Latency Minimization for Intelligent Reflecting Surface Aided Mobile Edge Computing

Tong Bai, *Member, IEEE*, Cunhua Pan, *Member, IEEE*, Yansha Deng, *Member, IEEE*, Maged Elkashlan, *Member, IEEE*, Arumugam Nallanathan, *Fellow, IEEE* and Lajos Hanzo, *Fellow, IEEE*

***Abstract*—Computation off-loading in mobile edge computing (MEC) systems constitutes an efficient paradigm of supporting resource-intensive applications on mobile devices. However, the benefit of MEC cannot be fully exploited, when the communications link used for off-loading computational tasks is hostile. Fortunately, the propagation-induced impairments may be mitigated by intelligent reflecting surfaces (IRS), which are capable of enhancing both the spectral- and energy-efficiency. Specifically, an IRS comprises an IRS controller and a large number of passive reflecting elements, each of which may impose a phase shift on the incident signal, thus collaboratively improving the propagation environment. In this paper, the beneficial role of IRSs is investigated in MEC systems, where single-antenna devices may opt for off-loading a fraction of their computational tasks to the edge computing node via a multi-antenna access point with the aid of an IRS. Pertinent latency-minimization problems are formulated for both single-device and multi-device scenarios, subject to practical constraints imposed on both the edge computing capability and the IRS phase shift design. To solve this problem, the block coordinate descent (BCD) technique is invoked to decouple the original problem into two subproblems, and then the computing and communications settings are alternatively optimized using low-complexity iterative algorithms. It is demonstrated that our IRS-aided MEC system is capable of significantly outperforming the conventional MEC system operating without IRSs. Quantitatively, about 20 % computational latency reduction is achieved over the conventional MEC system in a single cell of a 300 m radius and 5 active devices, relying on a 5-antenna access point.**

***Index Terms*—Intelligent reflecting surface, mobile edge computing, latency minimization.**

I. Introduction

A. Motivation and Scope

In the Internet-of-Things (IoT) era, myriads of machines and sensors are envisioned to be connected [1]. However, since these devices typically have limited computing capabilities, resource-intensive applications cannot be readily supported by these devices due to their resultant excessive computational latency. Aimed at tackling this issue, powerful computing nodes can be deployed at the edge of the network (typically co-located with the access points (APs)) [2]. As a benefit, the computational latency of these resource-intensive applications can be reduced, by employing both local computing on the devices and edge computing for processing these computational tasks, provided that these tasks can be successfully off-loaded. This paradigm is referred to as mobile edge computing (MEC) [3]–[14]. At the time of writing, the potential of this MEC paradigm has not been fully exploited, predominantly because the computation off-loading link is far from perfect. For example, the devices located at the cell edge typically suffer from a low off-loading success rate, and/or their computation off-loading may impose higher latency than computing their tasks locally. Hence these devices have to rely on their own computing resources, which is however often incapable of supporting resource-intensive applications. Therefore, it is imperative to improve the performance of MEC systems from a communications perspective.

The recent advances in programmable meta-materials [15] facilitate the construction of intelligent reflecting surfaces (IRSs) [16] for enhancing both the spectral- and energy-efficiency of wireless communications. Specifically, an IRS is comprised of an IRS controller and a large number of passive reflecting elements. Under the instructions of the IRS controller, each IRS reflecting element is capable of adjusting both the amplitude and the phase of the signals reflected, thus collaboratively modifying the signal propagation environment. The gain attained by IRSs is based on the combination of both the virtual array gain and the reflection-aided beamforming gain. To elaborate, the virtual array gain can be achieved by combining both the direct and IRS-reflected signals, while the reflection-aided beamforming gain is realized by proactively controlling the phase shift induced by the IRS elements. By beneficially combining these two types of gains, the IRS becomes capable of boosting the devices' off-loading success rate, hence improving the potential of MEC systems. In this treatise, our attention is focused on investigating the role of IRSs in MEC systems.

B. Related Works

1) Design of Mobile Edge Computing Systems: At the current state-of-the-art, MEC systems can be categorized into [5]: single-user [6]–[9] and multi-user systems [10]–[14]. Among the design metrics of single-user MEC systems, the

L. Hanzo would like to acknowledge the financial support of the Engineering and Physical Sciences Research Council projects EP/N004558/1, EP/P034284/1, EP/P003990/1 (COALESCE), of the Royal Society's Global Challenges Research Fund Grant as well as of the European Research Council's Advanced Fellow Grant QuantCom.

T. Bai, C. Pan, M. Elkashlan, and A. Nallanathan are with the School of Electronic Engineering and Computer Science, Queen Mary University of London, London, E1 4NS, U.K. (e-mail: t.bai@qmul.ac.uk, c.pan@qmul.ac.uk, maged.elkashlan@qmul.ac.uk, a.nallanathan@qmul.ac.uk).

Y. Deng is with the Department of Engineering, King's College London, London, WC2R 2LS, U.K. (e-mail: yansha.deng@kcl.ac.uk).

L. Hanzo is with the School of Electronics and Computer Science, University of Southampton, Southampton, SO17 1BJ, U.K. (e-mail: lh@ecs.soton.ac.uk).

computation off-loading strategy plays a crucial role. More explicitly, the binary off-loading strategy of [6] was proposed to decide whether the task is executed locally at the mobile device or remotely at the edge-cloud node. By contrast, Wang *et al.* [7] conceived a partial off-loading scheme for data-partitioning oriented applications, where a fraction of the data can be processed at the mobile device, while the rest at the edge. However, in realistic multi-user systems, inter-user interference is imposed both on the radio communications link and on the computing node at the edge, which may erode the overall performance of the MEC system. In order to cope with this hindrance, Sardellitti *et al.* [10] jointly optimized the transmit precoding matrices and the computational resources allocated to each user in a multi-cell multi-user scenario, while Sheng *et al.* [17] proposed an energy-efficient algorithm to optimize the resource allocation of terminals, radio access networks, and edge servers in a multi-carrier scenario. For the system where the devices have to make their off-loading decisions locally, Chen *et al.* [11] provided a distributed joint computation off-loading and channel selection policy relying on classic game theory. Recently, a specific user association scheme was also developed for multi-user systems served by multiple edge computing nodes [13], while a mobility-aware dynamic service scheduling algorithm was proposed for MEC systems [14]. Furthermore, Yang *et al.* [18] conceived an over-the-air computation aided federated learning algorithm for reducing both the latency and the power consumption, and for preserving the users' privacy in MEC systems. At the time of writing, the computation offloading issue of the devices in the face of hostile communications environments has not been well addressed. Against this background, in this paper the performance is improved by invoking IRSs. Let us now continue by reviewing the relevant research contributions on IRSs as follows.

2) Intelligent Reflecting Surface Aided Wireless Networks: In order to explore the benefits of IRSs in wireless communications, extensive research efforts have been invested into their ergodic capacity analysis [19], channel estimation [20], and practical reflection phase shift modeling [21], as well as into the associated phase shift design [22]–[29]. Specifically, a joint design of the IRS phase shift and of the precoding at the AP was proposed for minimizing the transmit power, while maintaining the target receive signal-to-interference-plus-noise ratio (SINR) [22], relying on the sophisticated techniques of the semidefinite relaxation and of alternating optimization. These investigations were then extended to the more practical discrete phase shift setting [23]. However, the excessive computational complexity of the algorithm developed in [22] prohibits its application in large-scale IRSs. In order to reduce the complexity, Guo *et al.* [24] proposed three low-complexity algorithms, while Pan *et al.* [26] provided a pair of majorization-minimization (MM) algorithms and complex circle manifold methods for multi-cell scenarios. Furthermore, in order to reduce the overhead during the IRS channel estimation, Yang *et al.* [29] grouped the IRS elements, where each group shares the same phase shift coefficient, and optimized the power allocation and phase shift in orthogonal frequency division multiplexing (OFDM)-based wireless systems. Apart from the conventional communications scenarios, the role of IRSs was also investigated both in terms of improving physical-layer security [30]–[33], and simultaneous wireless information and power transfer (SWIPT) [34], [35], where substantial gains were achieved. These impressive research contributions inspired us to exploit the beneficial role of IRSs in MEC systems.

C. Contributions and Organizations

Our main contributions are the employment of IRSs in MEC systems, and the joint design of computing and communications for minimizing the computational latency of IRS-aided MEC systems, detailed as follows.

- *New IRS-aided MEC system design and latency minimization problem formulation:* In order to further exploit the potential of MEC systems, we first propose IRS-aided MEC systems, for assisting the computational task off-loading of mobile devices. A latency-minimization problem is formulated for multi-device scenarios, which optimizes the computation off-loading volume, the edge computing resource allocation, the multi-user detection (MUD) matrix, and the IRS phase shift, subject to both the total edge computing capability and to the IRS phase shift constraints. Owing to the coupling effect of multiple optimization variables, the latency-minimization problem cannot be solved directly. Hence, relying on the block coordinate descent (BCD) technique, the original problem is decoupled into two subproblems for alternatively optimizing computing and communications settings.
- *Computing design:* Given a fixed communications setting, we decouple the computation off-loading volume and the edge computing resource allocation, again using the BCD technique. Our analysis reveals that given a fixed edge computing resource allocation, the optimal computation off-loading volume can be determined by assuming the equivalence of the latency induced by local computing and by edge computing. Given a fixed computation off-loading volume, the subproblem is proved to be a convex problem, and the optimal edge computing resource can be found by relying on the KKT conditions and on the classic bisection search method.
- *Communications design:* Given a computing setting, the objective function (OF) becomes available in a non-convex sum-of-ratios form, which cannot be solved using the algorithms developed in [22], [24], [26], [35]. To tackle this challenge, this problem is transformed to an equivalent parameterized form by introducing auxiliary variables. Our analysis reveals that this equivalent form can be decomposed into a series of tractable subproblems. Then, an iterative algorithm is developed to find the solution. In each iteration, the auxiliary variables are updated using the modified Newton's method, while upon reformulating this series of tractable subproblems by exploiting the equivalence between the weighted sum-rate maximization problem and the weighted mean square error (MSE) minimization problem, closed-form expressions are provided for the MUD matrix and for the IRS

phase shift, using the weighted minimum MSE method and MM algorithm, respectively. Our analysis reveals that the proposed algorithm exhibits a low complexity.

- *Study of the single-device scenario:* In order to complete the investigations, the single-device scenario is also studied, where neither the edge computing resource allocation nor the multi-user interference has to be considered. A low-complexity iterative algorithm is proposed by simplifying the algorithm developed in the aforementioned multi-device scenario.
- *Numerical validations and evaluations:* The numerical results verify the convergence of the proposed algorithms, and quantify the performance of our IRS-aided MEC system in terms of its latency in diverse simulation environments.

The rest of the paper is organized as follows. In Section II, we establish the system model and formulate the latency minimization problem. The solution of this latency minimization problem is provided in Section III. In Section IV, we investigate the solution of the special case, where a single device is served by the MEC system. Our numerical results are discussed in Section V. Finally, our conclusions are offered in Section VI.

Notation: In this paper, scalars are denoted by italic letters. Boldface lower- and upper-case letters denote vectors and matrices, respectively; $\mathbb{C}^{M\times N}$ represents the space of $M \times N$ complex matrices; $\boldsymbol{I}_N$ denotes an $N \times N$ identity matrix; j denotes the imaginary unit, i.e. $j^2 = -1$. The maths operations used throughout the paper are summarized in Table I.

Table I: Math operations

Notation	Operation
$\boldsymbol{x}^T$	the transpose of $\boldsymbol{x}$
$\boldsymbol{X}^T$	the transpose of $\boldsymbol{X}$
$\boldsymbol{x}^*$	the complex conjugate of $\boldsymbol{x}$
$\boldsymbol{X}^*$	the complex conjugate of $\boldsymbol{X}$
$\boldsymbol{x}^H$	the Hermitian transpose of $\boldsymbol{x}$
$\boldsymbol{X}^H$	the Hermitian transpose of $\boldsymbol{X}$
$\boldsymbol{X}^{-1}$	the inverse of $\boldsymbol{X}$
$\odot$	the Hadamard product
$\Re\{\cdot\}$	the real part of a complex number
$\arg\{\cdot\}$	the argument of a complex number
$\lvert\cdot\rvert$	the absolute value of a scalar
$\lVert\cdot\rVert$	the 2-norm of a vector
$\lfloor\cdot\rfloor$	the floor of a scalar
$\lceil\cdot\rceil$	the ceiling of a scalar
$\mathrm{diag}(\boldsymbol{x})$	the diagonal matrix where the diagonal elements are $\boldsymbol{x}$
$\mathrm{diag}(\boldsymbol{X})$	the vector whose elements are the diagonal elements of $\boldsymbol{X}$
$\mathcal{CN}(0,\sigma^2)$	Circularly Symmetric Complex Gaussian associated with zero-mean and variance σ^2

II. System Model and Problem Formulation

In this section, our system model is elaborated on, from both communications and computing perspectives. Following this, a latency-minimization problem is formulated for our IRS-aided MEC system, detailed as follows.

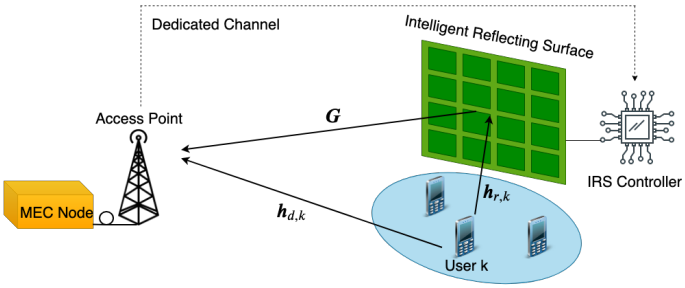

Figure 1: Illustration of the system model, where a N-element intelligent reflecting surface (IRS) assists the computation off-loading of K single-antenna devices to the edge computing node via the M-antenna access point.

A. Communications Model

As shown in Fig. 1, we consider an MEC system operating in a single-cell scenario, where K single-antenna devices may opt for off-loading a certain fraction of or all of their computational tasks to an edge computing node via an M-antenna AP through the wireless transmission link. The edge computing node and the AP are assumed to be co-located and connected using high-throughput low-latency optical fiber. Then, the latency imposed by the data communication between the AP and the edge computing node is deemed to be negligible. An IRS comprised of N reflecting elements is placed in the cell for assisting the devices' computation off-loading. We assume that both the antenna spacing at the AP and the element spacing of the IRS are high enough so that the small-scale fading associated both with two different antennas and with two different reflecting elements is independent, respectively.

The equivalent baseband channels spanning from the k-th device to the AP, and from the k-th device to the IRS, as well as from the IRS to the AP are denoted by $\boldsymbol{h}_{d,k} \in \mathbb{C}^{M\times 1}$, $\boldsymbol{h}_{r,k} \in \mathbb{C}^{N\times 1}$, and $\boldsymbol{G} \in \mathbb{C}^{M\times N}$, respectively. These channels are assumed to be perfectly estimated[1] and quasi-static, hence remaining near-constant when devices are scheduled for off-loading their computational tasks. As for the IRS, we simply set the amplitude reflection coefficient to 1 for all reflection elements and denote the phase shift coefficient vector by $\boldsymbol{\theta} = [\theta_1, \theta_2, \ldots, \theta_N]^T$, where $\theta_n \in [0, 2\pi)$ for all $n \in \{1, 2, \ldots, N\}$.[2] Then, we have the reflection-coefficient matrix of the IRS $\boldsymbol{\Theta} = \mathrm{diag}\{e^{j\theta_1}, e^{j\theta_2}, \ldots, e^{j\theta_N}\}$, where j represents the imaginary unit. It is assumed that the IRS phase shift setting is calculated at the AP in accordance with both the channel and computing dynamics, which is then sent to the IRS controller along the dedicated channel. The composite device-IRS-AP channel is modeled as a concatenation of the device-IRS link, the IRS reflection characterized by its phase shift, and the IRS-AP link.

Here, it is assumed that the computation off-loading of the K devices takes place over a given frequency band

[1]Naturally, this assumption is idealistic. Hence the algorithm developed in this paper can be deemed to represent the best-case bound for the latency performance of realistic scenarios.

[2]Due to the associated hardware limitations, only a limited number of discrete phase shifts can be provided for each IRS element in practice [23]. Our proposed algorithms provide the best-case bound for the latency of realistic scenarios. The phase-quantization effects are evaluated in Section V-A3.

B within the same time resource. Upon denoting the off-loading power, and off-loading signal of the K devices, as well as the noise vector by p_t, $\boldsymbol{s} = [s_1, s_2, \ldots, s_K]^T$, and $\boldsymbol{n} = [n_1, n_2, \ldots, n_M]^T$, respectively, the signal $\boldsymbol{y} \in \mathbb{C}^{M\times 1}$ received at the AP is readily formulated as

$$\boldsymbol{y} = \sqrt{p_t}\boldsymbol{H}\boldsymbol{s} + \boldsymbol{n} = \sqrt{p_t}\sum_{k=1}^{K}\left(\boldsymbol{h}_{d,k} + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,k}\right)s_k + \boldsymbol{n}, \quad (1)$$

where we assume $n_m \sim \mathcal{CN}(0, \sigma^2)$ for $m = 1, 2, \ldots, M$. Furthermore, we define $\boldsymbol{h}_k \triangleq \boldsymbol{h}_{d,k} + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,k}$ and $\boldsymbol{H} \triangleq \left[\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_K\right]$. As a computational complexity compromise at the AP, a linear MUD technique is invoked. Upon denoting the MUD matrix by $\boldsymbol{W} \in \mathbb{C}^{M\times K}$, the signal recovered at the AP is obtained as

$$\hat{\boldsymbol{s}} = \boldsymbol{W}^H\boldsymbol{y} = \boldsymbol{W}^H(\sqrt{p_t}\boldsymbol{H}\boldsymbol{s} + \boldsymbol{n}). \quad (2)$$

As for the k-th device, its recovered signal is formulated as

$$\hat{s}_k = \boldsymbol{w}_k^H\left[\sqrt{p_t}\sum_{j=1}^{K}\left(\boldsymbol{h}_{d,j} + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,j}\right)s_j + \boldsymbol{n}\right], \quad (3)$$

where $\boldsymbol{w}_k$ is the k-th column of the matrix $\boldsymbol{W}$. Then, the SINR of the k-th device's signal recovered is given by

$$\gamma_k(\boldsymbol{w}_k, \boldsymbol{\theta}) = \frac{p_t\left|\boldsymbol{w}_k^H\left(\boldsymbol{h}_{d,k} + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,k}\right)\right|^2}{p_t\sum_{j=1,j\neq k}^{K}\left|\boldsymbol{w}_k^H\left(\boldsymbol{h}_{d,j} + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,j}\right)\right|^2 + \sigma^2|\boldsymbol{w}_k^H|^2}. \quad (4)$$

Accordingly, upon assuming a perfect capacity-achieving transmission scheme is invoked, we arrive at the maximum achievable computation off-loading rate of the k-th device, formulated as

$$R_k(\boldsymbol{w}_k, \boldsymbol{\theta}) = B\log_2\left[1 + \gamma_k(\boldsymbol{w}_k, \boldsymbol{\theta})\right]. \quad (5)$$

B. Computing Model

We consider the data-partitioning based application of [7], where a fraction of the data can be processed locally, while the other part can be off-loaded to the edge node. The computing model is detailed for the local and edge computing as follows.

- *Local computing:* For the k-th device, L_k, ℓ_k, and c_k are used to represent its total number of bits to be processed, its computation off-loading volume in terms of the number of bits, and the number of CPU cycles required to process a single bit, respectively. As for the local computing, upon denoting the computational capability at the k-th device in terms of the number of CPU cycles per second by f_k^l, the time required for carrying out the local computation is formulated as $D_k^l(\ell_k) = (L_k - \ell_k)c_k/f_k^l$.
- *Edge computing:* we denote the maximum number of executable CPU cycles at the edge and the computational capability allocated to the k-th device by f_{total}^e and f_k^e, respectively, which obey $\sum_{k=1}^{K} f_k^e \leq f_{\text{total}}^e$. Here, it is assumed that the edge computing for the k-th device only begins its operation, when all its ℓ_k bits are completely off-loaded. In this case, the total latency of edge computing is jointly constituted by the computation off-loading, and by the edge computing, as well as by the end-to-end delay of sending the computational result back. Given that the computation result is typically of a small size [5], the feedback latency can be negligible, upon using the technique of ultra-reliable low-latency communications [36]. Then, the total latency imposed by the computation off-loading and the edge computing is given by $D_k^e(\boldsymbol{w}_k, \boldsymbol{\theta}, \ell_k, f_k^e) = \ell_k/R_k(\boldsymbol{w}_k, \boldsymbol{\theta}) + \ell_k c_k/f_k^e$.

To this end, the latency of the k-th device can be readily calculated by selecting the maximum value between those imposed by the local and by the edge computing, formulated as

$$\begin{aligned} D_k(\boldsymbol{w}_k, \boldsymbol{\theta}, \ell_k, f_k^e) &= \max\left\{D_k^l(\ell_k), D_k^e(\boldsymbol{w}_k, \boldsymbol{\theta}, \ell_k, f_k^e)\right\} \quad (6)\\ &= \max\left\{\frac{(L_k - \ell_k)c_k}{f_k^l}, \frac{\ell_k}{R_k(\boldsymbol{w}_k, \boldsymbol{\theta})} + \frac{\ell_k c_k}{f_k^e}\right\}. \end{aligned}$$

C. Problem Formulation

In this paper, we aim for minimizing the weighted computational latency of all the devices, by jointly optimizing the computation off-loading volume $\boldsymbol{\ell} = [\ell_1, \ell_2, \ldots, \ell_K]^T$, the edge computing resources $\boldsymbol{f}^e = [f_1^e, f_2^e, \ldots, f_K^e]^T$ allocated to each device, the MUD matrix $\boldsymbol{W}$, and the IRS phase shift $\boldsymbol{\theta}$. Specifically, the weighted delay minimization problem is formulated as

$$\begin{aligned} \mathcal{P}0: \min_{\boldsymbol{W},\boldsymbol{\theta},\boldsymbol{\ell},\boldsymbol{f}^e} & \sum_{k=1}^{K}\varpi_k D_k(\boldsymbol{w}_k, \boldsymbol{\theta}, \ell_k, f_k^e)\\ \text{s.t.} \quad & 0 \leq \theta_n < 2\pi, \quad n = 1, 2, \ldots, N, && (7a)\\ & \ell_k \in \{0, 1, \ldots, L_k\}, \quad k = 1, 2, \ldots, K, && (7b)\\ & \sum_{k=1}^{K} f_k^e \leq f_{\text{total}}^e, && (7c)\\ & f_k^e \geq 0, \quad k = 1, 2, \ldots, K. && (7d) \end{aligned}$$

where ϖ_k represents the weight of the k-th device. (7a) specifies the range of the phase shift of the IRS elements; (7b) indicates that the computation off-loading volume should be an integer between 0 and L_k for the k-th device; Finally, (7c) and (7d) restrict the range of the edge computing resources allocated to each device.

Remark 1. *In Problem $\mathcal{P}0$, we have a total of four optimization variables, namely, the off-loading volume, edge computing resource allocation, MUD matrix, and IRS phase shift. The optimization of the former two variables is related to the computing setting, while the optimization of the other two specifies the communications design. The difficulties of solving Problem $\mathcal{P}0$ are owing to three aspects. The first one is the segmented form of the OF. The second one is the coupling effect between the MUD matrix $\boldsymbol{W}$ and the IRS phase shift vector $\boldsymbol{\theta}$. The final one is that the OF is non-convex regarding the phase shift $\boldsymbol{\theta}$. Hence, it is an open challenge to obtain a globally optimal solution directly. In this paper, a locally optimal solution is provided. Specifically, upon using the popular BCD technique for decoupling the communications and computing designs, the segmented form of the OF can be*

easily transformed to a more tractable form. Similarly, optimal solutions can be provided for the MUD matrix $\boldsymbol{W}$ and for the IRS phase shift vector $\boldsymbol{\theta}$, after they are decoupled using the BCD technique in the communications design. To tackle the non-convexity regarding $\boldsymbol{\theta}$, the Majorization-Minimization (MM) algorithm is invoked, which is capable of iteratively approaching a locally optimal solution at low complexity.

III. Joint Optimization of Computing and Communications Setting

The joint optimization of computing and computations settings is realized relying on the BCD technique. The pivotal idea of the BCD technique is to optimize one of the variables while fixing the other variables in an alternating manner, until the convergence of the OF is achieved. In the rest of this section, the joint optimization of the off-loading volume and of the edge computing resource allocation is presented while fixing the communications setting, followed by the joint optimization of the MUD matrix and of the IRS phase shift while fixing the computing setting. Our goal is the joint optimization both of the communications and of the computing design.

A. Joint Optimization of the Off-loading Volume and the Edge Computing Resource Allocation While Fixing the Communications Settings

Given an MUD matrix $\boldsymbol{W}$ and an IRS phase shift vector $\boldsymbol{\theta}$, Problem $\mathcal{P}0$ can be simplified to

$$\mathcal{P}1: \min_{\boldsymbol{\ell}, \boldsymbol{f}^e} \sum_{k=1}^{K} \varpi_k D_k(\ell_k, f_k^e) \quad \text{s.t.} \quad (7b), (7c), (7d). \tag{8a}$$

The optimization of $\boldsymbol{\ell}$ and $\boldsymbol{f}^e$ can be decoupled, relying on the aforementioned BCD technique, detailed as follows.

1) Optimization of $\boldsymbol{\ell}$: The value of $\boldsymbol{\ell}$ can be optimized, with the aid of the proposition below.

Proposition 1. *Given an MUD matrix $\boldsymbol{W}$, and an IRS phase shift coefficient vector $\boldsymbol{\theta}$. as well as an edge computing resource allocation vector $\boldsymbol{f}^e$, the optimal number of off-loaded bits is given by*

$$\ell_k^* = \underset{\hat{\ell}_k \in \left\{ \lfloor \hat{\ell}_k^* \rfloor, \lceil \hat{\ell}_k^* \rceil \right\}}{\arg\min} D_k(\hat{\ell}_k), \tag{9}$$

where $\lfloor \cdot \rfloor$ and $\lceil \cdot \rceil$ represent the floor and ceiling operations, respectively, and $\hat{\ell}_k^*$ is selected for ensuring that the value of $D_k^l(\hat{\ell}_k)$ becomes equivalent to that of $D_k^e(\hat{\ell}_k)$, i.e.

$$\hat{\ell}_k^* = \frac{L_k c_k R_k f_k^e}{f_k^e f_k^l + c_k R_k \left(f_k^e + f_k^l\right)}. \tag{10}$$

Proof: See Appendix A. ∎

2) Optimization of $\boldsymbol{f}^e$: Here, the edge computing resource allocation $\boldsymbol{f}^e$ is optimized, while fixing the MUD matrix $\boldsymbol{W}$, the IRS phase shift coefficient vector $\boldsymbol{\theta}$, and the off-loading volume $\boldsymbol{\ell}$. Specifically, upon substituting (10) into the OF of Problem $\mathcal{P}1$, the problem can be reformulated as:

$$\mathcal{P}1\text{-}E: \min_{\boldsymbol{f}^e} \sum_{k=1}^{K} \frac{\varpi_k (L_k c_k^2 R_k + L_k c_k f_k^e)}{f_k^e f_k^l + c_k R_k (f_k^e + f_k^l)} \quad \text{s.t.} \quad (7c), (7d). \tag{11a}$$

Problem $\mathcal{P}1$-E can be proved to be a convex optimization problem following the proposition below.

Proposition 2. *Problem $\mathcal{P}1$-E is a convex optimization problem.*

Proof: See Appendix B. ∎

Since Problem $\mathcal{P}1$-E is convex and the Slater's condition [37] is satisfied [3], the Karush–Kuhn–Tucker (KKT) may be imposed on the problem for finding its optimal solution. Specifically, the Lagrangian function associated with Problem $\mathcal{P}1$-E is given by

$$\mathcal{L}(\boldsymbol{f}^e, \mu, \boldsymbol{\nu}) = \sum_{k=1}^{K} \frac{\varpi_k (L_k c_k^2 R_k + L_k c_k f_k^e)}{c_k R_k f_k^l + (f_k^l + c_k R_k) f_k^e} + \mu \left(\sum_{k=1}^{K} f_k^e - f_{\text{total}}^e \right), \tag{12}$$

where the variable μ is the non-negative Lagrange multiplier, while the optimal edge computing resource allocation vector $\boldsymbol{f}^{e*}$ and the optimal Lagrange multiplier μ^* should satisfy the following KKT conditions, for $k = 1, 2, \ldots, K$:

$$\frac{\partial \mathcal{L}}{\partial f_k^e} = \frac{-\varpi_k L_k c_k^3 R_k^2}{\left[c_k R_k f_k^l + (f_k^l + c_k R_k) f_k^{e*}\right]^2} + \mu^* = 0, \tag{13}$$

$$\mu^* \left(\sum_{k=1}^{K} f_k^{e*} - f_{\text{total}}^e \right) = 0, \tag{14}$$

$$f_k^{e*} \geq 0. \tag{15}$$

The value of f_k^e can be directly derived from (13) for a given

Algorithm 1 Joint optimization of $\boldsymbol{\ell}$ and $\boldsymbol{f}^e$, given $\boldsymbol{W}$ and $\boldsymbol{\theta}$

Input: $\boldsymbol{h}_{r,k}$, $\boldsymbol{h}_{d,k}$, $\boldsymbol{G}$, B, p_t, σ^2, K, ϖ_k, L_k, c_k, f_k^l, f_{total}^e, $t_1^{\max}$, ϵ, $\boldsymbol{W}$, $\boldsymbol{\theta}$, and $\hat{\boldsymbol{f}}^e$ satisfying (7c) and (7d)
Output: Optimal $\boldsymbol{\ell}^*$ and $\boldsymbol{f}^{e*}$, given $\boldsymbol{W}$ and $\boldsymbol{\theta}$
1. Initialization
initialize $t_1 = 0$, $\epsilon_1^{(0)} = 1$, $\boldsymbol{f}^{e(0)} \leftarrow \hat{\boldsymbol{f}}^e$
calculate $\boldsymbol{R}$ using (5)
2. Joint optimization of $\boldsymbol{\ell}$ and $\boldsymbol{f}^e$
while $\epsilon_1^{(t_1)} > \epsilon$ && $t_1 < t_1^{\max}$ **do**
• calculate $\boldsymbol{\ell}^{(t_1+1)}$ using (10)
• calculate $\boldsymbol{f}^{e(t_1+1)}$ and μ by using (16) and the bisection search method, respectively
• $\epsilon_1^{(t_1+1)} = \frac{\left| \text{obj}\left(\boldsymbol{\ell}^{(t_1+1)}, \boldsymbol{f}^{e(t_1+1)}\right) - \text{obj}\left(\boldsymbol{\ell}^{(t_1)}, \boldsymbol{f}^{e(t_1)}\right) \right|}{\text{obj}\left(\boldsymbol{\ell}^{(t_1+1)}, \boldsymbol{f}^{e(t_1+1)}\right)}$
• $t_1 \leftarrow t_1 + 1$
end while
3. Output optimal $\boldsymbol{\ell}^*$ and $\boldsymbol{f}^{e*}$
$\boldsymbol{\ell}^* \leftarrow \boldsymbol{\ell}^{(t_1)}$ and $\boldsymbol{f}^{e*} \leftarrow \boldsymbol{f}^{e(t_1)}$

[3] In words, the Slater's condition for convex programming states that strong duality holds if all constraints are satisfied and the nonlinear constraints are satisfied with strict inequalities.

μ, which is written as

$$f_k^e = \frac{\sqrt{\frac{\varpi_k L_k c_k^3 R_k^2}{\mu}} - c_k R_k f_k^l}{f_k^l + c_k R_k}, \quad k = 1, \ldots, K. \quad (16)$$

In order to ensure $f_k^e \geq 0$ in (16), we have $\sqrt{\frac{\varpi_k L_k c_k^3 R_k^2}{\mu}} - c_k R_k f_k^l \geq 0$, which is reformulated as $\mu \leq \frac{\varpi_k L_k c_k}{{f_k^l}^2}$. Given that $\mu \neq 0$ in (16), the optimal μ^* can be found in the range of $(\mu_l, \mu_u] = \left(0, \min_k \left(\frac{\varpi_k L_k c_k}{{f_k^l}^2}\right)\right]$ to ensure (14), using the well-known bisection search method associated with the termination coefficient of ϵ, because $\sum_{k=1}^K f_k^e$ can be proved to be monotonically decreasing with respect to μ. The procedure of solving Problem $\mathcal{P}1$ is summarized in Algorithm 1. The complexity of Algorithm 1 is dominated by calculating $\boldsymbol{f}^{e(t_1+1)}$ using (16) and by calculating μ using the bisection search method. Its complexity is on the order of $\mathcal{O}\big(\log_2(\frac{\mu_u - \mu_l}{\epsilon})K\big)$. Thus the total complexity of Algorithm 1 is $\mathcal{O}\big(t_1^{\max} \log_2(\frac{\mu_u - \mu_l}{\epsilon})K\big)$.

B. Joint Optimization of the MUD Matrix and the IRS Phase Shift Coefficient While Fixing the Computing Settings

Given an off-loading volume vector $\boldsymbol{\ell}$ and an edge computing resource allocation vector $\boldsymbol{f}^e$, Problem $\mathcal{P}0$ is reformulated as

$$\mathcal{P}2: \min_{\boldsymbol{W}, \boldsymbol{\theta}} \sum_{k=1}^K \varpi_k D_k(\boldsymbol{w}_k, \boldsymbol{\theta})$$
$$\text{s.t.} \quad 0 \leq \theta_n < 2\pi, \quad n = 1, 2, \ldots, N. \quad (17a)$$

Remark 2. *The challenges of solving Problem $\mathcal{P}2$ are due to two aspects. The first one is the segmented form of $D_k(\boldsymbol{w}_k, \boldsymbol{\theta})$ that is caused by the operation* max *as detailed in* (6)*, while the second issue is that the OF is the summation of fractional functions, with respect to $\boldsymbol{W}$ and $\boldsymbol{\theta}$ as shown in the OF of Problem $\mathcal{P}2$-$E1$ below, which makes the problem a non-convex sum-of-ratios optimization. In order to tackle these two issues, we transform the problem as follows.*

1) Problem Transformation: As detailed in Proposition 1, the optimal solution of Problem $\mathcal{P}0$ results in $D_k = D_k^l = D_k^e$. Hence upon replacing D_k by D_k^e and removing the constant terms, Problem $\mathcal{P}2$ is reformulated as:

$$\mathcal{P}2\text{-}E1: \min_{\boldsymbol{W}, \boldsymbol{\theta}} \sum_{k=1}^K \frac{\varpi_k \ell_k}{R_k(\boldsymbol{w}_k, \boldsymbol{\theta})}$$
$$\text{s.t.} \quad 0 \leq \theta_n < 2\pi, \quad n = 1, 2, \ldots, N. \quad (18a)$$

It is then rewritten as the following equivalent form:

$$\mathcal{P}2\text{-}E2: \min_{\boldsymbol{W}, \boldsymbol{\theta}, \boldsymbol{\beta}} \sum_{k=1}^K \beta_k$$
$$\text{s.t.} \quad \frac{\varpi_k \ell_k}{R_k(\boldsymbol{w}_k, \boldsymbol{\theta})} \leq \beta_k, \quad k = 1, 2, \ldots, K, \quad (19a)$$
$$0 \leq \theta_n < 2\pi, \quad n = 1, 2, \ldots, N. \quad (19b)$$

The following proposition may assist us in solving Problem $\mathcal{P}2$-$E2$.

Algorithm 2 Joint optimization of $\boldsymbol{W}$ and $\boldsymbol{\theta}$, given $\boldsymbol{\ell}$ and $\boldsymbol{f}^e$

Input: $\boldsymbol{h}_{r,k}$, $\boldsymbol{h}_{d,k}$, $\boldsymbol{G}$, B, p_t, σ^2, ϖ_k, $\boldsymbol{\ell}$, and $\boldsymbol{f}^e$
Output: Optimal $\boldsymbol{W}^*$ and $\boldsymbol{\theta}^*$, given $\boldsymbol{\ell}$ and $\boldsymbol{f}^e$
1. Initialization
initialize $t_2 = 0$, $\zeta \in (0,1)$, $\epsilon \in (0,1)$, and $\boldsymbol{\theta}^{(0)}$ satisfying (7a)
calculate $\boldsymbol{W}^{(0)}$ and $\boldsymbol{R}^{(0)}$ using (31) and (34), respectively
calculate $\boldsymbol{\lambda}^{(0)}$ and $\boldsymbol{\beta}^{(0)}$ using (26)
2. Joint optimization of $\boldsymbol{W}$, $\boldsymbol{\theta}$, $\boldsymbol{\lambda}$ and $\boldsymbol{\beta}$
repeat
- update $\boldsymbol{W}^{(t_2+1)}$ and $\boldsymbol{\theta}^{(t_2+1)}$ using Algorithm 3
- update $\boldsymbol{\lambda}^{(t_2+1)}$ and $\boldsymbol{\beta}^{(t_2+1)}$ as follows

$$\lambda_k^{(t_2+1)} = \lambda_k^{(t_2)} - \frac{\zeta^{i^{(t_2+1)}} \chi_k\big(\lambda_k^{(t_2)}\big)}{R_k\big(\boldsymbol{w}_k^{(t_2+1)}, \boldsymbol{\theta}^{(t_2+1)}\big)}, \quad (20)$$

and

$$\beta^{(t_2+1)} = \beta^{(t_2)} - \frac{\zeta^{i^{(t_2+1)}} \kappa_k\big(\beta_k^{(t_2)}\big)}{R_k\big(\boldsymbol{w}_k^{(t_2+1)}, \boldsymbol{\theta}^{(t_2+1)}\big)}, \quad (21)$$

where $i^{(t_2+1)}$ is the smallest integer among $i \in \{1, 2, 3, \ldots\}$ satisfying

$$\sum_{k=1}^K \left| \chi_k\left(\lambda_k^{(t_2)} - \frac{\zeta^i \chi_k(\lambda_k^{(t_2)})}{R_k(\boldsymbol{w}_k^{(t_2+1)}, \boldsymbol{\theta}^{(t_2+1)})}\right)\right|^2 + \sum_{k=1}^K \left| \kappa_k\left(\beta^{(t_2)} - \frac{\zeta^i \kappa_k(\beta^{(t_2)})}{R_k(\boldsymbol{w}_k^{(t_2+1)}, \boldsymbol{\theta}^{(t_2+1)})}\right)\right|^2 \leq (1 - \epsilon_3 \zeta^i)^2 \sum_{k=1}^K \left[\left|\chi_k\big(\lambda_k^{(t_2)}\big)\right|^2 + \left|\kappa_k\big(\beta^{(t_2)}\big)\right|^2 \right]. \quad (22)$$

- $t_2 \leftarrow t_2 + 1$

until the following conditions are achieved

$$\lambda_k^{(t_2)} R_k(\boldsymbol{w}_k^{(t_2)}, \boldsymbol{\theta}^{(t_2)}) - 1 = 0, \quad (23)$$

$$\beta_k R_k(\boldsymbol{w}_k^{(t_2)}, \boldsymbol{\theta}^{(t_2)}) - \varpi_k \ell_k = 0 \quad (24)$$

3. Output optimal $\boldsymbol{W}^*$ and $\boldsymbol{\theta}^*$
$\boldsymbol{W}^* \leftarrow \boldsymbol{W}^{(t_2)}$ and $\boldsymbol{\theta}^* \leftarrow \boldsymbol{\theta}^{(t_2)}$

Proposition 3. *If $(\boldsymbol{W}^*, \boldsymbol{\theta}^*, \boldsymbol{\beta}^*)$ is the solution of Problem $\mathcal{P}2$-$E2$, a $\boldsymbol{\lambda}^* = [\lambda_1, \lambda_2, \ldots, \lambda_K]$ exists that $(\boldsymbol{W}^*, \boldsymbol{\theta}^*)$ satisfies the KKT conditions of the following problem, when we set $\boldsymbol{\beta} = \boldsymbol{\beta}^*$ and $\boldsymbol{\lambda} = \boldsymbol{\lambda}^*$*

$$\mathcal{P}2\text{-}E3: \min_{\boldsymbol{W}, \boldsymbol{\theta}} \sum_{k=1}^K \lambda_k \big[\varpi_k \ell_k - \beta_k R_k(\boldsymbol{w}_k, \boldsymbol{\theta})\big]$$
$$\textit{s.t.} \quad 0 \leq \theta_n < 2\pi, \quad n = 1, 2, \ldots, N. \quad (25a)$$

Furthermore, $(\boldsymbol{W}^*, \boldsymbol{\theta}^*)$ also satisfies the following equations, when we set $\boldsymbol{\beta} = \boldsymbol{\beta}^*$ and $\boldsymbol{\lambda} = \boldsymbol{\lambda}^*$

$$\begin{cases} \lambda_k = \frac{1}{R_k(\boldsymbol{w}_k^*, \boldsymbol{\theta}^*)}, & k = 1, 2, \ldots, K, \\ \beta_k = \frac{\varpi_k \ell_k}{R_k(\boldsymbol{w}_k^*, \boldsymbol{\theta}^*)}, & k = 1, 2, \ldots, K. \end{cases} \quad (26)$$

Correspondingly, if $(\boldsymbol{W}^*, \boldsymbol{\theta}^*)$ is a solution to Problem $\mathcal{P}2$-$E3$ and satisfies (26) *when we set $\boldsymbol{\beta} = \boldsymbol{\beta}^*$ and $\boldsymbol{\lambda} = \boldsymbol{\lambda}^*$, $(\boldsymbol{W}^*, \boldsymbol{\theta}^*, \boldsymbol{\beta}^*)$ is the solution of Problem $\mathcal{P}2$-$E2$ associated with the Lagrange multiplier $\boldsymbol{\lambda} = \boldsymbol{\lambda}^*$.*

Proof: See Appendix C. ∎

To this end, the sum-of-ratios form in Problem $\mathcal{P}2$-$E1$

has been transformed to a parameterized subtractable form in Problem $\mathcal{P}2$-$E3$, which can be solved in two steps [38]–[40]: the first step is to obtain $\boldsymbol{W}^*$ and $\boldsymbol{\theta}^*$ by solving Problem $\mathcal{P}2$-$E3$, given $\boldsymbol{\beta}$ and $\boldsymbol{\lambda}$; the second step is to update $\boldsymbol{\beta}$ and $\boldsymbol{\lambda}$ using the modified Newton's method until the convergence is achieved. The procedure is summarized in Algorithm 2, where we have

$$\chi_k(\lambda_k) = \lambda_k R_k(\boldsymbol{w}_k^*, \boldsymbol{\theta}^*) - 1, \quad k = 1, 2, \ldots, K, \quad (27)$$

$$\kappa_k(\beta_k) = \beta_k R_k(\boldsymbol{w}_k^*, \boldsymbol{\theta}^*) - \varpi_k \ell_k, \quad k = 1, 2, \ldots, K. \quad (28)$$

The complexity of Algorithm 2 is analyzed at the end of Section III-B.

Let us now focus our attention on the first step of solving Problem $\mathcal{P}2$-$E3$, i.e. optimizing $\boldsymbol{W}^*$ and $\boldsymbol{\theta}^*$, given a set of $\boldsymbol{\beta}$ and $\boldsymbol{\lambda}$ as well as an off-loading volume vector $\boldsymbol{\ell}$. In this case, Problem $\mathcal{P}2$-$E3$ can be simplified to $\max_{\boldsymbol{W},\boldsymbol{\theta}} \sum_{k=1}^{K} \lambda_k \beta_k R_k(\boldsymbol{w}_k, \boldsymbol{\theta})$ subject to (25a), which constitutes a weighted sum-rate maximization problem. As revealed in [41], maximizing the weighted sum-rate can be accomplished via weighted MSE minimization. The latter problem is easier to handle, because it is convex regarding each optimization variable, while fixing others. As such, we focus our attention on constructing the corresponding weighted MSE minimization problem. Specifically, following Theorem I in [41], we introduce an auxiliary weight variable Υ_k for the k-th device and formulate the corresponding weighted MSE minimization problem as:

$$\begin{aligned}\mathcal{P}2\text{-}E4: \min_{\boldsymbol{W},\boldsymbol{\theta}} \sum_{k=1}^{K} \big[&\Upsilon_k e_k(\boldsymbol{W}, \boldsymbol{\theta}) \\ &- \lambda_k \beta_k \log_2(\lambda_k^{-1} \beta_k^{-1} \Upsilon_k) - \lambda_k \beta_k\big] \\ \text{s.t.} \quad &0 \le \theta_n < 2\pi, \quad \forall n \in \{1, 2, \ldots, N\}, \end{aligned} \quad (29a)$$

where the mathematical expression of $\{\Upsilon_k\}$ is given in Section III-B3 and e_k represents the MSE of the k-th user, which is given by

$$\begin{aligned} e_k(\boldsymbol{W}, \boldsymbol{\theta}) &\triangleq \mathbb{E}\big[(\hat{s}_k - s_k)(\hat{s}_k - s_k)^H\big] \\ &= \big[\sqrt{p_t}\boldsymbol{w}_k^H(\boldsymbol{h}_{d,k} + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,k}) - 1\big] \times \\ &\quad \big[\sqrt{p_t}\boldsymbol{w}_k^H(\boldsymbol{h}_{d,k} + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,k}) - 1\big]^H \\ &\quad + p_t \sum_{j \ne k}^{K} \boldsymbol{w}_k^H(\boldsymbol{h}_{d,j} + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,j})(\boldsymbol{h}_{d,j} + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,j})^H \boldsymbol{w}_k \\ &\quad + \sigma^2 \boldsymbol{w}_k^H \boldsymbol{w}_k. \end{aligned} \quad (30)$$

As such, compared to Problem $\mathcal{P}2$-$E3$, Problem $\mathcal{P}2$-$E4$ becomes more tractable, because given an IRS phase shift coefficient vector, the OF of Problem $\mathcal{P}2$-$E4$ is convex regarding an optimization variable, while fixing the other one. Again, the BCD technique is invoked for solving this problem as follows.

2) MUD Matrix Design: In Problem $\mathcal{P}2$-$E4$, fixing the phase shift coefficient vector $\boldsymbol{\theta}$ and the auxiliary variable Υ_k, the MUD vector can be obtained by forcing the first-order derivative of the OF with respect to $\boldsymbol{w}_k$ as 0. After several steps of mathematical manipulations, it is readily observed that the above minimization is equivalent to minimizing the weighted MSE. Then, the MUD vector is given by [42, Sec. 6.2.3]

$$\boldsymbol{w}_k = \sqrt{p_t}\boldsymbol{J}^{-1}(\boldsymbol{h}_{d,k} + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,k}), \quad (31)$$

where $\boldsymbol{J} = p_t \sum_{j=1}^{K}(\boldsymbol{h}_{d,j} + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,j})(\boldsymbol{h}_{d,j} + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,j})^H + \sigma^2 \boldsymbol{I}_M$.

3) Auxiliary Variable Design: Fixing $\boldsymbol{\theta}$ and $\boldsymbol{w}_k$, the optimal auxiliary variable can be obtained by minimizing the OF of Problem $\mathcal{P}2$-$E4$ with respect to Υ_k, given by

$$\Upsilon_k = \lambda_k \beta_k (e_k)^{-1}. \quad (32)$$

Furthermore, substituting (31) into (30), the MSE becomes

$$e_k^{\text{MMSE}} = 1 - p_t(\boldsymbol{h}_{d,k} + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,k})^H \boldsymbol{J}^{-1}(\boldsymbol{h}_{d,k} + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,k}). \quad (33)$$

Bearing in mind that the relationship between the SINR and the MSE of the system equipped with the minimum mean square error (MMSE) MUD is given by $\gamma_k = (e_k^{\text{MMSE}})^{-1} - 1$ [42, Sec. 6.2.3], (5) may be reformulated as

$$R_k = -B \log_2\big(e_k^{\text{MMSE}}\big). \quad (34)$$

4) IRS Phase Shift Coefficient Design: In this subsection, we focus our attention on optimizing the reflection phase shift coefficients $\boldsymbol{\theta}$, while fixing the auxiliary variable Υ_k and the MUD matrix $\boldsymbol{W}$. Specifically, by substituting (30) into the OF of Problem $\mathcal{P}2$-$E4$ and removing the terms that are independent of the phase shift coefficient vector $\boldsymbol{\theta}$, Problem $\mathcal{P}2$-$E4$ is reformulated as:

$$\begin{aligned}\mathcal{P}2\text{-}E5: \min_{\boldsymbol{\theta}} &\sum_{k=1}^{K}\sum_{j=1}^{K} \Upsilon_k p_t \boldsymbol{w}_k^H \boldsymbol{h}_j \boldsymbol{h}_j^H \boldsymbol{w}_k \\ &- \sum_{k=1}^{K} \Upsilon_k \sqrt{p_t} \boldsymbol{h}_k^H \boldsymbol{w}_k - \sum_{k=1}^{K} \Upsilon_k \sqrt{p_t} \boldsymbol{w}_k^H \boldsymbol{h}_k \\ \text{s.t.} \quad &0 \le \theta_n \le 2\pi, \quad \forall n \in \{1, 2, \ldots, N\}, \end{aligned} \quad (35)$$

where the first and the second terms in the OF can be respectively formulated in expansion forms as

$$\begin{aligned} &\sum_{k=1}^{K}\sum_{j=1}^{K} \Upsilon_k p_t \boldsymbol{w}_k^H \boldsymbol{h}_j \boldsymbol{h}_j^H \boldsymbol{w}_k \\ &= \sum_{k=1}^{K}\sum_{j=1}^{K} \big(\Upsilon_k p_t \boldsymbol{w}_k^H \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,j} \boldsymbol{h}_{r,j}^H \boldsymbol{\Theta}^H \boldsymbol{G}^H \boldsymbol{w}_k \\ &\quad + \Upsilon_k p_t \boldsymbol{w}_k^H \boldsymbol{h}_{d,j} \boldsymbol{h}_{r,j}^H \boldsymbol{\Theta}^H \boldsymbol{G}^H \boldsymbol{w}_k \\ &\quad + \Upsilon_k p_t \boldsymbol{w}_k^H \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_{r,j} \boldsymbol{h}_{d,j}^H \boldsymbol{w}_k + \Upsilon_k p_t \boldsymbol{w}_k^H \boldsymbol{h}_{d,j} \boldsymbol{h}_{d,j}^H \boldsymbol{w}_k\big), \end{aligned} \quad (36)$$

and

$$\begin{aligned} &\sum_{k=1}^{K} \Upsilon_k \sqrt{p_t} \boldsymbol{h}_k^H \boldsymbol{w}_k \\ &= \sum_{k=1}^{K} \big(\Upsilon_k \sqrt{p_t} \boldsymbol{h}_{d,k}^H \boldsymbol{w}_k + \Upsilon_k \sqrt{p_t} \boldsymbol{h}_{r,k}^H \boldsymbol{\Theta}^H \boldsymbol{G}^H \boldsymbol{w}_k\big). \end{aligned} \quad (37)$$

Upon defining $\boldsymbol{A} \triangleq \sum_{k=1}^{K} \Upsilon_k p_t \boldsymbol{G}^H \boldsymbol{w}_k \boldsymbol{w}_k^H \boldsymbol{G}$, $\boldsymbol{B} \triangleq \sum_{j=1}^{K} \boldsymbol{h}_{r,j} \boldsymbol{h}_{r,j}^H$, $\boldsymbol{C} \triangleq \sum_{k=1}^{K}\sum_{j=1}^{K} \Upsilon_k p_t \boldsymbol{h}_{r,j} \boldsymbol{h}_{d,j}^H \boldsymbol{w}_k \boldsymbol{w}_k^H \boldsymbol{G}$, and $\boldsymbol{D} \triangleq \sum_{k=1}^{K} \Upsilon_k \sqrt{p_t} \boldsymbol{h}_{r,k} \boldsymbol{w}_k^H \boldsymbol{G}$, Problem $\mathcal{P}2$-$E5$ may be rewrit-

Algorithm 3 Joint optimization of $\boldsymbol{W}$ and $\boldsymbol{\theta}$, given $\boldsymbol{\lambda}$ and $\boldsymbol{\beta}$

Input: $\boldsymbol{h}_{r,k}$, $\boldsymbol{h}_{d,k}$, $\boldsymbol{G}$, B, p_t, σ^2, ϖ_k, $t_2^{\max}$, ϵ, $\boldsymbol{\lambda}$ and $\boldsymbol{\beta}$
Output: Optimal $\boldsymbol{W}^*$ and $\boldsymbol{\theta}^*$, given $\boldsymbol{\lambda}$ and $\boldsymbol{\beta}$
1. Initialization
initialize $t_3 = 0$, $\epsilon_3^{(0)} = 1$, $\boldsymbol{\theta}^{(0)}$ satisfying (7a)
2. Joint optimization of $\boldsymbol{W}$ and $\boldsymbol{\theta}$
while $\epsilon_3^{(t_2)} > \epsilon$ && $t_3 < t_3^{\max}$ **do**
- calculate $\boldsymbol{W}^{(t_3+1)}$ using (31)
- calculate $\boldsymbol{\Upsilon}^{(t_3+1)}$ using (32)
- calculate $\boldsymbol{\theta}^{(t_3+1)}$ by solving Problem $\mathcal{P}2$-$E6$ with the aid of the MM algorithm
- $\epsilon_3^{(t_3+1)} = \frac{\left|\text{obj}\left(\boldsymbol{W}^{(t_3+1)},\boldsymbol{\theta}^{(t_3+1)}\right) - \text{obj}\left(\boldsymbol{W}^{(t_3)},\boldsymbol{\theta}^{(t_3)}\right)\right|}{\text{obj}\left(\boldsymbol{W}^{(t_3+1)},\boldsymbol{\theta}^{(t_3+1)}\right)}$
- $t_3 \leftarrow t_3 + 1$

end while
3. Output optimal $\boldsymbol{W}^*$ and $\boldsymbol{\theta}^*$, given $\boldsymbol{\lambda}$ and $\boldsymbol{\beta}$
$\boldsymbol{W}^* \leftarrow \boldsymbol{W}^{(t_3)}$ and $\boldsymbol{\theta}^* \leftarrow \boldsymbol{\theta}^{(t_3)}$

ten as:

$$\begin{aligned}\mathcal{P}2\text{-}E6: \min_{\boldsymbol{\theta}} \ & \text{tr}(\boldsymbol{\Theta}^H \boldsymbol{A}\boldsymbol{\Theta}\boldsymbol{B}) + \text{tr}\big[\boldsymbol{\Theta}^H(\boldsymbol{C}-\boldsymbol{D})^H\big] \\ & + \text{tr}\big[\boldsymbol{\Theta}(\boldsymbol{C}-\boldsymbol{D})\big] \\ \text{s.t.} \quad & 0 \le \theta_n \le 2\pi, \quad \forall n \in \{1,2,\ldots,N\}. \end{aligned} \tag{38}$$

Defining $\boldsymbol{\phi} \triangleq [\phi_1,\ldots,\phi_N]^T$ where $\phi_n = e^{j\theta_n}$, and $\boldsymbol{v} = \big[[\boldsymbol{C}-\boldsymbol{D}]_{1,1},\ldots,[\boldsymbol{C}-\boldsymbol{D}]_{N,N}\big]^T$, we have

$$\text{tr}(\boldsymbol{\Theta}^H \boldsymbol{A}\boldsymbol{\Theta}\boldsymbol{B}) = \boldsymbol{\phi}^H(\boldsymbol{A}\odot\boldsymbol{B})\boldsymbol{\phi}, \tag{39}$$

where $\odot$ represents the Hadamard product, and

$$\text{tr}\big[\boldsymbol{\Theta}^H(\boldsymbol{C}-\boldsymbol{D})^H\big] = \boldsymbol{v}^H\boldsymbol{\phi}^*, \ \text{tr}\big[\boldsymbol{\Theta}(\boldsymbol{C}-\boldsymbol{D})\big] = \boldsymbol{\phi}^T\boldsymbol{v}. \tag{40}$$

Further defining $\boldsymbol{\Psi} \triangleq \boldsymbol{A}\odot\boldsymbol{B}$, we may equivalently rewrite Problem $\mathcal{P}2$-$E6$ as:

$$\begin{aligned}\mathcal{P}2\text{-}E7: \min_{\boldsymbol{\phi}} \ & f(\boldsymbol{\phi}) = \boldsymbol{\phi}^H\boldsymbol{\Psi}\boldsymbol{\phi} + 2\Re\big\{\boldsymbol{\phi}^H\boldsymbol{v}^*\big\} \\ \text{s.t.} \quad & |\phi_n| = 1, \quad \forall n \in \{1,2,\ldots,N\}. \end{aligned} \tag{41}$$

Problem $\mathcal{P}2$-$E7$ is a non-convex one because of the unit modulus constraint on ϕ_n. In the following, the MM algorithm [43] is invoked for solving this problem, which has two steps. In the majorization step, we construct a continuous surrogate function $g(\boldsymbol{\phi}|\boldsymbol{\phi}^t)$, which represents the upperbound of $f(\boldsymbol{\phi})$. Then in the minimization step, $\boldsymbol{\phi}$ is updated by $\boldsymbol{\phi}^{t+1} \in \arg\min_{\boldsymbol{\phi}} g(\boldsymbol{\phi}|\boldsymbol{\phi}^t)$. As such, we may initialize $\boldsymbol{\phi}^0$ that satisfies the constraint (41), and then use the MM algorithm to generate a sequence of feasible vectors $\{\boldsymbol{\phi}^t\}$, where t refers to the iteration index. Now the surrogate function is constructed with the aid of the proposition below.

Proposition 4. *Denoting the maximum eigenvalue of $\boldsymbol{\Psi}$ by $\hat{\lambda}_{max}$ and given a solution $\boldsymbol{\phi}^t$ at the t-th iteration, we have the inequality below*

$$\begin{aligned} f(\boldsymbol{\phi}) \le \ & \boldsymbol{\phi}^H\hat{\lambda}_{max}\boldsymbol{I}_N\boldsymbol{\phi} - 2\Re\big\{\boldsymbol{\phi}^H(\hat{\lambda}_{max}\boldsymbol{I}_N - \boldsymbol{\Psi})\boldsymbol{\phi}^t\big\} \\ & + (\boldsymbol{\phi}^t)^H(\hat{\lambda}_{max}\boldsymbol{I}_N - \boldsymbol{\Psi})\boldsymbol{\phi}^t + 2\Re\big\{\boldsymbol{\phi}^H\boldsymbol{v}^*\big\}. \end{aligned} \tag{42}$$

Proof: See [26], [44]. ■

Here, the terms on the right side of (42) is defined by our surrogate function $g(\boldsymbol{\phi}|\boldsymbol{\phi}^t)$. Then, Problem $\mathcal{P}2$-$E7$ at the t-th iteration is reformulated as

$$\begin{aligned}\mathcal{P}2\text{-}E8: \min_{\boldsymbol{\phi}} \ & g(\boldsymbol{\phi}|\boldsymbol{\phi}^t) \\ \text{s.t.} \quad & |\phi_n| = 1, \quad \forall n \in \{1,2,\ldots,N\}. \end{aligned} \tag{43}$$

Since $(\boldsymbol{\phi}^t)^H(\hat{\lambda}_{max}\boldsymbol{I}_N - \boldsymbol{\Psi})\boldsymbol{\phi}^t$ is a constant for a given $\boldsymbol{\phi}^t$ and we have $\boldsymbol{\phi}^H\hat{\lambda}_{max}\boldsymbol{I}_N\boldsymbol{\phi} = M\hat{\lambda}_{max}$, Problem $\mathcal{P}2$-$E8$ can be equivalently written as

$$\begin{aligned}\mathcal{P}2\text{-}E9: \max_{\boldsymbol{\phi}} \ & \Re\Big\{\boldsymbol{\phi}^H\big[(\hat{\lambda}_{max}\boldsymbol{I}_N - \boldsymbol{\Psi})\boldsymbol{\phi}^t - \boldsymbol{v}^*\big]\Big\} \\ \text{s.t.} \quad & |\phi_n| = 1, \quad \forall n \in \{1,2,\ldots,N\}. \end{aligned} \tag{44}$$

Then, the optimal solution of Problem $\mathcal{P}2$-$E9$ is readily given by

$$\boldsymbol{\phi}^{t+1} = e^{j\arg\{(\hat{\lambda}_{max}\boldsymbol{I}_N - \boldsymbol{\Psi})\boldsymbol{\phi}^t - \boldsymbol{v}^*\}}. \tag{45}$$

Accordingly, the optimal solution to Problem $\mathcal{P}2$-$E6$ can be obtained as

$$\boldsymbol{\theta}^{t+1} = \arg\{(\hat{\lambda}_{max}\boldsymbol{I}_N - \boldsymbol{\Psi})\boldsymbol{\phi}^t - \boldsymbol{v}^*\}. \tag{46}$$

The termination condition of the MM algorithm is given by $\big|f(\boldsymbol{\phi}^{t+1}) - f(\boldsymbol{\phi}^t)\big|/f(\boldsymbol{\phi}^{t+1}) \le \epsilon$ or $t \ge t_{\text{MM}}^{\max}$. The procedure of solving Problem $\mathcal{P}2$-$E3$ is summarized in Algorithm 3.

The complexity of Algorithm 3 is dominated by its Step 2. Specifically, the complexity of calculating $\boldsymbol{W}^{(t_3+1)}$ by (31) is on the order of $\mathcal{O}\big(\max\{KM^3, KMN^2\}\big)$; the complexity of calculating $\boldsymbol{\Upsilon}^{(t_3+1)}$ by (32) is on the order of $\mathcal{O}(K)$. With regard to the calculation of $\boldsymbol{\theta}^{(t_3+1)}$ using the MM algorithm, the complexity of calculating the eigenvalue λ_{max} of $\boldsymbol{\Psi}$ is on the order of $\mathcal{O}(N^3)$, while for each iteration of the MM algorithm, the main complexity lies in the calculation of $\boldsymbol{\phi}^{t+1}$ in (45), whose complexity is on the order of $\mathcal{O}(N^2)$. Hence the complexity of the MM algorithm is $\mathcal{O}(N^3 + t_{\text{MM}}^{\max}N^2)$. Summing these three terms together, we obtain the total complexity of Algorithm 3 as $\mathcal{O}\big(\max\{N^3 + t_{\text{MM}}^{\max}N^2, KM^3, KMN^2\}\big)$. Finally, the complexity of Algorithm 2 is mainly dependent on updating $\boldsymbol{W}^{(t_2+1)}$ and $\boldsymbol{\theta}^{(t_2+1)}$ using Algorithm 3, because

Algorithm 4 Joint Optimization of $\boldsymbol{\ell}$, $\boldsymbol{f}^e$, $\boldsymbol{W}$ and $\boldsymbol{\theta}$

Input: $\boldsymbol{h}_{r,k}$, $\boldsymbol{h}_{d,k}$, $\boldsymbol{G}$, B, p_t, σ^2, ϖ_k, L_k, c_k, f_{total}^e, and ϵ
Output: Optimal $\boldsymbol{\ell}$, $\boldsymbol{f}^e$, $\boldsymbol{W}$ and $\boldsymbol{\theta}$
1. Initialization
initialize $t_4 = 0$, $\epsilon_4^{(0)} = 1$
initialize $\boldsymbol{\theta}^{(0)}$ satisfying (7a) and $\boldsymbol{f}^{e(0)}$ satisfying (7c) and (7d)
calculate $\boldsymbol{W}^{(0)}$ using (31)
2. Joint optimization of $\boldsymbol{\ell}$ and $\boldsymbol{f}^e$, given $\boldsymbol{W}^{(t_4)}$ and $\boldsymbol{\theta}^{(t_4)}$
calculate $\boldsymbol{\ell}^{(t_4+1)}$ and $\boldsymbol{f}^{e(t_4+1)}$ using Algorithm 1
3. Joint optimization of $\boldsymbol{W}$ and $\boldsymbol{\theta}$, given $\boldsymbol{\ell}^{(t_4+1)}$ and $\boldsymbol{f}^{e(t_4+1)}$
calculate $\boldsymbol{W}^{(t_4+1)}$ and $\boldsymbol{\theta}^{(t_4+1)}$ using Algorithm 2
4. Convergence checking
$\epsilon_4^{(t_4)} = \frac{\left|\text{obj}\left(\boldsymbol{\ell}^{(t_4+1)},\boldsymbol{f}^{e(t_4+1)},\boldsymbol{W}^{(t_4+1)},\boldsymbol{\theta}^{(t_4+1)}\right) - \text{obj}\left(\boldsymbol{\ell}^{(t_4)},\boldsymbol{f}^{e(t_4)},\boldsymbol{W}^{(t_4)},\boldsymbol{\theta}^{(t_4)}\right)\right|}{\text{obj}\left(\boldsymbol{\ell}^{(t_4+1)},\boldsymbol{f}^{e(t_4+1)},\boldsymbol{W}^{(t_4+1)},\boldsymbol{\theta}^{(t_4+1)}\right)}$
if $\epsilon_4^{(t_4)} > \epsilon$ && $t_4 < t_4^{\max}$ holds **then**
$t_4 = t_4 + 1$
Go to Step 2
else
integerize $\ell^{(t_4+1)}$ by (9)
Output the optimal $\boldsymbol{\ell}^*$, $\boldsymbol{f}^{e*}$, $\boldsymbol{W}^*$ and $\boldsymbol{\theta}^*$
end if

all other steps are given by explicit mathematical expressions.

C. Overall Algorithm to Solve Problem $\mathcal{P}0$

Based on the above discussions, we provide the detailed description of the BCD algorithm used for solving Problem $\mathcal{P}0$ in Algorithm 4. Note that a decreasing OF value of Problem $\mathcal{P}0$ is guaranteed in Step 2 and Step 3. Furthermore, the OF value has a lower bound due to the constraint on the total edge computing resources. Hence, Algorithm 4 is guaranteed to converge.

The computational complexity of Algorithm 4 is mainly dependent on its Step 2 and Step 3, whose complexities have been analyzed in the above subsections. Furthermore, the simulation results in Section V show that Algorithm 4 converges rapidly, which demonstrates the low complexity of our algorithms.

Algorithm 5 Joint Optimization of ℓ, $\boldsymbol{w}$ and $\boldsymbol{\theta}$ proposed for the single-user scenario

Input: $\boldsymbol{h}_r$, $\boldsymbol{h}_d$, $\boldsymbol{G}$, B, p_t, σ^2, L, c, f^e_{total}, and ϵ
Output: Optimal ℓ, $\boldsymbol{w}$ and $\boldsymbol{\theta}$
1. Initialization
initialize $t_5 = 0$, $\epsilon_5^{(0)} = 1$
initialize $\boldsymbol{\theta}^{(0)}$ satisfying (7a)
calculate $\boldsymbol{w}^{(0)}$ using (53)
2. Joint optimization of $\boldsymbol{w}$ and $\boldsymbol{\theta}$
repeat
- calculate $\boldsymbol{\theta}^{(t_5+1)}$ and $\boldsymbol{w}^{(t_5+1)}$ using (55) and (53), respectively
- $\epsilon_5^{(t_5)} = \frac{\left|\text{obj}\left(\boldsymbol{w}^{(t_5+1)},\boldsymbol{\theta}^{(t_5+1)}\right) - \text{obj}\left(\boldsymbol{w}^{(t_5)},\boldsymbol{\theta}^{(t_5)}\right)\right|}{\text{obj}\left(\boldsymbol{w}^{(t_5+1)},\boldsymbol{\theta}^{(t_5+1)}\right)}$, where the obj refers to the OF of Problem $\mathcal{P}3$-$E3$
- $t_5 = t_5 + 1$

until $\epsilon_5^{(t_5)} \le \epsilon \;||\; t_5 > t_5^{\max}$
3. Optimization of $\boldsymbol{\ell}$
calculate $\hat{\ell}^{(t_5+1)}$ using (49)
integerize $\ell^{(t_5+1)}$ by (9)

IV. Specific Case Study: The Single-Device Scenario

In order to fully characterize the IRS-aided MEC system, a special case is investigated in this section, where a single device is served by the MEC system. The optimization problem of the single-device scenario becomes much simpler for the following reasons. Firstly, the edge computing resource allocation no longer has to be considered, because all the edge computing resources can be assigned to this single device. Secondly, the sum-of-ratios form in Problem $\mathcal{P}2$-$E1$ becomes a single-ratio form, which implies that the optimization problem is more tractable. Thirdly, the multi-user interference does not have to be considered, when the detection vector and the IRS phase shift coefficient vector are optimized. The joint optimization is detailed as follows.

Problem $\mathcal{P}0$ can be simplified for the single-device scenario as

$$\begin{aligned}\mathcal{P}3: \min_{\boldsymbol{w},\boldsymbol{\theta},\ell}\; & D(\boldsymbol{w},\boldsymbol{\theta},\ell)\\ \text{s.t.}\quad & 0 \le \theta_n < 2\pi, \quad n = 1,2,\ldots,N, && \text{(47a)}\\ & \ell \in \{0,1,\ldots,L\}, && \text{(47b)}\end{aligned}$$

where the OF $D(\boldsymbol{w},\boldsymbol{\theta},\ell)$ becomes

$$D(\boldsymbol{w},\boldsymbol{\theta},\ell) = \max\left\{\frac{(L-\ell)c}{f^l}, \frac{\ell}{R(\boldsymbol{w},\boldsymbol{\theta})} + \frac{\ell c}{f^e_{\text{total}}}\right\}. \quad (48)$$

As illustrated in Proposition 1, for a given set of $\boldsymbol{w}$ and $\boldsymbol{\theta}$, $D(\boldsymbol{w},\boldsymbol{\theta},\ell)$ achieves its minimum value when ℓ is selected to ensure $\frac{(L-\ell)c}{f^l} = \frac{\ell}{R(\boldsymbol{w},\boldsymbol{\theta})} + \frac{\ell c}{f^e_{\text{total}}}$. Therefore, the optimal value of the relaxation of ℓ is given by

$$\hat{\ell}^* = \frac{LcRf^e_{\text{total}}}{f^e_{\text{total}}f^l + cR\left(f^e_{\text{total}} + f^l\right)}. \quad (49)$$

Then, Problem $\mathcal{P}3$ is reformulated as

$$\begin{aligned}\mathcal{P}3\text{-}E1: \min_{\boldsymbol{w},\boldsymbol{\theta}}\; & \frac{\ell}{R(\boldsymbol{w},\boldsymbol{\theta})} + \frac{\ell c}{f^e_{\text{total}}}\\ \text{s.t.}\quad & 0 \le \theta_n < 2\pi, \quad n = 1,2,\ldots,N, && \text{(50a)}\end{aligned}$$

which is equivalent to

$$\begin{aligned}\mathcal{P}3\text{-}E2: \max_{\boldsymbol{w},\boldsymbol{\theta}}\; & R(\boldsymbol{w},\boldsymbol{\theta})\\ \text{s.t.}\quad & 0 \le \theta_n < 2\pi, \quad n = 1,2,\ldots,N. && \text{(51a)}\end{aligned}$$

Substituting (4) and (5) into the OF of Problem $\mathcal{P}3$-$E2$, and taking several steps of mathematical manipulation, Problem $\mathcal{P}3$-$E2$ may be equivalently transformed into

$$\begin{aligned}\mathcal{P}3\text{-}E3: \max_{\boldsymbol{w},\boldsymbol{\theta}}\; & \frac{p_t\left|\boldsymbol{w}^H\left(\boldsymbol{h}_d + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_r\right)\right|^2}{\sigma^2|\boldsymbol{w}^H|^2}\\ \text{s.t.}\quad & 0 \le \theta_n < 2\pi, \quad n = 1,2,\ldots,N. && \text{(52a)}\end{aligned}$$

Again, the BCD technique is invoked for optimizing $\boldsymbol{w}$ and $\boldsymbol{\theta}$ in Problem $\mathcal{P}3$-$E3$. Specifically, given a $\boldsymbol{\theta}$, $\boldsymbol{w}$ can be optimized following the well-known maximum ratio combining (MRC) criterion [45], which is given by

$$\boldsymbol{w} = \sqrt{p_t}(\boldsymbol{h}_d + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_r)/\sigma, \quad (53)$$

while for a given $\boldsymbol{w}$, we have the following inequality for the OF of Problem $\mathcal{P}3$-$E3$,

$$\frac{p_t\left|\boldsymbol{w}^H\left(\boldsymbol{h}_d + \boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_r\right)\right|^2}{\sigma^2|\boldsymbol{w}^H|^2} \le \frac{p_t\left|\boldsymbol{w}^H\boldsymbol{h}_d\right|^2}{\sigma^2|\boldsymbol{w}^H|^2} + \frac{p_t\left|\boldsymbol{w}^H\boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_r\right|^2}{\sigma^2|\boldsymbol{w}^H|^2}. \quad (54)$$

The equality in (54) holds only when the IRS phase shift coefficient obeys $\arg\{\boldsymbol{w}^H\boldsymbol{h}_d\} = \arg\{\boldsymbol{w}^H\boldsymbol{G}\boldsymbol{\Theta}\boldsymbol{h}_r\}$. Accordingly, the reflection phase shift vector $\boldsymbol{\theta}$ may be readily obtained as

$$\boldsymbol{\theta} = \arg\{\boldsymbol{w}^H\boldsymbol{h}_d\} - \arg\{\text{diag}\{\boldsymbol{w}^H\boldsymbol{G}\}\boldsymbol{h}_r\}. \quad (55)$$

In Algorithm 5, we provide the overall algorithm that is used for solving our optimization problem for the single-device scenario.

The complexity of Algorithm 5 is dominated by calculating $\boldsymbol{\theta}^{(t_5+1)}$ and $\boldsymbol{w}^{(t_5+1)}$ using (55) and (53), whose complexities are on the order of $\mathcal{O}\left(\max\{MN, N^2\}\right)$ and of $\mathcal{O}(MN^2)$, respectively. Hence the complexity of Algorithm 5 is on the order of $\mathcal{O}(MN^2)$.

V. Numerical Results

In this section, the benefits of deploying the IRS in a MEC system are evaluated, relying on our algorithms developed in Section III and IV. We consider a single-cell MEC system for both the single-device and two-device as well as multi-device scenarios. As shown in Fig. 3, the AP's coverage radius is $R = 300$ m and the IRS is deployed at the cell edge. The location of the device is specified both by d and by $\overline{d}$ in the single-device scenario, while in the two-device scenario, the devices' locations are specified by $(d_1, \overline{d}_1)$ and $(d_2, \overline{d}_2)$, respectively. Furthermore, in the multiple-device scenario, it is assumed that the devices are uniformly distributed within a circle, whose size and location are prescribed by its radius r, as well as d and $\overline{d}$, respectively. The default value of these parameters are set in the "Location model" block of Table II. As for the communications channel, we consider both the small scale fading and the large scale path loss. Specifically, the small scale fading is i.i.d. and obeys the complex Gaussian distribution associated with zero mean and unit variance, while the path loss in dB is given by

$$\mathrm{PL} = \mathrm{PL}_0 - 10\alpha \log_{10}\left(\frac{d}{d_0}\right), \tag{56}$$

where PL_0 is the path loss at the reference distance d_0; d and α represent the distance of the communications link and its path loss exponent, respectively. Here we use α_{ua}, α_{ui} and α_{ia} to denote the path loss exponent of the link between the device and the AP, that of the link between the device and the IRS, as well as that of the link between the IRS and the AP, respectively. The zero-mean additive white Gaussian noise associated with the variable of σ^2 is imposed on the off-loaded signal. The default settings of these parameters are specified in the "Communications model" block of Table II. The variables L_k, c_k and f_k^l obey the uniform distribution, whose ranges are given in the "Computing model" block of Table II.

Table II: Default simulation parameter setting

Description	Parameter and Value
Location model	$R = 300$ m $\overline{d} = \overline{d}_1 = \overline{d}_2 = 10$ m
Communication model	Bandwidth $= 1$ MHz $\mathrm{PL}_0 = 30$ dB, $d_0 = 1$ m $\alpha_{ua} = 3.5$, $\alpha_{ui} = 2.2$, $\alpha_{ia} = 2.2$ $M = 5$ $p_t = 1$ mW $\sigma^2 = 3.98 \times 10^{-12}$ mW
Computing model	$L_k = [250, 350]$ Kb $c_k = [700, 800]$ cycle/bit $f_k^l = [4, 6] \times 10^8$ cycle/s
Weight	$\varpi_k = 1/K$
Convergence criterion	$\epsilon = 0.001$

The following subsections detail our simulation results, in terms of the properties of our proposed algorithm and of the latency performance in both the single-device, and two-device, as well as multi-device scenarios in various simulation environments. The following three schemes are considered:

- *With IRS:* The off-loading volume, edge computing resource allocation, MUD matrix, and IRS phase shift are optimized relying on Algorithm 4 and Algorithm 5 in the multi-device and single-device scenarios, respectively.
- *RandPhase:* The off-loading volume, edge computing resource allocation, as well as MUD matrix are optimized using Algorithm 4 and Algorithm 5 in multi-device and single-device scenarios, respectively, while skipping the step of designing the IRS phase shift, which is randomly set obeying the uniform distribution in the range of $[0, 2\pi)$.
- *Without IRS:* The composite channel $\boldsymbol{G\Theta h}_{r,k}$ taking into account the IRS is set to 0. The off-loading volume, edge computing resource allocation, and the MUD matrix are designed following Algorithm 4 and Algorithm 5 in the multi-device and single-device scenarios, respectively.

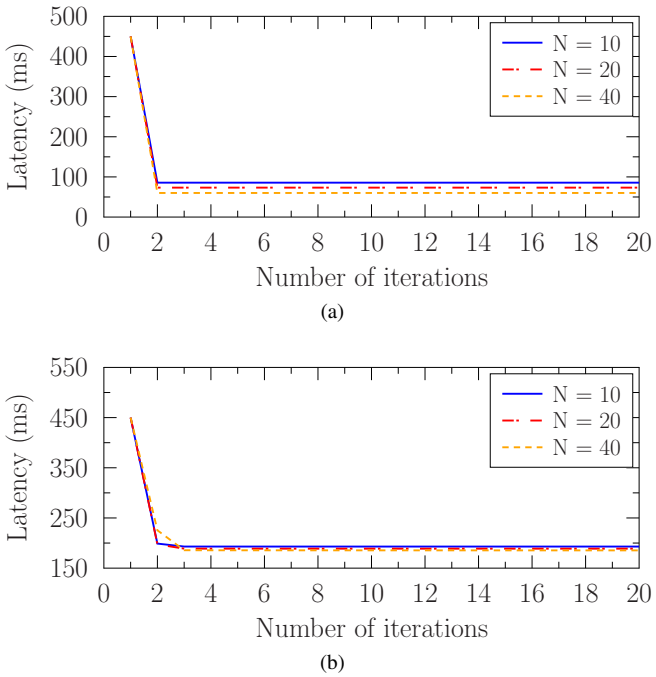

Figure 2: Convergence of the algorithms for (a) the single-device scenario using Algorithm 5 and (b) the two-device scenario using Algorithm 4. The parameters are set as follows: $f_{\text{total}}^e = 50 \times 10^9$ cycle/s; $c_k = 750$ cycle, $L_k = 300$ Kb, and $f_k^l = 0.5 \times 10^9$ cycle/s for all devices. (a): $d = 280$ m; (b): $d_1 = d_2 = 280$ m.

A. Properties of the Proposed Algorithms

In this subsection, the properties of Algorithm 4 and 5 are investigated, with the aid of numerical results.

1) Convergence: Fig. 2 shows the device-average latency versus the number of iterations under various settings of the IRS phase shift number, i.e. $N = 10$, 20, and 40, for both the single-device and multi-device scenarios. We have the following two observations. Firstly, a larger number of phase shifts leads to a slightly slower convergence, especially for the multi-device scenario. This is because more optimizing variables are involved. Secondly, the proposed algorithms are capable of achieving a convergence within 5 iterations, which validates its practical implementation.

2) Impact of the Initialization Settings: As elaborated on in Remark 1, locally optimal results are provided by our proposed

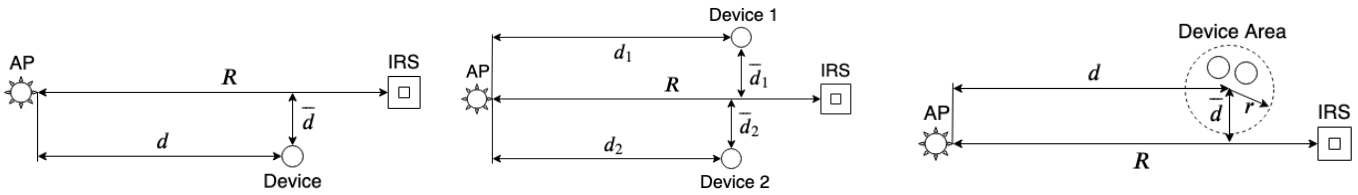

Figure 3: Top-view setting of (a) the single-device scenario, (b) the two-device scenario, and (c) the multiple-device scenario.

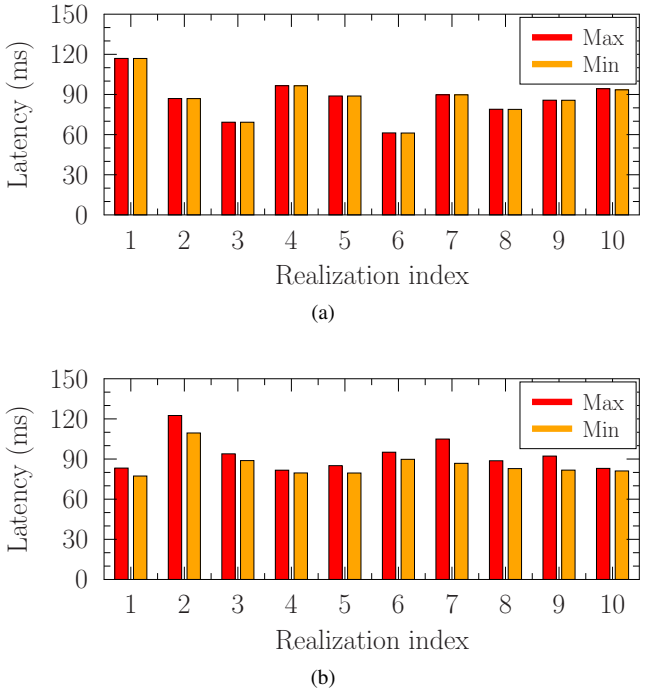

Figure 4: Simulation results of the the maximum and minimum latency versus the realization index obtained under 100 random initialization settings for (a) the single-device scenario using Algorithm 5 and (b) the two-device scenario using Algorithm 4. "Max" and "Min" refer to the maximum and minimum value, respectively. The parameters are set as follows: $N = 40$. (a): $d = 280$ m; (b): $d_1 = d_2 = 280$ m.

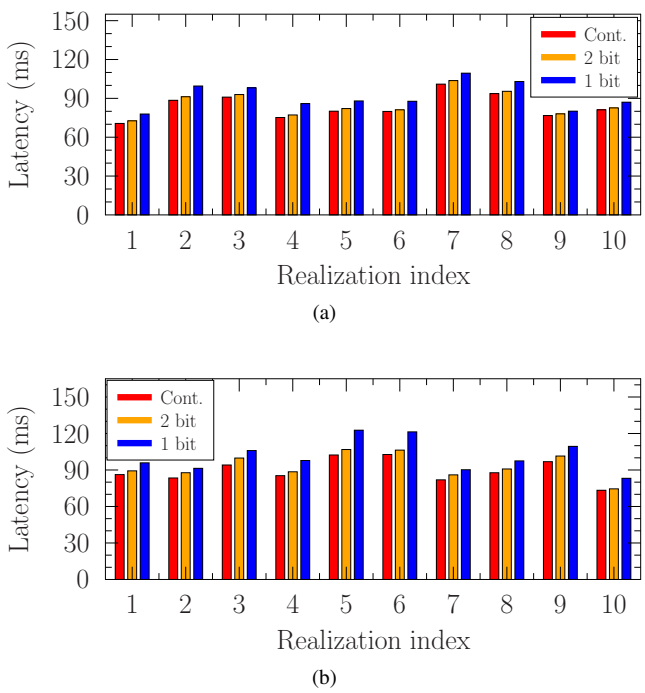

Figure 5: Simulation results of the latency versus the realization index under different assumptions of IRS phase shifts for (a) the single-device scenario and (b) the two-device scenario. "Cont.", "1-bit" and "2-bit" refer to the assumptions of continuous, 1-bit, and 2-bit phase shifts, respectively. The parameters are set as follows: $N = 40$. (a): $d = 280$ m; (b): $d_1 = d_2 = 280$ m.

algorithms. Hence, the results obtained are directly dependent on the initialization settings of Algorithm 4 and 5. In order to clarify its impact, Fig. 4 presents the latency performance under different initialization settings both for single- and multi-device scenarios. Specifically, for each realization of the wireless channels and computing tasks to be processed, 100 locally optimal results are obtained using our proposed algorithms, where each of the initializations is randomly set. Among these locally optimal results, the maximum latency value that can be deemed to be the worst-case result using our proposed algorithms is labeled as "Max" in Fig. 4, while the minimum latency value is labeled by "Min" in Fig. 4 which may resemble the globally optimal result. It is shown that these two values are almost identical for the single-device scenario, while their gap ranges from 2% to 17% in the multi-device scenario, which implies that our proposed algorithms are capable of approaching the optimal performance.

3) Impact of the Phase Quantization: Due to the associated hardware limitation, only a limited number of discrete IRS phase shifts can be provided in practice [23], which prohibits the direct implementation of our proposed algorithms. An intuitive practical solution to this issue is to round the continuous phase shift obtained to its nearest discrete phase shift. Naturally, a performance loss is imposed, owing to the associated quantization effect. Fig. 5 evaluates the impact of phase quantization on the latency, where three practical assumptions are considered. Specifically, under the assumption of continuous phase shifts, the phase shift of each IRS element can be set as an arbitrary value in the interval of $[0, 2\pi]$; Determined by a 1-bit control signal, the phase shift of each IRS element has to be either 0 or π under the assumption of 1-bit phase shift; for a 2-bit control signal, the phase shift of each IRS element has to be one of the values in the set of $\{0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}\}$. Particular to the schemes under discrete phase shift assumptions, the values of $\boldsymbol{W}$, $\boldsymbol{\ell}$, and $\boldsymbol{f}^e$ are updated

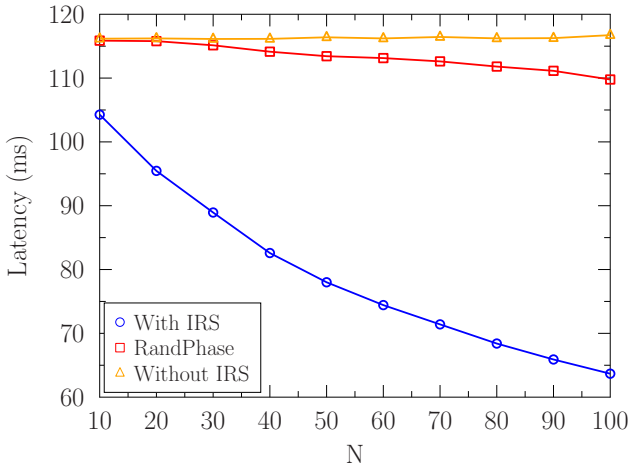

Figure 6: Simulation results of the latency versus the number of the IRS elements in the single-device scenario, where we set $d = 280$ m and $f^e_{\text{total}} = 50 \times 10^9$ cycle/s.

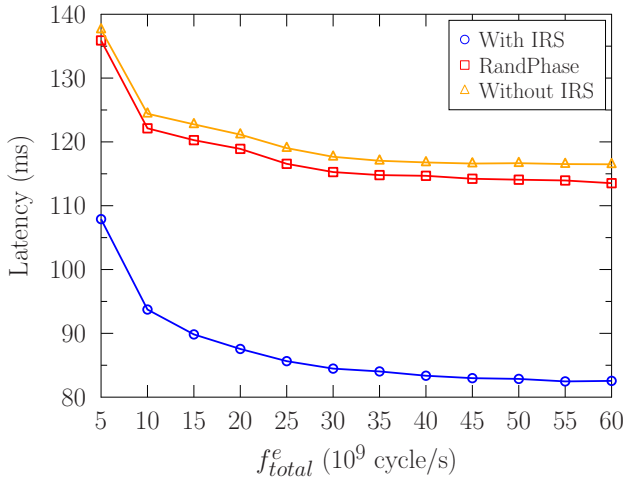

Figure 7: Simulation results of the latency versus the edge computing capability in the single-device scenario, where we set $d = 280$ m and $N = 40$.

based on the quantized phase shifts and then the latency is calculated accordingly. We have the following observations. Firstly, as expected, the latency decreases upon increasing the number of discrete phase shifts. Secondly, the performance gap between the schemes under the assumptions of continuous phase shifts and 2-bit phase shifts ranges from 1% to 5%, which implies that the quantization loss becomes negligible for as few as four phase shifts in practice.

B. Single-Device Scenario

Fig. 6-8 present the latency versus various parameter settings in the single-device scenario, discussed as follows.

1) Impact of the Number of Reflecting Elements: Fig. 6 presents the latency versus the number of the reflecting elements, for the various phase shift design schemes. Our observations are as follows. Firstly, the performance gap between the schemes "Without IRS" and "RandPhase" becomes higher upon increasing the number of reflecting elements, which implies that the IRS is capable of assisting the computation off-loading even without carefully designing the phase shift. This is because the received SINR can be improved by deploying an IRS for computation off-loading. The gain was termed as the virtual array gain in Section I. Secondly, the performance gain of the scheme "With IRS" over the scheme "RandPhase" is around 11 ms when we set $N = 10$, while it becomes 46 ms when we have $N = 100$. This implies that a sophisticated design of the IRS phase shift response provides a beamforming gain, and that increasing the number of IRS elements leads to a higher reflection-based beamforming gain. Combining these two types of gains together, IRSs are capable of efficiently reducing the latency in MEC systems.

2) Impact of the Edge Computing Capability: Fig. 7 shows the latency versus the edge computing capability, for various IRS phase shift schemes. Our observations are as follows. For all these three schemes, the increase of f^e_{total} drastically reduces the latency when f^e_{total} is of a small value, while the reduction of the latency becomes smaller when f^e_{total} reaches a certain threshold value, say 30×10^9 cycle/s. This is because

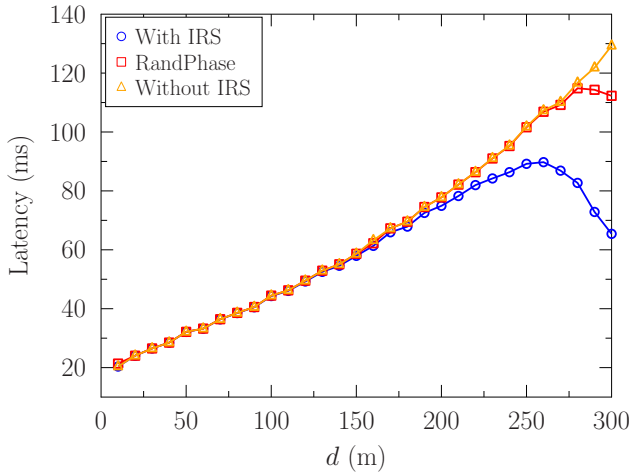

Figure 8: Simulation results of the latency versus the device location in the single-device scenario, where we set $N = 40$ and $f^e_{\text{total}} = 50 \times 10^9$ cycle/s.

the latency imposed by the edge computing dominates when f^e_{total} is of a small value, whereas the latency imposed by computation off-loading plays a dominant role when f^e_{total} reaches a high value. Therefore, it is not necessary to equip the edge computing node with an extremely powerful computing capability for latency minimization.

3) Impact of the Device Location: Fig. 8 depicts the latency versus the device location, equipped with various IRS phase shift schemes. Our observations are as follows. In the case where no IRS is employed, the latency increases upon increasing the distance between the AP and the device. In the case where the IRS's phase shift is randomly set, the advantage of using the IRS becomes visible when the distance between the device and the IRS is less than 20 m. By contrast, the benefit of the IRS becomes notable for a much larger coverage of 100 m for the "With IRS" scheme. This observation implies that a sophisticated design of the IRS phase shift response is capable of extending the coverage of the IRS. Furthermore, the latency reaches its maximum value at $d = 260$ m and thereafter becomes smaller for the "With IRS" scheme. This is

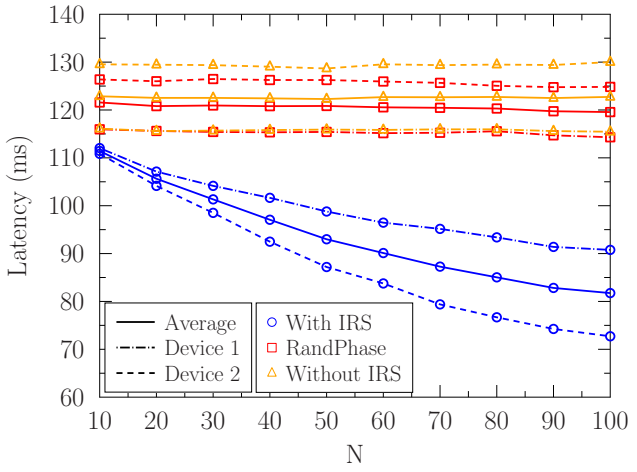

Figure 9: Simulation results of the latency versus the number of the IRS elements in the two-device scenario, where we set $d_1 = 260$ m, $d_2 = 280$ m, and $f^e_{\text{total}} = 50 \times 10^9$ cycle/s.

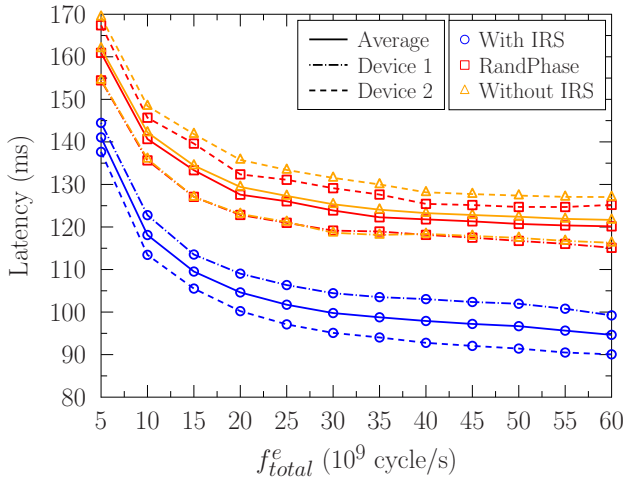

Figure 10: Simulation results of the latency versus the edge computing resource in the two-device scenario, where we set $d_1 = 260$ m, $d_2 = 280$ m, and $N = 40$.

because the direct device-AP link dominates the computation off-loading when the device's location obeys $d \leq 260$ m, while the composite device-IRS-AP link plays a dominant role, when we have $d \geq 260$ m. This observation further consolidates that a higher gain can be achieved in the near-IRS area, where the composite device-IRS-AP link dominates the computation off-loading.

C. Multi-Device Scenario

Fig. 9-12 present the latency in the two-device scenario, and Fig. 13-14 show the latency in the multiple-device scenario, which are discussed as follows.

1) Impact of the Number of IRS Elements: Fig. 9 depicts the latency versus the number of the IRS elements in the two-device scenario, equipped with various phase shift schemes. Apart from the insights obtained in the single-device scenario, we also have the following observations. Firstly, Device 2 outperforms Device 1 for the "With IRS" scheme, whilst Device 1 has a lower latency both for the "Without IRS" and "RandPhase" schemes compared to Device 2. This is in accordance with the comparative relationship between the devices located at $d = 260$ m and at $d = 280$ m in terms of the latency using those three phase shift schemes, as shown in Fig. 8. This also implies that IRSs may change the latency ranking of the devices in MEC systems. Secondly, upon increasing the number of IRS elements, Device 2 obtains a higher gain than Device 1. This is because Device 2 is located closer to the IRS, where the composite device-IRS-AP channel dominates the computation off-loading. Again, this implies that given a specific path loss exponent, a higher array and passive beamforming gain may be achieved if the device is located closer to the IRS.

2) Impact of the Edge Computing Capability: Fig. 10 presents the latency versus the edge computing capability in the multi-device scenario, equipped with various phase shift schemes. This scenario follows similar trends to the single-device case illustrated in Fig. 7.

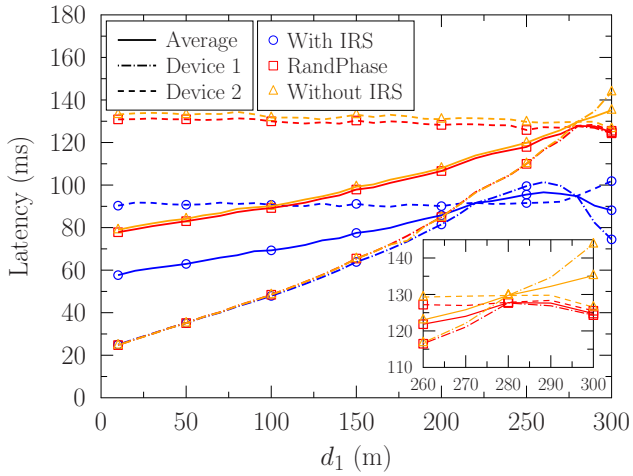

Figure 11: Simulation results of the latency versus the user location in the two-device scenario, where we set $d_2 = 280$ m, $N = 40$, and $f^e_{\text{total}} = 50 \times 10^9$ cycle/s.

3) Impact of the Device Location: Fig. 11 plots the latency versus the location of Device 1, while fixing the location of the AP, the IRS, and Device 2. As for the "With IRS" scheme, the curve of Device 1's latency intercepts that of Device 2 at $d_1 = 220$ m and $d_1 = 280$ m, which implies that the devices at these two locations have the same channel gain. In other words, the IRS is capable of assisting the device at $d = 280$ m to achieve the same latency as the device at $d = 220$ m. Note that the specific values of these two equivalent-latency locations are dependent on the specific values of the path loss exponents of the device-IRS, IRS-AP, and device-AP channels, as presented below.

4) Impact of the Path Loss Exponent: Fig. 12 illustrates the latency versus the path loss exponent value associated with the IRS. It can be observed that the intercept point disappears, when α_{IRS} is changed for the "With IRS" scheme. Furthermore, the latency of devices increases upon increasing α_{IRS}. This is because higher α_{IRS} leads to a lower array and beamforming gain by the IRS. This provides important insights

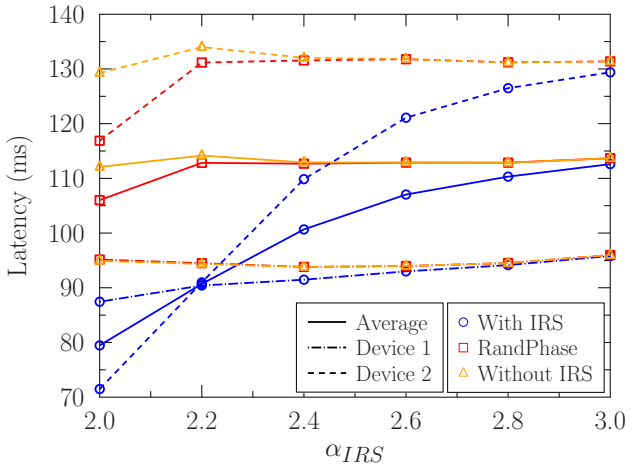

Figure 12: Simulation results of the latency versus the path loss exponent in the two-device scenario, where we set $\alpha_{ui} = \alpha_{ia} = \alpha_{\text{IRS}}$. The parameters are set as follows: $d_1 = 220$ m, $d_2 = 280$ m, $N = 40$, and $f^e_{\text{total}} = 50 \times 10^9$ cycle/s.

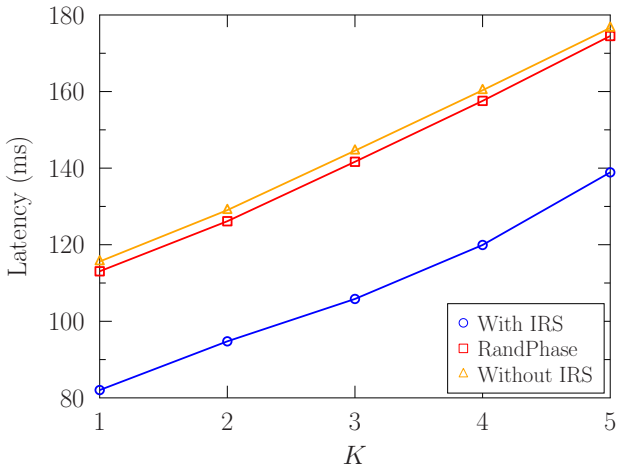

Figure 13: Simulation results of the device-average latency versus the number of devices K. The parameters are set as follows: $d = 280$ m, $\overline{d} = 10$ m, $r = 10$ m, $N = 40$, and $f^e_{\text{total}} = 50 \times 10^9$ cycle/s.

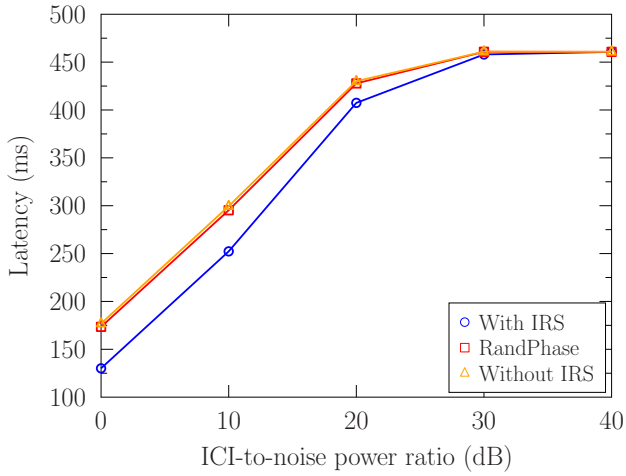

Figure 14: Simulation results of the device-average latency versus the ICI-to-noise ratio. The parameters are set as follows: $K = 3$, $d = 280$ m, $\overline{d} = 10$ m, $r = 10$ m, $N = 40$, and $f^e_{\text{total}} = 50 \times 10^9$ cycle/s.

for engineering design: the location of the IRS should be carefully selected to avoid obstacles, for achieving a lower α_{ui} and α_{ia}.

5) Impact of the Number of Devices: Fig. 13 shows the latency versus the number of devices in the cycle in multi-device scenario. It can be readily observed that the device-average latency increases upon increasing the number of devices in the IRS-aided MEC system. This is partially because of the reduced edge computational resources allocated to each device and partially due to the reduced beamforming gain achieved at each device. The former issue may be overcome by equipping the edge node with more powerful computing capability, while the latter problem can be solved by deploying more IRSs in the MEC system for forming stronger beams. Nonetheless, compared to the "Without IRS" scheme, our "With IRS" scheme is capable of reducing the device-average latency from 177 ms to 139 ms, when we have 5 devices in the MEC system. This again validates the benefits of our proposed system.

6) Impact of Inter-Cell Interference: In realistic scenarios, inter-cell interference (ICI) also degrades computation off-loading. To quantify the impact of ICI, Fig. 14 presents the latency versus the ICI-to-noise power ratio, where the BS is assumed to know the power of the received interference but not the specific signal transmitted from other cells. Observe that the benefit of employing IRSs in MEC systems decreases upon increasing the ICI-to-noise power ratio. To elaborate, when computation off-loading is used in the face of strong ICI, the fraction of tasks that can be off-loaded becomes marginal. In this case, the wireless devices have to rely on their own computing capabilities. In other words, the potential of IRSs may not be fully exploited. This observation suggests an important insight for engineering design: the spectrum allocation of adjacent cells has to be carefully managed for minimizing the ICI in IRS-aided MEC systems.

VI. Conclusions

In order to reduce the computational latency, an IRS was proposed for employment in MEC systems. Based on this model, a latency-minimization problem was formulated, subject to practical constraints on the total edge computing capability and IRS phase shifts. Sophisticated algorithms were developed for optimizing both the computing and communications settings. The benefits of using IRSs in the MEC system were evaluated under various simulation environments. Quantitatively, the device-average computational latency was reduced from 177 ms to 139 ms, compared to the conventional MEC system operating without IRSs in a single cell associated with the cell radius of 300 m, a 5-antenna access point and 5 active devices. Furthermore, the rapid convergence of our proposed algorithm was confirmed numerically, which validates their benefits. As our future work, an energy-minimization based design will be conceived for IRS-aided MEC systems.

APPENDIX A
THE PROOF OF PROPOSITION 1

$\hat{\ell}_k \in [0, L_k]$ is used to represent the relaxation [46] of the integer value $\ell_k \in \{0, 1, \ldots, L_k\}$. Furthermore, given the values of $\boldsymbol{W}, \boldsymbol{\theta}$ and $\boldsymbol{f}^e$, we define the delay associated with $\hat{\ell}_k$ to be $\hat{D}_k(\hat{\ell}_k) \triangleq \max\left\{D_k^l(\hat{\ell}_k), D_k^e(\hat{\ell}_k)\right\}$, which can be reformulated from (6) as a segmented form below

$$\hat{D}_k(\hat{\ell}_k) = \begin{cases} \frac{(L_k - \hat{\ell}_k)c_k}{f_k^l}, & 0 \le \hat{\ell}_k \le \frac{L_k c_k R_k f_k^e}{f_k^e f_k^l + c_k R_k (f_k^l + f_k^e)}, \\ \frac{\hat{\ell}_k}{R_k} + \frac{\hat{\ell}_k c_k}{f_k^e}, & \frac{L_k c_k R_k f_k^e}{f_k^e f_k^l + c_k R_k (f_k^l + f_k^e)} < \hat{\ell}_k \le L_k. \end{cases} \tag{57}$$

A glance at (57) reveals that $\hat{D}_k(\hat{\ell}_k)$ decreases upon increasing $\hat{\ell}_k$ in the range of $\hat{\ell}_k \in \left[0, \frac{L_k c_k R_k f_k^e}{f_k^e f_k^l + c_k R_k \left(f_k^e + f_k^l\right)}\right]$, while $\hat{D}_k(\hat{\ell}_k)$ increases upon increasing $\hat{\ell}_k$ in the range of $\hat{\ell}_k \in \left[\frac{L_k c_k R_k f_k^e}{f_k^e f_k^l + c_k R_k \left(f_k^e + f_k^l\right)}, L_k\right]$. Therefore, it is readily inferred that $\hat{D}_k(\hat{\ell}_k)$ achieves its minimum value when we set $\hat{\ell}_k = \frac{L_k c_k R_k f_k^e}{f_k^e f_k^l + c_k R_k \left(f_k^e + f_k^l\right)}$, which is denoted by $\hat{\ell}_k^*$. Bearing in mind that the optimal value of ℓ_k has to be an integer, it may be obtained by carrying out the operation $\ell_k^* = \underset{\hat{\ell} \in \left\{\lfloor \hat{\ell}_k^* \rfloor, \lceil \hat{\ell}_k^* \rceil\right\}}{\arg\min} D_k(\hat{\ell}_k)$. This completes the proof.

APPENDIX B
THE PROOF OF PROPOSITION 2

Let us denote the second derivative of the OF of Problem $\mathcal{P}1$-E with respect to f_k^e by $\Phi_{1\text{-}E}$, which is calculated as

$$\Phi_{1\text{-}E} = \frac{2\varpi_k L_k c_k^3 R_k^2 (f_k^l + c_k R_k)}{\left[f_k^e f_k^l + c_k R_k \left(f_k^e + f_k^l\right)\right]^3}. \tag{58}$$

Since the values of ϖ_k, c_k, R_k, f_k^l are all positive and we have $L_k \ge 0$, $f_k^e \ge 0$, it may be readily demonstrated that $\Phi_{1\text{-}E} \ge 0$. Hence the OF is a convex function with respect to f_k^e. Furthermore, the constraint functions (7c) and (7d) are all of linear forms. Hence, Problem $\mathcal{P}1$-E is shown to be a strictly convex problem.

APPENDIX C
THE PROOF OF PROPOSITION 3

The Lagrangian of Problem $\mathcal{P}2$-$E2$ is given by

$$\mathcal{L}(\boldsymbol{W}, \boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{\lambda}) = \sum_{k=1}^{K} \beta_k + \sum_{k=1}^{K} \lambda_k \left[\varpi_k \ell_k - \beta_k R_k(\boldsymbol{w}_k, \boldsymbol{\theta})\right] \tag{59}$$

where $\{\lambda_k\}$ is the non-negative Lagrange multiplier. If $(\boldsymbol{W}^*, \boldsymbol{\theta}^*, \boldsymbol{\beta}^*)$ is the solution of Problem $\mathcal{P}2$-$E2$, there exists $\boldsymbol{\lambda}^*$ satisfying the following KKT conditions

$$\frac{\partial \mathcal{L}}{\partial \theta_k} = -\lambda_k^* \beta_k^* \triangledown R_k(\boldsymbol{w}_k^*, \boldsymbol{\theta}^*) = 0, \quad k = 1, 2, \ldots, K, \tag{60}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{w}_k} = -\lambda_k^* \beta_k^* \triangledown R_k(\boldsymbol{w}_k^*, \boldsymbol{\theta}^*) = 0, \quad k = 1, 2, \ldots, K, \tag{61}$$

$$\frac{\partial \mathcal{L}}{\partial \beta_k} = 1 - \lambda_k^* R_k(\boldsymbol{w}_k^*, \boldsymbol{\theta}^*) = 0, \quad k = 1, 2, \ldots, K, \tag{62}$$

$$\lambda_k^* \left[\varpi_k \ell_k - \beta_k^* R_k(\boldsymbol{w}_k^*, \boldsymbol{\theta}^*)\right] = 0, \quad k = 1, 2, \ldots, K, \tag{63}$$

$$\lambda_k^* \ge 0, \quad k = 1, 2, \ldots, K, \tag{64}$$

$$\varpi_k \ell_k - \beta_k^* R_k(\boldsymbol{w}_k^*, \boldsymbol{\theta}^*) \le 0, \quad k = 1, 2, \ldots, K, \tag{65}$$

$$0 \le \theta_k^* \le 2\pi, \quad k = 1, 2, \ldots, K. \tag{66}$$

Since we have $R_k(\boldsymbol{w}_k, \boldsymbol{\theta}) > 0$, (62) is equivalent to

$$\lambda_k^* = \frac{1}{R_k(\boldsymbol{w}_k^*, \boldsymbol{\theta}^*)}, \quad \forall k \in \{1, 2, \ldots, K\}, \tag{67}$$

and then (63) is equivalently written as

$$\beta_k^* = \frac{\varpi_k \ell_k}{R_k(\boldsymbol{w}_k^*, \boldsymbol{\theta}^*)}, \quad \forall k \in \{1, 2, \ldots, K\}. \tag{68}$$

Furthermore, Eq. (60), (61) and (66) are exactly the KKT conditions of Problem $\mathcal{P}2$-$E3$, when we set $\boldsymbol{\lambda} = \boldsymbol{\lambda}^*$ and $\boldsymbol{\beta} = \boldsymbol{\beta}^*$. This proves the first conclusion of Proposition 3. Following the same procedure, the second conclusion of Proposition 3 may also be readily shown.

REFERENCES

[1] M. R. Palattella, M. Dohler, A. Grieco, G. Rizzo, J. Torsner, T. Engel, and L. Ladid, "Internet of things in the 5G era: Enablers, architecture, and business models," *IEEE J. Sel. Areas Commun.*, vol. 34, pp. 510–527, Mar. 2016.

[2] FP7 European Project (2012), "Distributed computing, storage and radio resource allocation over cooperative femtocells (TROPIC)." [Online]. http://www.ict-tropic.eu.

[3] 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, pp. 45–55, Nov. 2014.

[4] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, pp. 637–646, March 2016.

[5] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surv. Tutor.*, vol. 19, pp. 2322–2358, April 2017.

[6] 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, pp. 4569–4581, Sep. 2013.

[7] Y. Wang, M. Sheng, X. Wang, L. Wang, and J. Li, "Mobile-edge computing: Partial computation offloading using dynamic voltage scaling," *IEEE Trans. Commun.*, vol. 64, pp. 4268–4282, Oct. 2016.

[8] J. Ren, G. Yu, Y. Cai, and Y. He, "Latency optimization for resource allocation in mobile-edge computation offloading," *IEEE Trans. Wireless Commun.*, vol. 17, pp. 5506–5519, Aug. 2018.

[9] T. Bai, J. Wang, Y. Ren, and L. Hanzo, "Energy-efficient computation offloading for secure UAV-edge-computing systems," *IEEE Trans. Veh. Technol.*, vol. 68, pp. 6074–6087, June 2019.

[10] S. Sardellitti, 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, pp. 89–103, June 2015.

[11] 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, pp. 2795–2808, Oct. 2015.

[12] X. Lyu, W. Ni, H. Tian, R. P. Liu, X. Wang, G. B. Giannakis, and A. Paulraj, "Optimal schedule of mobile edge computing for Internet of Things using partial information," *IEEE J. Sel. Areas Commun.*, vol. 35, pp. 2606–2615, Nov. 2017.

[13] Y. Dai, D. Xu, S. Maharjan, and Y. Zhang, "Joint computation offloading and user association in multi-task mobile edge computing," *IEEE Trans. Veh. Technol.*, vol. 67, pp. 12313–12325, Dec. 2018.

[14] T. Ouyang, Z. Zhou, and X. Chen, "Follow me at the edge: Mobility-aware dynamic service placement for mobile edge computing," *IEEE J. Sel. Areas Commun.*, vol. 36, pp. 2333–2345, Oct 2018.

[15] T. J. Cui, M. Q. Qi, X. Wan, J. Zhao, and Q. Cheng, "Coding metamaterials, digital metamaterials and programmable metamaterials," *Light: Science & Applications*, vol. 3, no. 10, p. e218, 2014.

[16] M. Di Renzo, M. Debbah, D.-T. Phan-Huy, *et al.*, "Smart radio environments empowered by reconfigurable AI meta-surfaces: an idea whose time has come," *EURASIP J. Wireless Commun. Netw.*, vol. 2019, pp. 1–20, May 2019.

[17] M. Sheng, Y. Wang, X. Wang, and J. Li, "Energy-efficient multiuser partial computation offloading with collaboration of terminals, radio access network, and edge server," *IEEE Trans. Commun.*, vol. 68, pp. 1524–1537, March 2020.

[18] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Trans. Wireless Commun.*, vol. 19, pp. 2022–2035, March 2020.

[19] Y. Han, W. Tang, S. Jin, C.-K. Wen, and X. Ma, "Large intelligent surface-assisted wireless communication exploiting statistical CSI," *IEEE Trans. Veh. Technol.*, vol. 68, pp. 8238–8242, Aug. 2019.

[20] B. Zheng and R. Zhang, "Intelligent reflecting surface-enhanced OFDM: Channel estimation and reflection optimization." [Online]. Available: https://arxiv.org/abs/1909.03272.

[21] S. Abeywickrama, R. Zhang, and C. Yuen, "Intelligent reflecting surface: Practical phase shift model and beamforming optimization." [Online]. Available: https://arxiv.org/abs/1907.06002.

[22] Q. Wu and R. Zhang, "Intelligent reflecting surface enhanced wireless network via joint active and passive beamforming," *IEEE Trans. Wireless Commun.*, pp. 1–1, 2019.

[23] Q. Wu and R. Zhang, "Beamforming optimization for intelligent reflecting surface with discrete phase shifts." [Online]. Available: https://arxiv.org/abs/1810.03961.

[24] H. Guo, Y.-C. Liang, J. Chen, and E. G. Larsson, "Weighted sum-rate optimization for intelligent reflecting surface enhanced wireless networks." [Online]. Available: https://arxiv.org/abs/1905.07920.

[25] J. Ye, S. Guo, and M.-S. Alouini, "Joint reflecting and precoding designs for SER minimization in reconfigurable intelligent surfaces assisted MIMO systems." [Online]. Available: https://arxiv.org/abs/1906.11466.

[26] C. Pan, H. Ren, K. Wang, W. Xu, M. Elkashlan, A. Nallanathan, and L. Hanzo, "Intelligent reflecting surface for multicell MIMO communications." [Online]. Available: https://arxiv.org/abs/1907.10864.

[27] G. Zhou, C. Pan, H. Ren, K. Wang, W. Xu, and A. Nallanathan, "Intelligent reflecting surface aided multigroup multicast MISO communication systems." [Online]. Available: https://arxiv.org/abs/1909.04606.

[28] C. Huang, G. C. Alexandropoulos, C. Yuen, and M. Debbah, "Indoor signal focusing with deep learning designed reconfigurable intelligent surfaces." [Online]. Available: https://arxiv.org/abs/1905.07726.

[29] Y. Yang, B. Zheng, S. Zhang, and R. Zhang, "Intelligent reflecting surface meets OFDM: Protocol design and rate maximization." [Online]. Available: https://arxiv.org/abs/1906.09956.

[30] M. Cui, G. Zhang, and R. Zhang, "Secure wireless communication via intelligent reflecting surface," *IEEE Wireless Commun. Lett.*, pp. 1–1, 2019.

[31] H. Shen, W. Xu, S. Gong, Z. He, and C. Zhao, "Secrecy rate maximization for intelligent reflecting surface assisted multi-antenna communications," *IEEE Commun. Lett.*, vol. 23, pp. 1488–1492, Sep. 2019.

[32] D. Xu, X. Yu, Y. Sun, D. W. K. Ng, and R. Schober, "Resource allocation for secure IRS-assisted multiuser MISO systems." [Online]. Available: https://arxiv.org/abs/1907.03085.

[33] J. Chen, Y.-C. Liang, Y. Pei, and H. Guo, "Intelligent reflecting surface: A programmable wireless environment for physical layer security." [Online]. Available: https://arxiv.org/abs/1905.03689.

[34] Q. Wu and R. Zhang, "Weighted sum power maximization for intelligent reflecting surface aided SWIPT." [Online]. Available: https://arxiv.org/abs/1907.05558.

[35] C. Pan, H. Ren, K. Wang, M. Elkashlan, A. Nallanathan, J. Wang, and L. Hanzo, "Intelligent reflecting surface enhanced MIMO broadcasting for simultaneous wireless information and power transfer." [Online]. Available: https://arxiv.org/abs/1908.04863.

[36] M. Bennis, M. Debbah, and H. V. Poor, "Ultrareliable and low-latency wireless communication: Tail, risk, and scale," *Proc. IEEE*, vol. 106, pp. 1834–1853, Oct. 2018.

[37] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.

[38] Y. Jong, "An efficient global optimization algorithm for nonlinear sum-of-ratios problem." [Online]. Available: http://www.optimization-online.org/DB_FILE/2012/08/3586.pdf, 2012.

[39] S. He, Y. Huang, L. Yang, and B. Ottersten, "Coordinated multicell multiuser precoding for maximizing weighted sum energy efficiency," *IEEE Trans. Signal Process.*, vol. 62, pp. 741–751, March 2013.

[40] Y. Pan, C. Pan, Z. Yang, M. Chen, and J. Wang, "A caching strategy towards maximal D2D assisted offloading gain," *IEEE Trans. Mobile Comput.*, pp. 1–1, 2019.

[41] Q. Shi, M. Razaviyayn, Z.-Q. Luo, and C. He, "An iteratively weighted MMSE approach to distributed sum-utility maximization for a MIMO interfering broadcast channel," *IEEE Trans. Signal Process.*, vol. 59, pp. 4331–4340, Sep. 2011.

[42] L.-L. Yang, *Multicarrier communications*. John Wiley & Sons, 2009.

[43] Y. Sun, P. Babu, and D. P. Palomar, "Majorization-minimization algorithms in signal processing, communications, and machine learning," *IEEE Trans. Signal Process.*, vol. 65, pp. 794–816, Feb. 2016.

[44] J. Song, P. Babu, and D. P. Palomar, "Sequence design to minimize the weighted integrated and peak sidelobe levels," *IEEE Trans. Signal Process.*, vol. 64, pp. 2051–2064, Apri 2015.

[45] A. Goldsmith, *Wireless Communications*. Cambridge University Press, 2005.

[46] M. L. Fisher, "The Lagrangian relaxation method for solving integer programming problems," *Management Science*, vol. 27, pp. 1–18, Jan. 1981.