ symbol periods for training and the remaining $T-\tau$ symbol periods for data transmission. Then, the upper bound of the average transmission time and energy (including the training time and energy) can be written, respectively, as

$$t_{k,\mathsf{IP}}^{\mathsf{t,ub}} = \frac{T}{T-\tau}t_{k,\mathsf{IP1}}^{\mathsf{t,ub}}, \tag{17}$$

$$\xi_{k,\mathsf{IP}}^{\mathsf{t,ub}} = \left(\frac{\tau(p^{\mathsf{tr}}+p_{k,c})}{T-\tau}+p_k+p_{k,c}\right)t_{k,\mathsf{IP1}}^{\mathsf{t,ub}}, \tag{18}$$

where $t_{k,\mathsf{IP1}}^{\mathsf{t,ub}} = (\hat{r}_k^{\mathsf{lb}})^{-1}\sum_{l_k\in\mathcal{L}_k}(1-s_{k,l_k})b_{k,l_k}$.

The total latency experienced by an offloaded task has different components, namely the time required for sending transmission bits (e.g., program states) to the cloud, the computation time in the cloud, and the time required for downloading the results to the UE from the cloud. For many practical applications, the amount of data required to report the computation outcome is typically much smaller than the amount of offloading data [19]. Therefore, we first omit the time for sending back the computation outcome from the cloud server to mobile users in our design and we will consider it in Section 5.

2.4 Problem Formulation

In this paper, we consider minimizing the energy consumption from the users' perspective to prolong their lifetime as studied in many recent works [9], [19]. Moreover, our design aims to achieve fairness for different users. Toward this end, we jointly optimize the offloading decisions, computation and radio resource allocation to minimize the maximum weighted energy consumption at mobile users considering latency and limited computation-radio resource constraints. This problem can be formulated as follows:

$$\begin{aligned}(\mathcal{P}_1)\quad &\min_{\boldsymbol{S},\boldsymbol{f},\boldsymbol{f}^{\mathsf{c}},\boldsymbol{p}}\ \max_k\ w_k(\xi_k^{\mathsf{lo}}+\xi_k^{\mathsf{t}})\\ \text{s.t}\quad &(\text{C1}):\ t_k^{\mathsf{lo}}\le\eta_k,\quad \forall k,\\ &(\text{C2}):\ t_k^{\mathsf{t}}+t_k^{\mathsf{c}}\le\eta_k,\ \forall k,\\ &(\text{C3}):\ s_{k,l_k}\in\{0,1\},\quad \forall k\\ &(\text{C4}):\ \sum_{k\in\mathcal{K}_1}f_k^{\mathsf{c}}\le F^{\mathsf{c}},\quad f_k^{\mathsf{c}}\ge 0,\\ &(\text{C5}):\ 0\le f_k\le F_k,\quad \forall k,\\ &(\text{C6}):\ 0\le p_k\le P_k,\quad \forall k,\end{aligned}$$

where w_k denotes the energy weight of UE k, $\boldsymbol{S}=\{\boldsymbol{s}_k,\forall k\}$, $\boldsymbol{s}_k=\{s_{k,l_k},\forall l_k\}$, $\{\boldsymbol{f},\boldsymbol{f}^{\mathsf{c}},\boldsymbol{p}\}=\{f_k,f_k^{\mathsf{c}},p_k,\forall k\}$, η_k is the maximum allowable delay of UE k, F_k denotes the maximum computation capacity of UE k, and P_k represents the maximum transmit power of UE k, t_k^{t} and ξ_k^{t} stand for transmission time and energy, respectively, which can be expressed for the P-CSI and IP-CSI scenarios as

$$t_k^{\mathsf{t}} = \begin{cases} t_{k,\mathsf{P}}^{\mathsf{t,ub}},\ \text{P-CSI}\\ t_{k,\mathsf{IP}}^{\mathsf{t,ub}},\ \text{IP-CSI}\end{cases}\quad \xi_k^{\mathsf{t}} = \begin{cases}\xi_{k,\mathsf{P}}^{\mathsf{t,ub}},\ \text{P-CSI}\\ \xi_{k,\mathsf{IP}}^{\mathsf{t,ub}},\ \text{IP-CSI.}\end{cases}$$

In this problem, (C1) captures the delay requirements for local computation while (C2) represents the total latency requirements for the offloaded tasks. The binary offloading decision is described in constraints (C3) while the limited computing resources are represented in constraints (C4) and (C5), where (C4) describes this constraint for the cloud and (C5) captures these constraints at the UEs. Finally, (C6) describes the UEs' maximum transmit power constraints.

2.5 Problem Transformation

To gain insights into the non-smooth min-max objective function, we introduce an auxiliary variable ξ and transform $(\mathcal{P}_1)$ to the following equivalent problem:

$$\begin{aligned}(\mathcal{P}_2)\quad &\min_{\boldsymbol{S},\boldsymbol{f},\boldsymbol{f}^{\mathsf{c}},\boldsymbol{p},\xi}\quad \xi\\ \text{s.t}\quad &(\text{C0}):\ w_k(\xi_k^{\mathsf{lo}}+\xi_k^{\mathsf{t}})\le\xi,\forall k,\ (\text{C1})-(\text{C6}).\end{aligned}$$

We now state an important result for this transformed problem in the following proposition.

Proposition 1. *The optimal value of f_k is equal to $(\eta_k)^{-1}\sum_{l_k\in\mathcal{L}_k}s_{k,l_k}c_{k,l_k}$ if there exists a feasible set of s_{k,l_k} such as f_k is less than or equal to F_k for all UE k.*

Proof. It can be verified from (2) that the local energy computation of UE k increases with the CPU clock speed f_k. Therefore, this energy component achieves its minimal value at the smallest possible value of f_k. From (3), (C1) and (C5), one can infer that $f_k = (f_k)_{\min} = (\eta_k)^{-1}\sum_{l_k\in\mathcal{L}_k}s_{k,l_k}c_{k,l_k}\le F_k$ at optimality. □

From the results of Proposition 1, the local computation energy ξ_k^{lo} and constraints (C5) can be rewritten as the function of the offloading decision variable s_{k,l_k} as

$$\xi_k^{\mathsf{lo}} = \alpha_k\eta_k^{-2}\left(\sum_{l_k\in\mathcal{L}_k}s_{k,l_k}c_{k,l_k}\right)^3,\quad \forall k\in\mathcal{K}, \tag{19}$$

$$\sum_{l_k \in \mathcal{L}_k} s_{k,l_k} c_{k,l_k} \leq \eta_k F_k, \quad \forall k \in \mathcal{K}. \tag{20}$$

Therefore, the objective function and constraint functions of problem $(\mathcal{P}_2)$ can be expressed in $\boldsymbol{S}, \boldsymbol{f}^{\mathsf{c}}, \boldsymbol{p}, \xi$ and this problem can be recast as

$$(\mathcal{P}_2) \min_{\boldsymbol{S}, \boldsymbol{f}^{\mathsf{c}}, \boldsymbol{p}, \xi} \xi \quad \text{s.t} \quad (\mathrm{C0}), (\mathrm{C2}) - (\mathrm{C4}), (\mathrm{C6}), (20).$$

This problem is difficult to solve due to the complex fractional form of the transmission time and energy, the logarithmic transmission rate function, binary variables $\boldsymbol{S}$ and continuous variables $\boldsymbol{f}^{\mathsf{c}}, \boldsymbol{p}, \xi$. In the following sections, we will present our proposed algorithms to solve problem $(\mathcal{P}_2)$ for the P-CSI and IP-CSI scenarios.

3 Algorithm Design for P-CSI Scenario

To solve the difficult MINLP $(\mathcal{P}_2)$, we propose two algorithms where the first one (P-O) can find the global optimal solution while the second one (P-SO) achieves a solution with lower complexity. In (P-O) algorithm, we first employ the bisection search for ξ where the upper-bound $\xi_{\max}$ and lower-bound $\xi_{\min}$ of ξ are iteratively updated until the difference between them is sufficiently small. This updating mechanism is based on the feasibility verification of problem $(\mathcal{P}_2)$ for a given value of ξ as follows. If the set of constraints is feasible, then the upper-bound of the objective function will decrease and is set equal to ξ; otherwise, its lower-bound will increase and is set equal to ξ. To verify the feasibility, we decompose this problem into individual users' subproblems, then find the minimum allocated computing resource from the cloud server to each user. This is done by searching all offloading decision combinations of each user. The total required computing resource of all users and its available budget from the cloud server is compared to determine the feasibility condition.

In the (P-SO) algorithm, the original problem is decomposed into the offloading optimization (OP) and power allocation (PA) subproblems which are solved iteratively. For the (OP) subproblem, we directly find the offloading decisions via linearizing the non-convex constraints. Besides, we also propose an indirect method to tackle this problem via the search of offloading decision combinations; however, it can be done rapidly compared to the search in the (P-O) algorithm, which will be presented more detail later. For the (PA) subproblem, we apply the bisection search and in each iteration, we also compare the required and available computing resource from cloud server to verify the feasibility of convex constraints.

3.1 P-CSI - Optimal Algorithm (P-O)

The feasibility verification of non-convex mixed integer constraints for a given value of ξ is still very challenging. Fortunately, the lower-bound of the achievable rate in (13) depends only on p_k and the number of offloading UEs $|\mathcal{K}_1|$, which means that all constraints (C0), (C2), (C3), (C6) and (20) are independent for different UEs for a given value of ξ. Therefore, we can decompose this problem into the feasibility verification problems for individual UEs if we can deal with the dependent relation of UEs in constraints (C4). Toward this end, we first remove constraints (C4), determine $|\mathcal{K}_1|$, and then find the minimum allocated computing resource from the cloud server for each user. The constraint (C4) will be then verified by using the obtained computing resource allocation solution. Specifically, for a given value of ξ, UE k should upload its computation tasks if the local execution consumes the total energy greater than ξ. Therefore, the number of offloading UEs $|\mathcal{K}_1|$ can be computed as follows:

$$|\mathcal{K}_1| = \sum_k \delta_k; \delta_k = \begin{cases} 1, & \text{if } w_k \xi_k^{\mathsf{lo}} > \xi, \ \forall l_k \text{ st } s_{k,l_k} = 1 \\ 0, & \text{otherwise.} \end{cases} \tag{21}$$

We now study the following subproblems that can be solved independently by individual UEs $(k \in \mathcal{K}_1)$ for given values of ξ and $|\mathcal{K}_1|$:

$$(\mathcal{P}_3)_k \min_{s_k, f_k^{\mathsf{c}}, p_k} [f_k^{\mathsf{c}}]^+ \quad \text{s.t.} \quad (\mathrm{C0})_k, (\mathrm{C2})_k, (\mathrm{C3})_k, (\mathrm{C6})_k, (20)_k,$$

where $[f_k^{\mathsf{c}}]^+ = \max(f_k^{\mathsf{c}}, 0)$, constraints $(\mathrm{C0})_k$, $(\mathrm{C2})_k$, $(\mathrm{C3})_k$, $(\mathrm{C6})_k$, $(20)_k$ denote the corresponding constraints (C0), (C2), (C3), (C6), (20) for UE k, respectively.

Suppose that all sub-problems $(\mathcal{P}_3)_k, \forall k$ can be solved. Then, constraint (C4), which couples all UEs, can be addressed by using the result in the following proposition.

Proposition 2. *For a given value of ξ, problem $(\mathcal{P}_2)$ is feasible if all subproblems $(\mathcal{P}_3)_k, \forall k$ are feasible and $\sum_{k \in \mathcal{K}_1} f_{k,\xi}^{\mathsf{c,min}} \leq F^{\mathsf{c}}$ where $f_{k,\xi}^{\mathsf{c,min}}$ is the optimal value of $(\mathcal{P}_3)_k$.*

Proof. If all subproblems $(\mathcal{P}_3)_k, \forall k \in \mathcal{K}_1$ are feasible, constraints (C0), (C2), (C3), (C6), (20) are satisfied and the required CPU clock speeds f_k^{c} of all UEs are at their minimum; thus, the total CPU clock speed $\sum_{k \in \mathcal{K}_1} f_{k,\xi}^{\mathsf{c,min}}$ is also minimum. As a result, constraint (C4) is satisfied if the minimum required computing resources for different UEs satisfy $\sum_{k \in \mathcal{K}_1} f_{k,\xi}^{\mathsf{c,min}} \leq F^{\mathsf{c}}$, which then implies that problem $(\mathcal{P}_2)$ is feasible. □

Algorithm 1. Optimal Algorithm—P-CSI (P-O)

1: **Initialize**: choose ϵ, $\xi_{\min} = 0$ and $\xi_{\max} = \min(\max(w_k \xi_k^{\mathsf{lo}} | s_{k,l_k} = 1, \sum_{l_k \in \mathcal{L}_k} c_{k,l_k} \leq \eta_k F_k), \xi^{\infty})$.
2: **while** $\xi_{\max} - \xi_{\min} < \epsilon$ **do**
3: Assign $\xi = (\xi_{\max} + \xi_{\min})/2$.
4: Determine set $\mathcal{K}_1$ as in (21).
5: Solve $(\mathcal{P}_3)_k$ to get $f_{k,\xi}^{\mathsf{c,min}}$ for all $k \in \mathcal{K}_1$.
6: Assign $feasibility = true$ if all subproblems $(\mathcal{P}_3)_k$ are feasible and $\sum_{k \in \mathcal{K}_1} f_{k,\xi}^{\mathsf{c,min}} \leq F^{\mathsf{c}}$.
7: Assign $(\xi_{\max}, \xi_{\min})$ = bisectionSearch$(feasibility, \xi)$
8: **end while**

Using the results in Proposition 2, we propose an optimal algorithm to solve problem $(\mathcal{P}_2)$ as described in Algorithm 1. In this algorithm, the offloading decisions and the allocation of cloud computing resource can be decided by the UEs. Moreover, the BS broadcasts the value of ξ and UEs report their computation demands in terms of CPU clock frequency f_k^{c}. The remaining challenge now is to solve small-scale non-convex MINLP subproblems $(\mathcal{P}_3)_k$ to obtain the global optimum solution. Problem $(\mathcal{P}_3)_k$ is still difficult to tackle because it is a non-convex MINLP problem. Fortunately, the number of parallel tasks of each UE is not large in practice; therefore,

we can solve p_k and f_k^{c} by exploring all possible sets of $\boldsymbol{s}_k$ (i.e., all possible offloading solutions of UE k). Specifically, for each set of $\boldsymbol{s}_k$ satisfying $(20)_k$, let $\xi^{\mathsf{a}} = (w_k b_k^{\mathsf{a}})^{-1} W(\xi - w_k \xi_k^{\mathsf{lo}})$, $c_k^{\mathsf{a}} = \sum_{l_k \in \mathcal{L}_k} (1 - s_{k,l_k}) c_{k,l_k}$, and $b_k^{\mathsf{a}} = \sum_{l_k \in \mathcal{L}_k} (1 - s_{k,l_k}) b_{k,l_k}$, we need to solve the following subproblem:

$$
\begin{aligned}
(\mathcal{P}_3)'_k \quad & \min_{f_k^{\mathsf{c}}, p_k} [f_k^{\mathsf{c}}]^+ \\
\text{s.t} \quad & (\mathrm{C0}')_k : \; p_k + p_{k,c} - \xi^{\mathsf{a}} \log_2(1 + p_k \beta_k^{\mathsf{a}}) \le 0, \\
& (\mathrm{C2})_k : \; \frac{b_k^{\mathsf{a}}}{W \log_2(1 + p_k \beta_k^{\mathsf{a}})} + \frac{c_k^{\mathsf{a}}}{f_k^{\mathsf{c}}} - \eta_k \le 0, \; (\mathrm{C6})_k.
\end{aligned}
$$

The optimal objective value of $(\mathcal{P}_3)_k$ is equal to the minimum of the optimal objective values of $(\mathcal{P}_3)'_k$ considering all different combinations of s_{k,l_k} ($l_k \in \mathcal{L}_k$). The optimal solution structure for $(\mathcal{P}_3)'_k$ is presented in the following proposition.

Proposition 3. *The solution of $(\mathcal{P}_3)'_k$ corresponding to a set $\boldsymbol{s}_k$ can be expressed as*

$$
(f_{k,\xi}^{\mathsf{c,min}})^{(\boldsymbol{s}_k)} = \frac{c_k^{\mathsf{a}} \log_2(1 + p_k^{\star} \beta_k^{\mathsf{a}})}{\eta_k \log_2(1 + p_k^{\star} \beta_k^{\mathsf{a}}) - b_k^{\mathsf{a}}/W}, \tag{22}
$$

where $p_k^{\star} = P_k$ if $g_k(P_k) \le 0$ (note that $g_k(p_k) = p_k + p_{k,c} - \xi^{\mathsf{a}} \log_2(1 + p_k \beta_k^{\mathsf{a}})$ defines the left-hand-side of constraints $(\mathrm{C0}')_k$); otherwise, $p_k^{\star}$ is the root of equation $g_k(p_k) = 0$ and it satisfies $p_k^{\mathsf{a}} < p_k^{\star} < P_k$ where $p_k^{\mathsf{a}} = (\beta_k^{\mathsf{a}})^{-1} (2^{b_k^{\mathsf{a}}/(\eta_k W)} - 1)$.

Proof. The proof is given in Appendix A. □

Proposition 3 implies that we need to solve equation $g_k(p_k^{\star}) = 0$ to obtain the optimal solution if $g_k(P_k) > 0$. Toward this end, it is necessary to check whether or not there exists a value of $p_k^{\star}$ satisfying $p_k^{\mathsf{a}} < p_k^{\star} < P_k$ and $g_k(p_k^{\star}) = 0$. Because of the convexity of $g_k(p_k)$, the requisite conditions to ensure $(\mathrm{C0}')_k$ holds are $p_k^{\mathsf{b}} \le p_k^{\star} < P_k$ and $g_k(p_k^{\mathsf{b}}) \le 0$ where p_k^{b} is the stationary point of $g_k(p_k)$ (the point for which the first derivative of $g_k(p_k)$ equals to zero), which can be derived as $p_k^{\mathsf{b}} = \frac{\xi^{\mathsf{a}}}{\ln 2} - \frac{1}{\beta_k^{\mathsf{a}}}$. If $p_k^{\mathsf{b}} \le p_k^{\mathsf{a}}$ and $g_k(p_k^{\mathsf{a}}) > 0$, then $g_k(p_k) > 0$ for $p_k^{\mathsf{a}} < p_k < P_k$. If all requisite conditions are satisfied, we can find the root $p_k^{\star} \ge p_k^{\mathsf{b}}$ by employing the Newton-Raphson search method through the following iterative updates:

$$
p_k^{t+1} = p_k^{\mathsf{t}} - \frac{g_k(p_k^{\mathsf{t}})(1 + p_k^{\mathsf{t}} \beta_k^{\mathsf{a}}) \ln(2)}{(1 + p_k^{\mathsf{t}} \beta_k^{\mathsf{a}}) \ln(2) - \xi^{\mathsf{a}} \beta_k^{\mathsf{a}}}, \tag{23}
$$

where the initial point p_k^0 can be set equal to P_k. Note that we can also apply one dimensional bisection search to find the largest possible value of p_k.

In summary, the optimal solution of $(\mathcal{P}_3)'_k$ can be obtained as described in Algorithm 2 where $\mathbb{1}_x$ denotes an indicator function (i.e., $\mathbb{1}_x = 1$ if condition x holds and $\mathbb{1}_x = 0$, otherwise). Moreover, $(\mathcal{P}_3)_k$ is feasible if there exists at least one set $\boldsymbol{s}_k$ so that $(\mathcal{P}_3)'_k$ is feasible and we then have $f_{k,\xi}^{\mathsf{c,min}} = \min_{\boldsymbol{s}_k} (f_{k,\xi}^{\mathsf{c,min}})^{(\boldsymbol{s}_k)}$.

Algorithm 2. Solving Problem $(\mathcal{P}_3)'_k$ for Set $\boldsymbol{s}_k$

Set $feasibility = true$ and compute b_k^{a}, ξ^{a} and c_k^{a}
Calculate $fg1 = \mathbb{1}_{b_k^{\mathsf{a}}/\eta_k - W \log_2(1 + P_k \beta_k^{\mathsf{a}}) < 0}$.
if $fg1 = 1$ **then** Calculate $fg2 = \mathbb{1}_{g_k(P_k) \le 0}$.
  if $fg2 = 1$ **then**
    Assign $p_k^{\star} = P_k$ and $(f_{k,\xi}^{\mathsf{c,min}})^{(\boldsymbol{s}_k)}$ as in (22).
  else Calculate $fg3 = \mathbb{1}_{g_k(p_k^{\mathsf{b}}) \le 0}$
    if $fg3 = 1$ **then**
      Calculate $fg4 = \mathbb{1}_{p_k^{\mathsf{a}} - p_k^{\mathsf{b}} \ge 0} \cdot \mathbb{1}_{g_k(p_k^{\mathsf{a}}) > 0}$
      if $fg4 = 1$ **then** Assign $feasibility = false$
      else ; Apply (23) to obtain $p_k^{\star}$.
      end if
    else Assign $feasibility = false$
    end if
    if $feasibility$ **then** Assign $(f_{k,\xi}^{\mathsf{c,min}})^{(\boldsymbol{s}_k)}$ as in (22).
    end if
  end if
else Assign $feasibility = false$
end if

3.2 P-CSI—Low-Complexity Algorithm (P-SO)

The complexity of the optimal algorithm (i.e., Algorithm 1) strongly depends on the number of possible combinations of task offloading decisions. Thus, it can be highly complex to find the optimal solution when the number of computation tasks is large. To deal with such complexity, we propose a low-complexity algorithm which iteratively solves two subproblems decomposed from problem $(\mathcal{P}_2)$ where the first one, i.e., the offloading optimization (OP) subproblem, determines offloading decision and computing resource allocation while the second one, i.e., the power allocation (PA) subproblem, performs uplink power allocation and reassigns the computing resource. First, for a given value of $\boldsymbol{p}$, the (OP) subproblem is given as follows:

$$
(\mathcal{P}_2^{\mathsf{OP}}) \; \min_{\boldsymbol{S}, \boldsymbol{f}^{\mathsf{c}}, \xi} \; \xi \quad \text{s.t} \quad (\mathrm{C0}), (\mathrm{C2}) - (\mathrm{C4}), (20).
$$

Second, with the offloading solution $\boldsymbol{S}$ obtained by solving $(\mathcal{P}_2^{\mathsf{OP}})$, the (PA) subproblem is given as

$$
(\mathcal{P}_{2,\mathsf{P}}^{\mathsf{PA}}) \; \min_{\boldsymbol{p}, \boldsymbol{f}^{\mathsf{c}}, \xi} \; \xi \quad \text{s.t} \quad (\mathrm{C0}'), (\mathrm{C2}), (\mathrm{C4}), (\mathrm{C6}).
$$

The proposed algorithm (P-SO), which iteratively solves $(\mathcal{P}_2^{\mathsf{OP}})$ and $(\mathcal{P}_{2,\mathsf{P}}^{\mathsf{PA}})$ until convergence, is described in Algorithm 3. Besides, this approach is the key for solving problem $(\mathcal{P}_2)$ in the IP-CSI scenario when the finding of optimal solution would be impossible. Note that $\xi^{(q)}|_{(\mathcal{P}_{2,\mathsf{P}}^{\mathsf{PA}})}$ is the optimal of subproblem $(\mathcal{P}_{2,\mathsf{P}}^{\mathsf{PA}})$ at iteration q. We describe how to solve (OP) and (PA) in the following.

3.2.1 Offloading Subproblem (OP)

In order to tackle this subproblem, we will apply the decomposition technique as employed in Section 3.1 to further decompose this subproblem into individual users' small-scale subproblems:

$$
(\mathcal{P}_2^{\mathsf{OP}})_k \; \min_{\boldsymbol{s}_k, f_k^{\mathsf{c}}} [f_k^{\mathsf{c}}]^+ \; \text{s.t.} \quad (\mathrm{C0})_k, (\mathrm{C2})_k, (\mathrm{C3})_k, (20)_k.
$$

To solve $(\mathcal{P}_2^{\mathsf{OP}})_k$, we propose two methods: the first one will transform the non-convex constraints into convex constraints and then solve the corresponding convex MINLP

(Method 1) while the other will find $f_{k,\xi}^{\mathsf{c,min}}$ by directly dealing with all possible combinations of $\boldsymbol{s}_k$ (Method 2).

a) Method 1: We first rewrite the offloading time constraint $(\mathrm{C2})_k$ as

$$r_k^{-1}\sum_{l_k\in\mathcal{L}_k} f_k^{\mathsf{c}}(1-s_{k,l_k})b_{k,l_k}+\sum_{l_k\in\mathcal{L}_k}(1-s_{k,l_k})c_{k,l_k}-f_k^{\mathsf{c}}\eta_k\leq 0. \quad (24)$$

The non-convex term $z_{k,l_k}=f_k^{\mathsf{c}}s_{k,l_k}$ can be transformed into a linear form as follows:

$$0\leq z_{k,l_k}\leq s_{k,l_k}F^{\mathsf{c}}, \quad (25)$$

$$0\leq f_k^{\mathsf{c}}-z_{k,l_k}\leq (1-s_{k,l_k})F^{\mathsf{c}}. \quad (26)$$

The small-scale subproblem $(\mathcal{P}_2^{\mathsf{OP}})_k$ with these transformations becomes a convex mixed integer non-linear (cubic polynomial) problem of $\xi,\boldsymbol{s}_k,z_k,f_k^{\mathsf{c}}$, which can be solved efficiently by available solvers such as GAMS-BARON, CVX-Gurobi, CVX-MOSEK thanks to the convexity property [32].

b) Method 2: In this method, we apply bisection search on ξ as in Algorithm 1, except for some difference in step 4 and step 5 to find $f_{k,\xi}^{\mathsf{c,min}}$. Let $\boldsymbol{S}_k^{\mathsf{bi}}\in\mathbb{R}^{2^{|\mathcal{L}_k|}\times|\mathcal{L}_k|}$ denote the binary matrix whose rows represent all possible combinations of task offloading decisions of UE k. For example, $\boldsymbol{S}_k^{\mathsf{bi}}=\begin{bmatrix}0&0&1&1\\0&1&0&1\end{bmatrix}^{\mathsf{t}}$ in case of $|\mathcal{L}_k|=2$. Then, it can be verified that the minimum value of f_k^{c} for a given value of ξ can be computed as

$$\begin{aligned}
f_{k,\xi}^{\mathsf{c,min}} &= \min((LC0_k\odot LC2_k\odot L20_k)\backslash\{0\}),\\
LC0_k &= \mathbb{1}_{\xi\times\mathbf{1}_{2^{|\mathcal{L}_k|}}-LC0'_k\geq 0},\\
LC0'_k &= w_k\alpha_k\eta_k^{-2}(\boldsymbol{S}_k^{\mathsf{bi}}\boldsymbol{c}_k)^3+\frac{w_k(p_k+p_{k,c})(1-\boldsymbol{S}_k^{\mathsf{bi}})\boldsymbol{b}_k}{r_k^{\mathsf{lb}}},\\
LC2_k &= \left[(1-\boldsymbol{S}_k^{\mathsf{bi}})\boldsymbol{c}_k\oslash\left(\eta_k\times\mathbf{1}_{2^{|\mathcal{L}_k|}}-\frac{(1-\boldsymbol{S}_k^{\mathsf{bi}})\boldsymbol{b}_k}{r_k^{\mathsf{lb}}}\right)\right]^+,\\
L20_k &= \mathbb{1}_{\eta_kF_k-\boldsymbol{S}_k^{\mathsf{bi}}\boldsymbol{c}_k\geq 0},
\end{aligned} \quad (27)$$

where $\odot$ and $\oslash$ denote the Hadamard product and division, respectively, $\mathbf{1}_n$ represents the $n\times 1$ vector of ones, $\mathbb{1}_{\boldsymbol{x}\geq 0}$ is the indicator function and $\boldsymbol{x}^+=\max(\boldsymbol{x},0)$. In above expressions, the elements of $LC0_k$ and $L20_k$ will be equal to 1 if the corresponding row of $\boldsymbol{S}_k^{\mathsf{bi}}$ satisfies constraint $(\mathrm{C0})_k$ and $(20)_k$, respectively. The vector of $LC2_k$ describes the minimum value of f_k^{c} corresponding to each row of $\boldsymbol{S}_k^{\mathsf{bi}}$.

It is noted that $LC0'_k, LC2_k$ and $L20_k$ do not depend on the value of ξ; thus, we just need to compute them at the beginning of the bisection search (the 'while-loop' in Algorithm 1) and use them to update $LC0_k$ and $f_{k,\xi}^{\mathsf{c,min}}$ corresponding to the updated value of ξ. Because this method considers all possible values of $\boldsymbol{s}_k$, it is more suitable for the setting with a small number of tasks per UE.

3.2.2 Uplink Power Allocation Subproblem (PA)

With the solution of $(\mathcal{P}_2^{\mathsf{OP}})$, we can then solve the $(\mathcal{P}_{2,\mathsf{P}}^{\mathsf{PA}})$ subproblem to obtain the optimal solutions of transmit power $\boldsymbol{p}_k$ and computing resource allocation $\boldsymbol{f}^{\mathsf{c}}$. This can be fulfilled by using a similar process employed in Section 3.1 which applies the bisection search on ξ and solving subproblem $(\mathcal{P}_3)'_k$. We state important results for Algorithm 3 in the following proposition.

Algorithm 3. Low-Complexity Algorithm—P-CSI (P-SO)

1: **Initialize**: choose ϵ, initial $p_k^{(0)}=P_{\max}/2,\forall k$.
2: **while** $\xi^{(q)}|_{(\mathcal{P}_2^{\mathsf{OP}})}-\xi^{(q)}|_{(\mathcal{P}_2^{\mathsf{PA}})}<\epsilon$ **do**
3: Assign $q=q+1$;
4: Solve $\mathcal{P}_2^{\mathsf{OP}}$ to get $\boldsymbol{S}^{(q)}$, $(\boldsymbol{f}^{\mathsf{c}})^{(q)}$ and $\xi^{(q)}|_{(\mathcal{P}_2^{\mathsf{OP}})}$.
5: Solve $\mathcal{P}_{2,\mathsf{P}}^{\mathsf{PA}}$ to get $\boldsymbol{p}^{(q)}$, $(\boldsymbol{f}^{\mathsf{c}})^{(q)}$ and $\xi^{(q)}|_{(\mathcal{P}_{2,\mathsf{P}}^{\mathsf{PA}})}$.
6: **end while**

Proposition 4. *Algorithm 3 creates a sequence of feasible solutions for $(\mathcal{P}_2)$ where the objective function value of this problem monotonically decreases over iterations.*

Proof. Let $(\mathcal{V}^{\mathsf{OP}})^{(q-1)}$ denote the optimal point of $(\mathcal{P}_2^{\mathsf{OP}})$ and $(\mathcal{V}^{\mathsf{PA}})^{(q-1)}$ denote the optimal point of $(\mathcal{P}_{2,\mathsf{P}}^{\mathsf{PA}})$ at iteration $q-1$. Clearly, $((\mathcal{V}^{\mathsf{PA}})^{(q-1)},\boldsymbol{S}^{(q-1)})$ is a feasible solution of $(\mathcal{P}_2^{\mathsf{OP}})$ at iteration q. Thus, at iteration q, we have $\xi^{(q)}|_{(\mathcal{P}_2^{\mathsf{OP}})}=\min_{\mathcal{V}^{\mathsf{OP}}\supset\{(\mathcal{V}^{\mathsf{PA}})^{(q-1)},\boldsymbol{S}^{(q-1)}\}}\xi|_{(\mathcal{P}_2^{\mathsf{OP}})}\leq\xi^{(q-1)}|_{(\mathcal{P}_{2,\mathsf{P}}^{\mathsf{PA}})}=\min_{\mathcal{V}^{\mathsf{PA}}\supset\{(\mathcal{V}^{\mathsf{OP}})^{(q-1)},\boldsymbol{p}^{(q-2)}\}}\xi|_{(\mathcal{P}_{2,\mathsf{P}}^{\mathsf{PA}})}\leq\xi^{(q-1)}|_{(\mathcal{P}_2^{\mathsf{OP}})}$. Hence, the iterative process will converge in a finite number of iterations. □

4 Algorithm Design for IP-CSI Scenario

We employ a similar approach, which is used to develop the low-complexity algorithm (P-SO) for the P-CSI scenario to tackle the problem in this IP-CSI scenario. Specifically, we tackle problem $(\mathcal{P}_2)$ by iteratively solving two subproblems $(\mathcal{P}_2^{\mathsf{OP}})$ and $(\mathcal{P}_{2,\mathsf{IP}}^{\mathsf{PA}})$ until convergence. The proposed algorithm for the IP-CSI scenario is refered to as (IP-SO) in the sequel. This (IP-SO) algorithm is similar to Algorithm 3; hence, we do not present it for brevity. Because the IP-CSI only affects the transmission energy and transmission time, the (OP) subproblem $(\mathcal{P}_2^{\mathsf{OP}})$ can be solved as in Section 3.2.1. We only need to consider the (PA) subproblem $(\mathcal{P}_{2,\mathsf{IP}}^{\mathsf{PA}})$, which can be written as

$$\begin{aligned}
(\mathcal{P}_{2,\mathsf{IP}}^{\mathsf{PA}})\quad &\min_{\boldsymbol{p},\boldsymbol{f}^{\mathsf{c}}}\ \xi\\
\text{s.t}\quad &(\mathrm{C0}):\ w_k(\xi_{k,\mathsf{IP}}^{\mathsf{t,ub}}+\xi_k^{\mathsf{lo}})\leq\xi,\\
&(\mathrm{C2}):\ t_{k,\mathsf{IP}}^{\mathsf{t,ub}}+\frac{c_k^{\mathsf{a}}}{f_k^{\mathsf{c}}}\leq\eta_k,\ \ (\mathrm{C4}),(\mathrm{C6}).
\end{aligned}$$

Assuming that the UE's training power is fixed in the CSI estimation phase, the NP-hardness of $(\mathcal{P}_{2,\mathsf{IP}}^{\mathsf{PA}})$ is stated in the following proposition.

Proposition 5. *The subproblem $(\mathcal{P}_{2,\mathsf{IP}}^{\mathsf{PA}})$ is NP-hard.*

Proof. The proof is given in Appendix B. □

Even though $(\mathcal{P}_{2,\mathsf{IP}}^{\mathsf{PA}})$ is non-convex and NP-hard, we can solve it by using the bisection search method. The key step in this bisection search is to perform the feasibility verification of $(\mathcal{P}_{2,\mathsf{IP}}^{\mathsf{PA}})$ for a given ξ (as can be seen in Algorithm 1). We propose to employ the DC optimization method to convexify and tackle this feasibility verification problem [33]. We will show later that our proposed algorithm to solve $(\mathcal{P}_{2,\mathsf{IP}}^{\mathsf{PA}})$ in this IP-CSI scenario converges to a stationary

point, which, therefore, guarantees the convergence of the main algorithm (IP-SO) as stated in Proposition 4.

We now proceed to address the feasibility verification problem for $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})$. The transmission rate in (16) is the difference between two concave functions, which makes $1/\hat{r}^{\mathsf{lb}}_k(\boldsymbol{p})$ in constraints (C0) and (C2) non-convex. To convexify $1/\hat{r}^{\mathsf{lb}}_k(\boldsymbol{p})$, we approximate $\hat{r}^{\mathsf{lb}}_k(\boldsymbol{p})$ at point $\boldsymbol{p}^{(q)}$ by a concave function $\tilde{r}^{\mathsf{lb}}_k(\boldsymbol{p}|\boldsymbol{p}^{(q)})$. Indeed, it can be verified that $1/x$ is a convex and non-increasing function of x and if $\tilde{r}^{\mathsf{lb}}_k(\boldsymbol{p}|\boldsymbol{p}^{(q)})$ is a concave function of $\boldsymbol{p}$, the composition function $1/\tilde{r}^{\mathsf{lb}}_k(\boldsymbol{p}|\boldsymbol{p}^{(q)})$ is convex in $\boldsymbol{p}$ [34].

To obtain a concave function $\tilde{r}^{\mathsf{lb}}_k(\boldsymbol{p}|\boldsymbol{p}^{(q)})$, the second concave term $v_k(\boldsymbol{p}) = W\log_2\left(\boldsymbol{p}^{\mathsf{t}}\boldsymbol{\lambda}_k + \sigma_k\right)$ at point $\boldsymbol{p}^{(q)}$ of $\hat{r}^{\mathsf{lb}}_k(\boldsymbol{p})$ in (16) can be approximated by a linear function as

$$v_k(\boldsymbol{p}) \le \tilde{v}_k(\boldsymbol{p}) = v_k(\boldsymbol{p}^{(q)}) + \nabla v_k(\boldsymbol{p}^{(q)})(\boldsymbol{p} - \boldsymbol{p}^{(q)}), \tag{28}$$

where $\nabla v_k(\boldsymbol{p}^{(q)})$ is the gradient of v_k at point $\boldsymbol{p}^{(q)}$. Using this approximation, for a given value of ξ and point $\boldsymbol{p}^{(q)}$, we can obtain the concave approximation function $\tilde{r}^{\mathsf{lb}}_k(\boldsymbol{p}|\boldsymbol{p}^{(q)})$ of $\hat{r}^{\mathsf{lb}}_k(\boldsymbol{p})$ by using its lower bound as follows:

$$\begin{aligned}\hat{r}^{\mathsf{lb}}_k(\boldsymbol{p}) \ge{}& W\log_2(p_k + \boldsymbol{p}^{\mathsf{t}}\boldsymbol{\lambda}_k + \sigma_k) - W\log_2((\boldsymbol{p}^{(q)})^{\mathsf{t}}\boldsymbol{\lambda}_k + \sigma_k)\\ &- \frac{\boldsymbol{\lambda}_k(\boldsymbol{p} - \boldsymbol{p}^{(q)})}{\log(2)\left((\boldsymbol{p}^{(q)})^{\mathsf{t}}\boldsymbol{\lambda}_k + \sigma_k\right)} = \tilde{r}^{\mathsf{lb}}_k(\boldsymbol{p}|\boldsymbol{p}^{(q)}).\end{aligned} \tag{29}$$

As a result, for a given value of ξ and point $\boldsymbol{p}^{(q)}$, constraints (C0) and (C2) can be approximated by the following constraints, respectively:

$$p_k + p_{k,c} + \frac{\tau(p^{\mathsf{tr}} + p_{k,c})}{T - \tau} - \xi^{\mathsf{a}}\tilde{r}^{\mathsf{lb}}_k(\boldsymbol{p}|\boldsymbol{p}^{(q)}) \le 0, \tag{30}$$

$$\left(\frac{T}{T-\tau}\right)\frac{b^{\mathsf{a}}_k}{\tilde{r}^{\mathsf{lb}}_k(\boldsymbol{p}|\boldsymbol{p}^{(q)})} + \frac{c^{\mathsf{a}}_k}{f^{\mathsf{c}}_k} - \eta_k \le 0. \tag{31}$$

From (29), the feasibility verification of $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})$ for a given value of ξ is now equivalent to find at least one point ($\boldsymbol{f}^{\mathsf{c}}$, $\boldsymbol{p}$, $\boldsymbol{p}^{(q)}$) that makes constraints (C4) and (C6) and inequalities (30), (31) feasible. Toward this end, we will iteratively update $\boldsymbol{p}^{(q)}$ to make the approximation in (29) tighter and find the minimum χ for a given $\boldsymbol{p}^{(q)}$, where χ is an upper-bound of the functions in the left-hand-side of (30), (31) for all users and for all $\boldsymbol{p}, \boldsymbol{f}^{\mathsf{c}}$ satisfying (C4) and (C6). It is clear that if we can find such a minimum $\chi \le 0$, then constraints (30), (31) are satisfied; therefore, $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})$ will be feasible. Specifically, the minimum value of χ for a given $\boldsymbol{p}^{(q)}$ can be found by solving the following problem:

$$\begin{aligned}(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})^{(q)}\ \min_{\boldsymbol{p},\boldsymbol{f}^{\mathsf{c}},\chi}\ & \chi\\ \text{s.t}\quad & p_k + p_{k,c} + \frac{\tau(p^{\mathsf{tr}} + p_{k,c})}{T - \tau} - \xi^{\mathsf{a}}\tilde{r}^{\mathsf{lb}}_k(\boldsymbol{p}|\boldsymbol{p}^{(q)}) \le \chi,\\ & \left(\frac{T}{T-\tau}\right)\frac{b^{\mathsf{a}}_k}{\tilde{r}^{\mathsf{lb}}_k(\boldsymbol{p}|\boldsymbol{p}^{(q)})} + \frac{c^{\mathsf{a}}_k}{f^{\mathsf{c}}_k} - \eta_k \le \chi, (\mathrm{C4}), (\mathrm{C6}).\end{aligned} \tag{32}$$

As shown above, the convexity of $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})^{(q)}$ is guaranteed; therefore, we can effectively solve this problem by using the CVX-SDPT3 solver. Finally, the feasibility verification is presented in Algorithm 4, and the bisection search to solve $(\mathcal{P}^{\mathsf{PA}}_{2,\mathrm{IP}})$ is similar to Algorithm 1, except for the difference in step 5, where the feasibility verification is done as described in Algorithm 4.

Algorithm 4. PA Feasibility Verification—IP-CSI

1: **Initialize**: choose $\boldsymbol{p}^{(0)}$ as the previous solution of $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})$.
2: **repeat**
3: $q = q + 1$;
4: At $\boldsymbol{p} = \boldsymbol{p}^{(q-1)}$, solve $(\mathcal{P}^{\mathsf{PA}}_{2})^{(q-1)}$ to get $\boldsymbol{p}^{(q)}$, $\boldsymbol{f}^{\mathsf{c}}$
5: **if** $\chi < 0$ **then** Assign $feasibility = true$
6: Return $\boldsymbol{p}^{(q)}$, $\boldsymbol{f}^{\mathsf{c}}$; **break**;
7: **else** Assign $feasibility = false$
8: Compute $\tilde{r}^{\mathsf{lb}}_k(\boldsymbol{p}|\boldsymbol{p}^{(q)})$ for all k
9: **end If**
10: **until** convergence

We will show that our proposed algorithm to solve $(\mathcal{P}^{\mathsf{PA}}_{2,\mathrm{IP}})$ converges to a stationary point, which is stated in the following Propositions 6 and 7.

Proposition 6. *Proposition 6: For a given ξ, using D.C to approximate the transmission rate (using the rate lower bound in (29)) and iteratively solving problem $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})^{(q)}$ leads to convergence.*

Proof. The proof is given in Appendix C. □

Proposition 7. *If the optimal value of $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})^{(Q)}$ is equal to zero at the convergence of Algorithm 4, then the solution of $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})^{(Q)}$ combining with ξ gives a stationary point of subproblem $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})$, where Q denotes the final iteration index at the convergence of Algorithm 4.*

Proof. When the optimal value of $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})^{(Q)}$ is equal to zero at the convergence, the bisection search of ξ will terminate and the solution of $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})^{(Q)}$ combining with ξ is the solution of $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})$. On the other hand, the Karush-Kuhn-Tucker (KKT) conditions of $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})$ and $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})^{(Q)}$ at convergence are the same. Moreover, in each iteration of $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})^{(Q)}$ we always obtain the optimal solution, which therefore satisfies the KKT conditions of $(\mathcal{P}^{\mathsf{PA}}_{2,\mathsf{IP}})$. □

5 Extension and Complexity Analysis

5.1 Consideration of Downlink Transmission

For certain applications such as virtual-reality games, the amount of downlink data and time to send back the computation result from the BS can be non-negligible and it must be taken into account in the computation offloading design. In this section, we extend the proposed design described in the previous sections to consider this downlink transmission.

Let b^{dl}_{k,l_k} denote the number of downlink bits related to the computation result of task l_k, which must be sent from the BS to UE k. We assume the time division duplexing (TDD) wireless system[1], the downllink beamforming can be realized by using the same estimated channel matrix $\hat{\boldsymbol{H}}$ in the uplink. Let p^{dl}_k denote the power that BS uses to transmit the offloading result data to UE k, then the downlink signal-to-noise-plus-interference ratio of UE k is given by

1. The TDD approach has been advocated recently because it can significantly reduce the CSI estimation overhead, especially in massive MIMO wireless systems.

$$\gamma_{k,\mathsf{dl}} = \frac{p_k^{\mathsf{dl}} |\hat{\boldsymbol{h}}_k^H \hat{\boldsymbol{a}}_k|^2}{\sum_{i \neq k, i \in \mathcal{K}_1} p_i^{\mathsf{dl}} |\hat{\boldsymbol{h}}_i^H \hat{\boldsymbol{a}}_k|^2 + \sum_{i \in \mathcal{K}_1} p_i^{\mathsf{dl}} |\epsilon_i^H \hat{\boldsymbol{a}}_k|^2 + \sigma_k^{\mathsf{dl}}}, \quad (33)$$

where σ_k^{dl} represents the received noise power at UE k. In the P-CSI scenario, the error in the second term of the denominator is zero so this term does not exist. Assuming that ZF precoder is employed, the lower-bound ergodic downlink rate can be espressed as

$$\hat{r}_k^{\mathsf{dl,lb}} = \begin{cases} W \log_2\left(1 + p_k^{\mathsf{dl}}/\sigma_k^{\mathsf{dl}}\right) & \text{(P-CSI)} \\ W \log_2\left(1 + \frac{p_k^{\mathsf{dl}}}{\sum_{i \in \mathcal{K}_1} p_i^{\mathsf{dl}} \lambda_{k,i} + \sigma_k^{\mathsf{dl}}}\right) & \text{(IP-CSI).} \end{cases} \quad (34)$$

Then, the constraint (C2) on the total latency, which includes the computation time, the upload and download time, can be expressed as

$$\frac{T}{T-\tau}\left(\frac{b_k^{\mathsf{a}}}{\hat{r}_k^{\mathsf{lb}}} + \frac{b_k^{\mathsf{a,dl}}}{\hat{r}_k^{\mathsf{dl,lb}}}\right) + \frac{c_k^{\mathsf{a}}}{f_k^c} \leq \eta_k, \quad \forall k \in \mathcal{K}_1, \quad (35)$$

where $b_k^{\mathsf{a,dl}} = \sum_{l_k \in \mathcal{L}_k} (1 - s_{k,l_k}) b_{k,l_k}^{\mathsf{dl}}$.

On the other hand, the average transmit power of BS can be computed as

$$P_{\mathsf{BS}} = \sum_{k \in \mathcal{K}_1} p_k^{\mathsf{dl}} \mathbb{E}(\|\hat{\boldsymbol{a}}_k\|^2) = \begin{cases} \sum_{k \in \mathcal{K}_1} \frac{p_k^{\mathsf{dl}}}{(M - |\mathcal{K}|_1)\beta_k}, \text{ P-CSI} \\ \sum_{k \in \mathcal{K}_1} \frac{p_k^{\mathsf{dl}}(\tau p^{\mathsf{tr}} \beta_k + \sigma_{bs})}{(M - |\mathcal{K}|_1)\tau p^{\mathsf{tr}} \beta_k^2}, \text{ IP-CSI.} \end{cases}$$

The total transmit power at BS must be constrained by its maximum power $P_{\mathsf{max}}^{\mathsf{dl}}$, which can be expressed as

$$P_{\mathsf{BS}} \leq P_{\mathsf{max}}^{\mathsf{dl}}. \quad (36)$$

We can now formulate the joint computation offloading and resource allocation problem considering both uplink and downlink data transmissions as follows:

$$(\mathcal{P}_2^{\mathsf{ext}}) \min \xi \quad \text{s.t} \quad (\text{C0}), (\text{C3}), (\text{C4}), (\text{C6}), (20), (35), (36).$$

The difference between $(\mathcal{P}_2)$ and $(\mathcal{P}_2^{\mathsf{ext}})$ is in constraints (35) and (36). To tackle this difficult problem, we can again decompose it into two subproblems as in previous sections. In particular, we iteratively solve the (OP) subproblem (with constraints (C0), (C3), (C4), (20), (35)) to find the optimal $\boldsymbol{s}, \boldsymbol{f}^{\mathbf{c}}$ and solve the extended (PA) subproblem (with constraints (C0), (C4), (C6), (35), (36)) to find $\boldsymbol{p}, \boldsymbol{p}^{\mathbf{dl}}$. Because the optimization variables $\boldsymbol{p}, \boldsymbol{p}^{\mathsf{dl}}$ are only captured in the extended (PA) subproblem, we can solve the (OP) subproblem as in Section 3.2.1. We now discuss how to solve the extended (PA) subproblem which is stated as follows:

$$(\mathcal{P}_2^{\mathsf{PA,ext}}) \min \xi \quad \text{s.t} \quad (\text{C0}), (\text{C4}), (\text{C6}), (35), (36).$$

Considering the first delay term in (35), the two delay components, which correspond to the uplink transmission time of the incurred data and download transmission time of the computation outcome, respectively, have the same structure. Therefore, we can apply the same techniques as in the previous sections to deal with the downlink rate.

Specifically, we will apply the bisection search to find ξ for which we have to perform feasibility verification for a given value of ξ (to update the upper and lower bounds of ξ). For the P-CSI scenario, the non-convex constraint (C0) can be convexified by rewriting it as (C0′); therefore, the feasibility verification can be completed by using the CVX solver. For the IP-CSI scenario, to solve the (PA) subproblem, we employ the DC optimization technique as in Section 4 to deal with non-convex constraints involving both uplink rate and downlink rate.

5.2 Complexity Analysis

We analyze the computational complexity the complexity of the proposed algorithms in term of the number of required arithmetic operations. In Algorithm 1, the main complexity comes from the while-loop for the bisection search and the process of solving subproblem $(\mathcal{P}_3)_k$ in step 4. The bisection search of ξ requires $\log_2(\frac{\xi_{\max} - \xi_{\min}}{\epsilon})$ iterations. Besides, the Newton-Raphson search method to solve $g_k(p_k^\star) = 0$ typically converges within tens of iterations, denoted by N_1, and each iteration has complexity of $\mathcal{O}(1)$. Therefore, the computational complexity involved in solving subproblem $(\mathcal{P}_3)_k$ is $\mathcal{O}(2^{|\mathcal{L}_k|} N_1)$. Thus, the overall complexity of Algorithm 1 is $\mathcal{O}(\log_2(\frac{\xi_{\max} - \xi_{\min}}{\epsilon}) 2^{|\mathcal{L}_k^{\mathsf{m}}|} N_1 K)$, where $|\mathcal{L}_k^{\mathsf{m}}| = \max_k |\mathcal{L}_k|$.

For the (P-SO) algorithm (Algorithm 3) described in Section 3.2, we describe the worst-case complexity with exhaustive search using Method 2 to solve (OP) subproblem. The computational complexity involved in solving (OP) is $\mathcal{O}(2^{|\mathcal{L}_k^{\mathsf{m}}|} K)$ because the exhaustive search using Method 2 just needs to compute at the beginning of the bisection search. Therefore, the worst-case complexity of Algorithm 3 is $\mathcal{O}(N_2(2^{|\mathcal{L}_k^{\mathsf{m}}|} K + \log_2(\frac{\xi_{\max} - \xi_{\min}}{\epsilon}) N_1 K))$, where N_2 denotes the number of iterations required by two subproblems (OP) and (PA) to achieve convergence. As shown in the simulation result later, N_2 is typically no more than 6.[2]

For the IP-CSI scenario, the iterative process in Algorithm 4 converges in a few iterations, which is denoted as N_3. In each iteration, the convex problem $(\mathcal{P}_2^{\mathsf{PA}})^{(q)}$ and $(\mathcal{P}_2^{\mathsf{PA,ext}})^{(q)}$ can be solved by using the interior-point method with complexity $\mathcal{O}(m^{1/2}(m+n)n^2)$, where m denotes the number of inequality constraints, and n represents the number of variables [35]. Therefore, the complexity of solving $(\mathcal{P}_{2,\mathsf{IP}}^{\mathsf{PA}})$ and $(\mathcal{P}_{2,\mathsf{IP}}^{\mathsf{PA,ext}})$ are similar, which is equal to $\mathcal{O}(K^{3.5} N_3)$. Consequently, the overall complexity of solving $(\mathcal{P}_2)$ and $(\mathcal{P}_2^{\mathsf{ext}})$ in the IP-CSI scenario is $\mathcal{O}(N_2(2^{|\mathcal{L}_k^{\mathsf{m}}|} K + \log_2(\frac{\xi_{\max} - \xi_{\min}}{\epsilon}) K^{3.5} N_3))$.

6 Numerical Results

We consider an MEC system with the channel bandwidth of 10 MHz and $K = 20$ UEs randomly distributed in a cell coverage area with the radius of 900 m. All UEs are assumed to have the same maximum clock frequency of 2.4 GHz, the same maximum transmit power (i.e., $P_k = P_{\max}$), which is set equal to 0.22 (Watts) according to the 3GPP technical report [36] and the circuit power is set equal to 0.05 (Watts). In our simulation setting, all UEs have the same number of parallel tasks and the same total computation demand of

2. The complexity involved in solving the relaxed convex problem of $(\mathcal{P}_2^{\mathsf{OP}})_k$ is $\mathcal{O}(|\mathcal{L}_k|^{3.5})$ and our numerical studies suggest that the *average complexity* required to solve the original MINLP $(\mathcal{P}_2^{\mathsf{OP}})_k$ is approximately $\mathcal{O}(|\mathcal{L}_k|^{6.5})$, which is much smaller than its worst-case complexity.

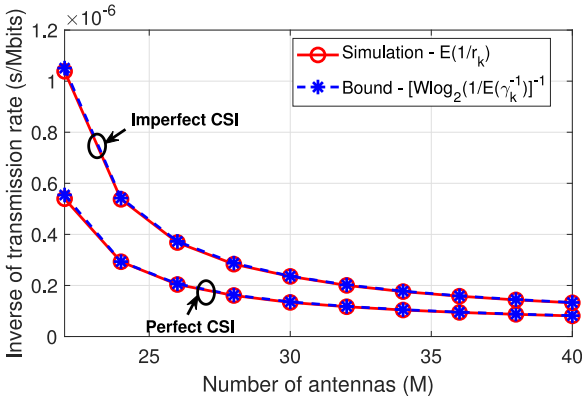

Fig. 1. Simulated and lower-bound of inverse rate.

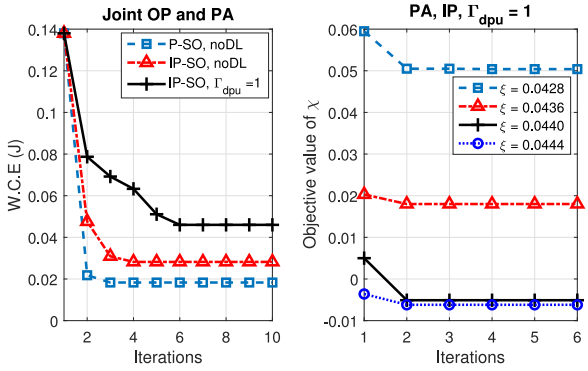

Fig. 2. Convergence of proposed algorithms.

0.24 Gcycles, but the number of CPU cycles per task is set randomly. The total number of transmission bits for all tasks is set to be the same for all UEs while the number of bits per task is generated randomly. For performance evaluation of the proposed design, we choose the ratio between the total number of transmission bits and the total required CPU cycles (BPC) to be about 4.2×10^{-3} (except for the results in Figs. 3 and 5), which is close to its highest possible value for the applications considered in [3].

The energy weights w_k are set equal to 1 for fair comparison with the no-computation-offload case. The energy coefficient is set as $\alpha_k = \alpha = 0.1 \times 10^{-27}$, which corresponds to the realistic measurement value in [37] and recent development in mobile chipset technology. The effect of this parameter on the performance of the offloading design will be clarified in Fig. 6. For all considered simulation scenarios, we set $|\mathcal{L}_k| = L = 5$ (except in Fig. 8), $M = 30$ (except in Fig. 1), $F^{\mathrm{c}} = 40$ GHz (except in Fig. 6), maximum allowable delay $\eta_k = \eta$ for all UEs, and $P^{\mathrm{dl}}_{\mathrm{max}} = 10$ (Watts).

The noise powers at the mobile user side and BS are given as $\sigma_{bs} = \sigma^{\mathrm{dl}}_k$ = bandwidth $\times\, k_B \times T_0 \times$ noise figure (W), where $k_B = 1.381 \times 10^{-23}$ (Joule per Kelvin) is the Boltzmann constant, $T_0 = 290$ (Kelvin) is the noise temperature, and noise figure = 0.9 (Watts). The small scale channel fading coefficient is generated according to the Rayleigh distribution and the path-loss is defined according to 3GPP technical report as $\beta_k\ (\mathrm{dB}) = 128.1 + 37.6 \log_{10}(d_k)$ where d_k is the geographical distance between UE k and the BS (in km) [38]. In the IP-CSI scenario, we take $T = 200$ symbols, which corresponds to the coherence bandwidth of 200 kHz and a coherence time of 1 (ms). The simulation results are obtained by averaging the results over 100 realizations except for Figs. 1 and 4.

Fig. 1 compares the upper-bound of the inverse of transmission rate given in the right-hand-side of (10) and its simulated values obtained by averaging over 10000 realizations in both P-CSI and IP-CSI scenarios when the transmit power $p_k = P_{\max}/2$ is set equally for all UEs. This figure confirms that the value of ergodic inverse of transmission rate is indeed close to its upper-bound and the gap between them is negligible when the number of antennas at the BS M becomes relatively large compared to the number of UEs K. In fact, when the SINR is good enough to support the data transmission required by the offloading process, as shown in (9), the second-order derivative of the inverse of the transmission rate is nearly equal to zero. Therefore, the equality condition of the Jensen's inequality in (10) holds with high probability.

The convergence of the proposed low-complexity algorithms based on bisection search for both P-CSI scenario (P-SO) and IP-CSI scenario with or without downlink transmission (IP-SO) is illustrated in Fig. 2. Specifically, we show the variations of the maximum W.C.E (ξ) over iterations in the left subfigure where no-downlink transmission is indicated as noDL and with-downlink transmission is shown together with the ratio between downlink data size and uplink data size $\Gamma_{\mathrm{dpu}} = 1$. The right subfigure shows the variations of χ in (32) over iterations for different values of ξ, which is used to perform feasibility verification in the bisection search in the (PA) subproblem. It can be seen that χ converges to a negative value for some values of ξ, which indicates the feasibility condition.

The benefit of joint optimization of radio and computing resource allocation in the computation offloading design is illustrated in Fig. 3 for varying maximum allowable delay η. In this figure, considering no downlink data transmission and P-CSI scenario, we compare the achievable performance in four scenarios: task processing at mobile devices ('No-offload'), partial offloading with optimal offloading decision and cloud-resource allocation with fixed transmit power for all UEs $p_k = P_{\max}/2$ and $p_k = P_{\max}$, and with optimal transmit power allocation ('Optimal p_k'). The left and right subfigures show the achieved min-max W.C.E for different values of transmission bits per CPU cycle (BPC). From this figure, we can see that the minimum required latency that the mobile device can process its tasks locally

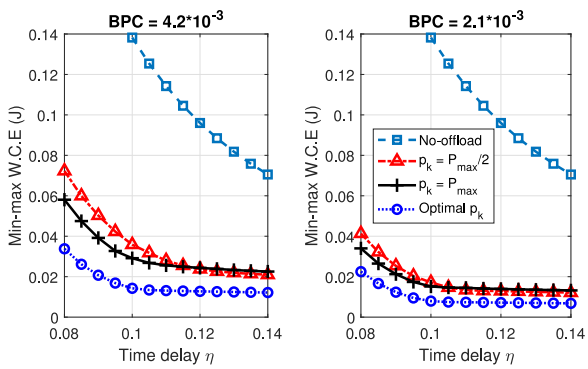

Fig. 3. Performance comparison of with/without offloading and with/without optimization of radio and computing resource.

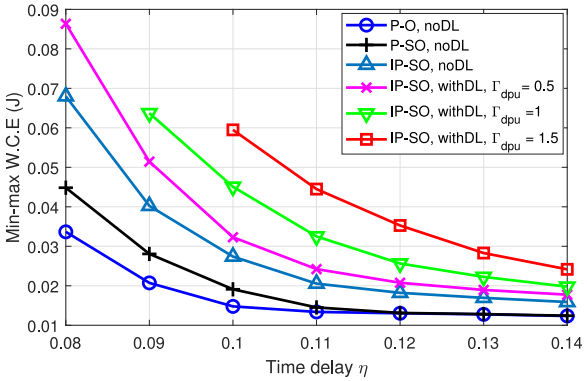

Fig. 4. Min-max W.C.E versus maximum allowable latency.

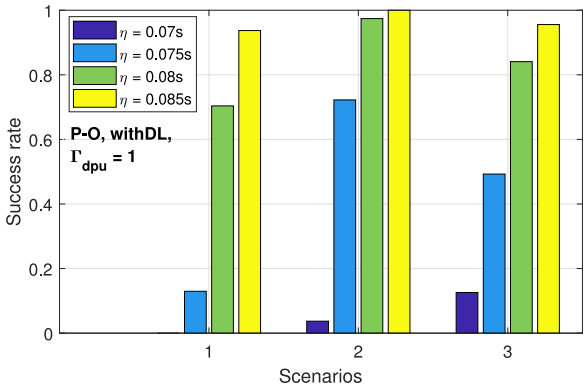

Fig. 5. Success rate for task processing with $\Gamma_{\text{dpu}} = 1$.

('No-offload' case) is 0.1s while the minimum required latency in the remaining cases are 0.08 s. This means that computation offloading allows mobile devices to achieve lower latency. Moreover, the consumed energy in the partial offloading scheme is significantly smaller than that in the 'No-offload' case. For instance, the min-max W.C.E at $\eta = 0.1\,\text{s}$ in the left subfigure is equal to 0.138, 0.036, 0.029, 0.014 for the 'No-offload', fixed transmit power of '$p_k = P_{\max}/2$', '$p_k = P_{\max}$' and 'Optimal p_k', respectively. This means that partial offloading enables to save about 5 times of energy with no optimization of the transmit power and save about 10 times of energy with optimal transmit power. Moreover, the difference in the consumed energy among the offloading and no-offloading schemes becomes larger for smaller number of transmission bits.

Fig. 4 presents the achieved performance of different design scenarios considered in this paper: optimal solution with P-CSI—no downlink data ('P-O, noDL'), solution with P-CSI—no downlink data ('P-SO, noDL'), and IP-CSI—no downlink data ('IP-SO, noDL'). We also consider different application scenarios with small, medium and large amount of downlink data in comparison with amount of uplink data where the performance of our low-complexity algorithm for the IP-CSI scenario is investigated. Specifically, we set the ratio between the amount of downlink data and the amount of uplink data (Γ_{dpu}) (i.e., computed as $\sum_{l_k \in \mathcal{L}_k} b^{\text{dl}}_{k,l_k} / \sum_{l_k \in \mathcal{L}_k} b_{k,l_k}$) equal to 0.5, 1, and 1.5 corresponding to low downlink data ('$\Gamma_{\text{dpu}} = 0.5$'), medium downlink data ('$\Gamma_{\text{dpu}} = 1$'), and large downlink data ('$\Gamma_{\text{dpu}} = 1.5$'), respectively.

It can be observed from this figure that the low-complexity algorithm achieves close-to-optimal performance when the maximum delay constraint is less stringent ('blue' and 'black' curves). For the IP-CSI scenario, mobile users will require more energy for data transmission to compensate for the CSI estimation errors. When the amount of downlink data becomes larger, more time is required to transfer the download data which means that less time is available for uploading the uplink data and computation at the cloud server. In some cases, increasing the transmit power to its maximum value may not lead to improved SINR, and the low transmission rate may prevent successful uplink data transmission in the offloading process. In all studied scenarios, even for the high value of Γ_{dpu}, the partial offloading scheme enables us to save energy significantly.

Fig. 5 shows the success rate for which computation task processing can be completed successfully in the MEC system with limited computing resource and three different scenarios: 1) $\sum_{l_k \in \mathcal{L}_k} b_{k,l_k} = 1$ Mbits, $F^{\text{c}} = 40$ GHz; 2) $\sum_{l_k \in \mathcal{L}_k} b_{k,l_k} = 0.75$ Mbits, $F^{\text{c}} = 40$ GHz, and 3) $\sum_{l_k \in \mathcal{L}_k} b_{k,l_k} = 1$ Mbits, $F^{\text{c}} = 80$ GHz. The success rate is obtained by calculating the ratio between the number of successful computations and the total 200 different realizations. The results are obtained for the optimal algorithm, P-CSI and no downlink data transmission. We can see from this figure that, with the same requirement on computation, the larger amount of computing resource available at the cloud server and the smaller amount of uplink data, the better performance (i.e., lower latency and higher success rate) that mobile users can achieve through computation offloading.

Fig. 6 illustrates the offloading performance gain versus the energy coefficient of mobile devices. This performance gain is computed as $(\xi^{\text{no-offload}} - \xi)/\xi$, which is used to compare the relative difference between the offloading and no-offloading cases. As shown in Fig. 6, larger performance gain can be obtained for applications with more stringent delay requirement. Moreover, the resource-rich cloud can lead to a significant performance gain for low-cost devices equipped with chipsets having higher coefficient α.

Fig. 7 presents the ratio between the energy components due to transmission and local processing ($\xi^{\text{t}}/\xi^{\text{lo}}$) for different values of Γ_{dpu}, where $\xi^{\text{t}} = \frac{1}{K}\sum_k \xi^{\text{t}}_k$ and $\xi^{\text{lo}} = \frac{1}{K}\sum_k \xi^{\text{lo}}_k$. It can be observed that more energy will be needed for transmitting data when the maximum allowable delay is smaller; otherwise, more energy will be used for local computation.

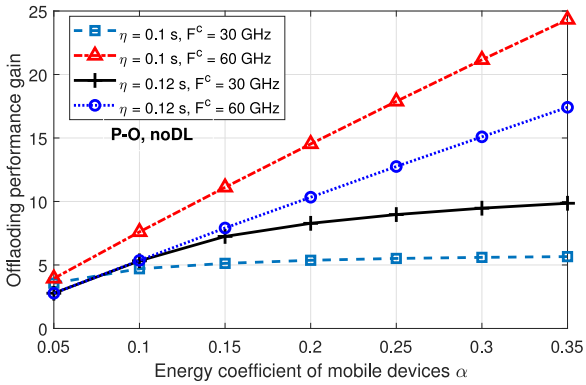

Fig. 6. Performance gain versus energy coefficient of mobile device.

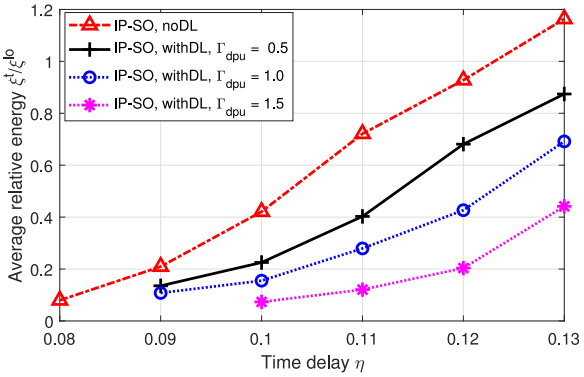

Fig. 7. Average ratio of energy components versus time delay.

In fact, the previous figure suggests that if the system has enough resource to guarantee the successful data transmission and remote computation, computation offloading is preferred to save energy. Therefore, larger number of tasks can be processed remotely and less energy is used for executing tasks locally for larger maximum allowable delay. It means that the proportion of local computation energy will becomes smaller. Moreover, when the required transmit energy is smaller than ones for local computation, the increase in the transmit energy to compensate for the download time will become smaller than the increase in the energy required for computing more tasks, which cannot be offloaded with larger Γ_{dpu}. Thus, the relative energy ratio ξ_t/ξ^{lo} will also decrease.

The achieved W.C.E versus allowable delay is illustrated in Fig. 8 for different number of parallel tasks per UE L. It can be observed that the min-max W.C.E decreases quite drastically as the number of tasks increases, especially in the regime with a small number of parallel tasks. When L increases to a sufficiently large value for which the radio and computing resources can be effectively allocated to all UEs, the difference in performance due to increasing L will become insignificant.

Fig. 9 shows the fairness achieved for different UEs when applying the proposed min-max based computation offloading strategy in the IP-CSI scenario with no download data and $\eta = 0.1$ s. On the average, each UE offloads more than half of its required computation demand and the resulting consumed energy is fairly similar among the UEs. Furthermore, the average weighted consumed energy for each user is quite smaller than average min-max W.C.E (shown in Fig. 4).

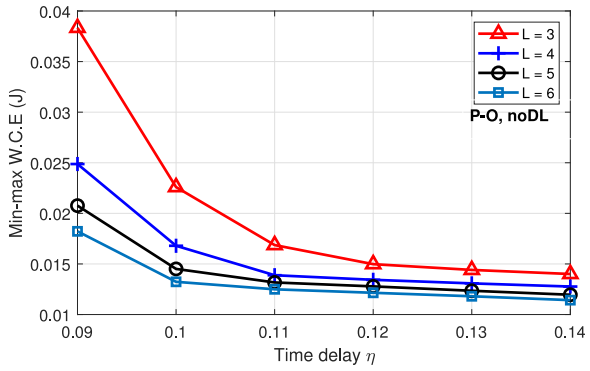

Fig. 8. Performance with difference number of parallel tasks.

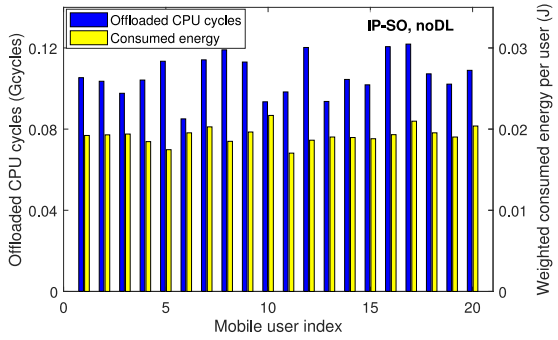

Fig. 9. Computation allocation and total consumed energy with allowable latency of 0.1 s.

This is reasonable for the underlying min-max strategy because the min-max W.C.E value corresponds to the highest weighted consumed energy among all users in the system.

7 Conclusion

In this paper, we have developed both optimal and low-complexity algorithms to tackle general joint computation offloading and resource allocation for the MIMO based mobile cloud computing system considering P-CSI and IP-CSI. Our numerical studies have confirmed that significant energy saving of the proposed design compared to the no-offloading scenario can be achieved; the proposed sub-optimal algorithm achieves close to optimal performance when the delay constraint is not very stringent; and our proposed designs can provide great fairness for different users.

In the current work, we assume that different user's tasks are independent. We plan to address the offloading scenario for the MIMO-based MEC system where user's tasks are dependent in our future work. Furthermore, offloading design for the MIMO-based multi-task multi-user setting in the hierarchical fog-cloud system will be considered.

Appendix A
Proof of Proposition 3

It can be verified that the transmission time in $(\text{C2})_k$ decreases with p_k. Therefore, f_k^{c} achieves its minimum value when p_k is equal to its largest possible value. Moreover, f_k^{c} can be greater than zero if and only if the minimum transmission time is less than the maximum allowable delay, i.e., $\frac{b_k^{\text{a}}}{W\log_2(1+P_k\beta_k^{\text{a}})} < \eta_k$. If $g_k(P_k)$ is less than or equal to zero, all constraints are satisfied at P_k and the objective function achieves its minimum value. Otherwise, we will consider the case where $g_k(P_k) > 0$. The constraints $(\text{C2})_k$ and $(\text{C6})_k$ can be rewritten as $p_k^{\text{a}} < p_k < P_k$. On the other hand, it can be verified that $g_k(p_k)$ is a convex function of p_k. Therefore, from the Karush Kuhn Tucker (KKT) conditions which can be used to find the maximum value of p_k satisfying constraint $(\text{C0}')_k$ and $p_k^{\text{a}} < p_k^{\star} < P_k$, we can deduce that the optimal $p_k^{\star}$ must satisfy $g_k(p_k^{\star}) = 0$.

Appendix B
Proof of Proposition 5

For given values of ξ and f_k^{c}, constraints (C0) and (C2) can be rewritten as

$$\hat{r}_k^{\mathsf{lb}} = W\log_2(p_k + \boldsymbol{p}^{\mathsf{t}}\boldsymbol{\lambda}_k + \sigma_k) - W\log_2(\boldsymbol{p}^{\mathsf{t}}\boldsymbol{\lambda}_k + \sigma_k)$$
$$\geq \max\left(\left(\frac{T}{T-\tau}\right)\frac{b_k^{\mathsf{a}}}{\eta_k - \frac{c_k^{\mathsf{a}}}{f_k^{\mathsf{c}}}}, \left(\frac{\tau(p^{\mathsf{tr}} + p_{k,c})}{T-\tau} + p_k + p_{k,c}\right)\frac{b_k^{\mathsf{a}}}{\frac{\xi}{w_k} - \xi_k^{\mathsf{lo}}}\right). \tag{37}$$

The left-hand-side of (37) is the sum of concave and convex functions; hence, it is a sigmoidal function. Consequently, $(\mathcal{P}_2^{\mathsf{PA}})$ with given values of ξ and f_k^{c} is a sigmoidal program, which is NP-hard and NP-hard to approximate [39]. Thus, the original subproblem $(\mathcal{P}_2^{\mathsf{PA}})$ is also NP-hard.

Appendix C
Proof of Proposition 6

Let $g_{1,k}^{(q)}(\boldsymbol{p}) = p_k + p_{k,c} + \frac{\tau(p^{\mathsf{tr}} + p_{k,c})}{T-\tau} - \xi^{\mathsf{a}}\tilde{r}_k^{\mathsf{lb}}(\boldsymbol{p}|\boldsymbol{p}^{(q)})$ and $g_{2,k}^{(q)}(\boldsymbol{p}, f_k^{\mathsf{c}}) = (\frac{T}{T-\tau})\frac{b_k^{\mathsf{a}}}{\tilde{r}_k^{\mathsf{lb}}(\boldsymbol{p}|\boldsymbol{p}^{(q)})} + \frac{c_k^{\mathsf{a}}}{f_k^{\mathsf{c}}} - \eta_k$. We have

$$\begin{aligned}
\chi^{(q)} &= \max_k \max\{g_{1,k}^{(q)}(\boldsymbol{p}^{(q)}), g_{2,k}^{(q)}(\boldsymbol{p}^{(q)}, (f_k^{\mathsf{c}})^{(q)}\} \\
&\geq \min_{\boldsymbol{p}} \max_k \max\{g_{1,k}^{(q)}(\boldsymbol{p}), g_{2,k}^{(q)}(\boldsymbol{p}, (f_k^{\mathsf{c}})^{(q)})\} \\
&= \max_k \max\{g_{1,k}^{(q)}(\boldsymbol{p}^{(q+1)}), g_{2,k}^{(q)}(\boldsymbol{p}^{(q+1)}, (f_k^{\mathsf{c}})^{(q)})\} \\
&\overset{(a)}{\geq} \max_k \max\{g_{1,k}^{(q+1)}(\boldsymbol{p}^{(q+1)}), g_{2,k}^{(q+1)}(\boldsymbol{p}^{(q+1)}, (f_k^{\mathsf{c}})^{(q)})\} \\
&\geq \min_{f_k^{\mathsf{c}}} \max_k \max\{g_{1,k}^{(q+1)}(\boldsymbol{p}^{(q+1)}), g_{2,k}^{(q+1)}(\boldsymbol{p}^{(q+1)}, f_k^{\mathsf{c}})\} \\
&= \max_k \max\{g_{1,k}^{(q+1)}(\boldsymbol{p}^{(q+1)}), g_{2,k}^{(q+1)}(\boldsymbol{p}^{(q+1)}, (f_k^{\mathsf{c}})^{(q+1)})\} \\
&= \chi^{(q+1)},
\end{aligned} \tag{38}$$

where $\boldsymbol{p}^{(q+1)} = \operatorname{argmin}_{\boldsymbol{p}} \max_k \max\{g_{1,k}^{(q)}(\boldsymbol{p}), g_{2,k}^{(q)}(\boldsymbol{p}, (f_k^{\mathsf{c}})^{(q)})\}$ and $(f_k^{\mathsf{c}})^{(q+1)} = \operatorname{argmin}_{f_k^{\mathsf{c}}} \max_k \max\{g_{1,k}^{(q+1)}(\boldsymbol{p}^{(q+1)}), g_{2,k}^{(q+1)}(\boldsymbol{p}^{(q+1)}, f_k^{\mathsf{c}})\}$. Inequality (a) holds since $\tilde{r}_k^{\mathsf{lb}}(\boldsymbol{p}|\boldsymbol{p}^{(q)}) \leq \tilde{r}_k^{\mathsf{lb}}(\boldsymbol{p}^{(q+1)}|\boldsymbol{p}^{(q)}) \leq \hat{r}_k^{\mathsf{lb}}(\boldsymbol{p}^{(q+1)}) = \tilde{r}_k^{\mathsf{lb}}(\boldsymbol{p}^{(q+1)}|\boldsymbol{p}^{(q+1)})$. Therefore, for a given ξ, using D.C to approximate transmission rate creates a sequence of feasible and improving solutions for $(\mathcal{P}_{2,\mathsf{IP}}^{\mathsf{PA}})$, which, therefore, converges.

Acknowledgments

Preliminary results of this paper have been published at IEEE ICC 2018. This research is funded by Vietnam National University HoChiMinh City (VNU-HCM) under grant number B2018-26-01.

References

[1] D. Kong, "Science driven innovations powering mobile product: Cloud AI vs. device AI solutions on smart device," (2017). [Online]. Available: arXiv:1711.07580

[2] S. Rallapalli, H. Qiu, A. Bency, S. Karthikeyan, R. Govindan, B. Manjunath, and R. Urgaonkar, "Are very deep neural networks feasible on mobile devices," in *Proc. ACM HotMobile Workshop*, pp. 1–7, 2016.

[3] J. Kwak, Y. Kim, J. Lee, and S. Chong, "DREAM: Dynamic resource and task allocation for energy minimization in mobile cloud systems," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 12, pp. 2510–2523, Dec. 2015.

[4] R. Van Noorden, "A better battery," *Nature News*, vol. 507, no. 7490, pp. 26–28, 2014.

[5] S. Barbarossa, S. Sardellitti, and P. Di Lorenzo, "Communicating while computing: Distributed mobile cloud computing over 5G heterogeneous networks," *IEEE Signal Process. Mag.*, vol. 31, no. 6, pp. 45–55, Nov. 2014.

[6] 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, Aug. 2017.

[7] H. Flores and S. N. Srirama, "Mobile cloud middleware," *J. Syst. Softw.*, vol. 92, no. 1, pp. 82–94, Jun. 2014.

[8] E. Cuervo, A. Balasubramanian, D. Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl, "MAUI: Making smartphones last longer with code offload," in *Proc. ACM 8th Int. Conf. Mobile Syst. Appl. Services*, 2010, pp. 49–62.

[9] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Trans. Netw.*, vol. 24, no. 5, pp. 2795–2808, Oct. 2016.

[10] H. Wu, Y. Sun, and K. Wolter, "Energy-efficient decision making for mobile cloud offloading," *IEEE Trans. Cloud Comp.*, vol. PP, no. 99, p. 1, Jan. 2018.

[11] K. Wang, K. Yang, C. Pan, and J. Wang, "Joint offloading framework to support communication and computation cooperation," (2017). [Online]. Available: arXiv:1705.10384

[12] W. Zhang, Y. Wen, K. Guan, D. Kilper, H. Luo, and D. O. Wu, "Energy-optimal mobile cloud computing under stochastic wireless channel," *IEEE Trans. Wireless Commun.*, vol. 12, no. 9, pp. 4569–4581, Sep. 2013.

[13] X. Xiang, C. Lin, and X. Chen, "Energy-efficient link selection and transmission scheduling in mobile cloud computing," *IEEE Wireless Commun. Lett.*, vol. 3, no. 2, pp. 153–156, Apr. 2014.

[14] D. Huang, P. Wang, and D. Niyato, "A dynamic offloading algorithm for mobile computing," *IEEE Trans. Wireless Commun.*, vol. 11, no. 6, pp. 1991–1995, Jun. 2012.

[15] W. Zhang, Y. Wen, and D. O. Wu, "Collaborative task execution in mobile cloud computing under a stochastic wireless channel," *IEEE Trans. Wireless Commun.*, vol. 14, no. 1, pp. 81–93, Jan. 2015.

[16] C. You, K. Huang, H. Chae, and B. Kim, "Energy-efficient resource allocation for mobile-edge computation offloading," *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1397–1411, Mar. 2017.

[17] L. Pu, X. Chen, J. Xu, and X. Fu, "D2D fogging: An energy-efficient and incentive-aware task offloading framework via network-assisted D2D collaboration," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3887–3901, Dec. 2016.

[18] J. Du, L. Zhao, J. Feng, and X. Chu, "Computation offloading and resource allocation in mixed fog/cloud computing systems with min-max fairness guarantee," *IEEE Trans. Commun.*, vol. 66, no. 4, pp. 1594–1608, Apr. 2018.

[19] C. Wang, F. R. Yu, C. Liang, Q. Chen, and L. Tang, "Joint computation offloading and interference management in wireless cellular networks with mobile edge computing," *IEEE Trans. Veh. Tech.*, vol. 66, no. 8, pp. 7432–7445, Aug. 2017.

[20] K. Zhang, Y. Mao, S. Leng, Q. Zhao, L. Li, X. Peng, L. Pan, S. Maharjan, and Y. Zhang, "Energy-efficient offloading for mobile edge computing in 5G heterogeneous networks," *IEEE Access*, vol. 4, pp. 5896–5907, Aug. 2016.

[21] Y. Mao, J. Zhang, S. Song, and K. B. Letaief, "Stochastic joint radio and computational resource management for multi-user mobile-edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 16, no. 9, pp. 5994–6009, Sep. 2017.

[22] S. Sardellittia, G. Scutari, and S. Barbarossa, "Joint optimization of radio and computational resources for multicell mobile-edge computing," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 1, no. 2, pp. 89–103, Jun. 2015.

[23] A. Al-Shuwaili, O. Simeone, A. Bagheri, and G. Scutari, "Joint uplink/downlink optimization for backhaul-limited mobile cloud computing with user scheduling," *IEEE Trans. Sig. Inf. Process. Netw.*, vol. 3, no. 4, pp. 787–802, Dec. 2017.

[24] F. Wang, J. Xu, X. Wang, and S. Cui, "Joint offloading and computing optimization in wireless powered mobile-edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 1784–1797, Mar. 2018.

[25] E. Hossain, M. Rasti, and L. B. Le, *Radio Resource Management in Wireless Networks: An Engineering Approach*, Cambridge, U.K.: Cambridge Univ. Press, 2017.

[26] Y. Geng, Y. Yang, and G. Cao, "Energy-efficient computation offloading for multicore-based mobile devices," in *Proc. IEEE Conf. Comput. Commun.*, 2018, pp. 46–54.

[27] M.-H. Chen, M. Dong, and B. Liang, "Resource sharing of a computing access point for multi-user mobile cloud offloading with delay constraints," *IEEE Trans. Mobile Comput.*, vol. 17, no. 12, pp. 2868–2881, Dec. 2018.
[28] Y. Dai, D. Xu, S. Maharjan, and Y. Zhang, "Joint computation offloading and user association in multi-task mobile edge computing," *IEEE Trans. Veh. Tech.*, vol. 67, no. 12, pp. 12313–12325, Dec. 2018.
[29] S. E. Mahmoodi, R. Uma, and K. Subbalakshmi, "Optimal joint scheduling and cloud offloading for mobile applications," *IEEE Trans. Cloud Comp.*, vol. PP, no. 99, p. 1, Apr. 2016.
[30] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 568–576.
[31] H. Q. Ngo, E. G. Larsson, and T. L. Marzetta, "Energy and spectral efficiency of very large multiuser MIMO systems," *IEEE Trans. Commun.*, vol. 61, no. 4, pp. 1436–1449, Apr. 2013.
[32] B. V. Patil, P. S. Nataraj, and S. Bhartiya, "Global optimization of mixed-integer nonlinear (polynomial) programming problems: The Bernstein polynomial approach," *Springer Comput.*, vol. 94, no. 2–4, pp. 325–343, Mar. 2012.
[33] H. H. Kha, H. D. Tuan, and H. H. Nguyen, "Fast global optimal power allocation in wireless networks by local DC programming," *IEEE Trans. Wireless Commun.*, vol. 11, no. 2, pp. 510–515, Feb. 2012.
[34] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[35] A. Nemirovski, "Interior point polynomial time methods in convex programming," *Lecture Notes*, (2004). [Online]. Available: https://www2.isye.gatech.edu/~nemirovs/Lect_IPM.pdf
[36] LTE; Evolved universal terrestrial radio access (E-UTRA); user equipment (UE) radio transmission and reception, document (3GPP TS 36.101 version 13.2.1 Release 13), 3GPP, May 2016.
[37] A. P. Miettinen and J. K. Nurminen, "Energy efficiency of mobile clients in cloud computing," in *Proc. USENIX Conf. Hot Topics Cloud Compt.*, 2010, pp. 4–11.
[38] Evolved universal terrestrial radio access (E-UTRA); further advancements for E-UTRA physical layer aspects, document (3GPP TR 36.814 version 9.0.0 Release 9), 3GPP, Mar. 2010.
[39] Y. Li, M. Sheng, X. Wang, Y. Zhang, and J. Wen, "Max-min energy-efficient power allocation in interference-limited wireless networks," *IEEE Trans. Veh. Tech.*, vol. 64, no. 9, pp. 4321–4326, Sep. 2015.

Ti Ti Nguyen received the BEng degree in electrical engineering from the Ho Chi Minh City University of Technology, Vietnam, in 2013, the MEng degree in embedded system from the University of Rennes 1, France, in 2015. He is currently working toward the PhD degree at INRS, University of Quebec, Canada. His current research interests include mobile edge computing and radio resource management. He is a student member of the IEEE.

Long Bao Le (S'04-M'07-SM'12) received the BEng degree in electrical engineering from Ho Chi Minh City University of Technology, Vietnam, in 1999, the MEng degree in telecommunications from Asian Institute of Technology, Thailand, in 2002, and the PhD degree in electrical engineering from the University of Manitoba, Canada, in 2007. Since 2010, he has been with the Institut National de la Recherche Scientifique (INRS), Université du Québec, Montréal, QC, Canada where he is currently an associate professor. His current research interests include smartgrids, cognitive radio, radio resource management, network control and optimization, and emerging enabling technologies for 5G wireless systems. He is a senior member of the IEEE.

Quan Le-Trung received the bachelor of engineering (BEng) degree from the Bach-Khoa University of Technology (BKU), Vietnam, in 1998, the master of engineering (MEng) degree from the Asian Institute of Technology, in Thailand, in 2002, and the doctor of technical science (Dr. techn.) degree from the Department of Telecooperation, Johannes Kepler University-Linz, Austria, in 2007, then he has spent 9 months postdoc there, and another 24 months postdoc at Networks & Distributed Systems group, Department of Informatics, University of Oslo, Norway. He is currently an associate professor position at the Department of Computer Networks, University of Information Technology - VNUHCM. His research interests span a variety of areas in networking with current focus on the Wireless Embedded Internet, Cyber Physical Systems, Internet of Things, Cloud Computing.