Joint Access and Resource Allocation in Ultradense mmWave NOMA Networks With Mobile Edge Computing

Nima Nouri, *Student Member, IEEE*, Jamshid Abouei, *Senior Member, IEEE*, Muhammad Jaseemuddin, *Member, IEEE*, and Alagan Anpalagan, *Senior Member, IEEE*

***Abstract*—This article considers a two-tier heterogeneous network consisting of conventional sub-6-GHz macrocells along with millimeter-wave (mmWave) small cells, where mobile devices (MDs) can connect to either macrocell or small cells opportunistically via the nonorthogonal multiple access (NOMA) protocol. We employ the queuing theory in our network model to conduct an assessment on the execution delay, energy consumption and the total cost of offloading tasks in a mobile-edge computation offloading (MECO) system. The main goal is to design an energy-efficient MECO decision algorithm in an ultradense Internet of Thing (UD-IoT) network to analyze the tradeoff between execution delay and energy consumption. The proposed scheme jointly optimizes the communication and computation resource management, subject to the energy and delay constraints. Due to the mixed-integer nonlinear problem (MINLP) for resource allocation and computation offloading, an iterative algorithm along with the successive convex approximation (SCA) is proposed to achieve the optimum local frequency scheduling, power allocation, and computation offloading. The superior performance of the proposed MECO algorithm in our UD-IoT network is verified by the extensive numerical results.**

***Index Terms*—Internet of Things (IoT) networks, millimeter-wave (mmWave), IoT, mobile edge computing, nonorthogonal multiple access (NOMA) technique, successive convex approximation (SCA), ultradense (UD).**

I. Introduction

RECENTLY, various Internet of Things (IoT) applications in smart cities, such as healthcare services and intelligent transportation systems have employed narrowband IoT [1] and the low-power IPv6 communication module in the wireless personal area networks [2]. In addition, the deployment of ultradense (UD) small cells along with a large number of IoT smart devices has led to an expansion of the conventional IoT toward UD Internet of Thing (UD-IoT) networks [3]–[5]. However, from the quality-of-service (QoS) points of view, the UD-IoT poses a number of new requirements on the existing wireless networks [6]. Although, narrowband IoT applications of smart bike sharing contain elastic necessities on the reliability of the networks, the IoT-based smart healthcare and transportation systems have stringent requirements on the bit error rate, latency, and throughput. Thus, due to the diverse requirements of quality-of-experience (QoE) and QoS, upcoming UD-IoT networks will face some challenges such as how to provide resources to a computation-hungry popular application on the resource-limited IoT mobile devices (MDs) [7], [8]. As an example, many processing tasks performed in IoT MDs, such as face recognition, interactive gaming, and speech processing, impose the computational costs and high power consumptions [9]. However, because of the physical size limitations, lightweight IoT MDs always confront the issue of restricted battery life and the computational resources. One feasible solution to tackle this problem is the mobile-edge computation offloading (MECO) [10]–[14]. Such a scheme significantly decreases the processing time of big data and the energy consumption of smart MDs through offloading massive computational tasks in MDs to the various edge servers located at radio access networks (RANs) in small cells (e.g., relays and femtocells), Wi-Fi access points (APs), and the macrocell. MECO consequently leads to the enhancement of both user-QoE and QoS in UD-IoT networks [15], [16].

Nowadays two methods are extremely attractive in UD-IoT wireless networks: 1) network densification employing small cells [17] and 2) millimeter-wave (mmWave) communications [18]. The cellular network densification consists of dense deployments of small cells coexisting with the macrocell architecture. Traditionally, small cells have been employed in sub-6-GHz frequencies by aiming to offload the burden in the edge server deployed in the macro base station (MBS). To further increase the data rates and due to having very high bandwidth, mmWave small cells have gained high popularity. Apart from achieving high data rates in the large bandwidths, other special features of mmWave communications include noise-limited due to directionality and large bandwidth, and high directional beamforming gains leading to greatly reducing the co-channel interference [19]. Highlighted by the aforementioned benefits, mmWave communication has been

Manuscript received August 10, 2019; revised November 2, 2019; accepted November 17, 2019. Date of publication November 27, 2019; date of current version February 11, 2020. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada. *(Corresponding author: Jamshid Abouei.)*

N. Nouri is with the Department of Electrical Engineering, Yazd University, Yazd 89195-741, Iran (e-mail: nimanouri68@gmail.com).

J. Abouei was with the Department of Electrical, Computer and Biomedical Engineering, Ryerson University, Toronto, ON M5B 2K3, Canada. He is now with the Department of Electrical Engineering, Yazd University, Yazd 89195-741, Iran (e-mail: abouei@yazd.ac.ir).

M. Jaseemuddin and A. Anpalagan are with the Department of Electrical, Computer and Biomedical Engineering, Ryerson University, Toronto, ON M5B 2K3, Canada (e-mail: jaseem@ee.ryerson.ca; alagan@ee.ryerson.ca).

Digital Object Identifier 10.1109/JIOT.2019.2956409

nominated for short-range communications. Nevertheless, very poor penetration into buildings or around solid objects is one of the disadvantages of this type of communication.

Most of the current researches have concentrated on the single-tier base station (BS) framework with very simple MECO schemes [20], [21]. Particularly, they demonstrate that a part of computation tasks may be executed locally at the user side, and other parts should be offloaded to the rich-source edge servers located at the MBS to accomplish the intensive tasks on the behalf of IoT MDs. Besides this, however, the MBS is being congested when the number of users' requests in UD wireless networks increases, and this severely affects the user-QoE and QoS. In order to lessen the burden of the MBS, the lightweight edge servers close to IoT MDS in small cells, or new emerging unmanned aerial vehicles (UAVs), can be utilized to process the computationally intensive tasks. In recent years, the attention of researchers has been devoted to the MECO topic in the UD-5G networks that deploy IoT MDs [22]–[25]. However, the proposed schemes mainly take some simple computation task scenarios into consideration and ignore the cases in which the type of tasks are randomly requested by the IoT devices. Motivated by the above limitations of the current solutions, this article addresses the edge computation offloading issue in a UD-5G heterogeneous network and formulate the joint execution delay and energy consumption optimization problem for the IoT MDs in the network.

II. Contributions

The key contributions of this article are summarized as follows.

1) In this article, we study a mobile edge computation framework with sub-6-GHz macrocells and low-power mmWave small cells in an IoT-based UD-5G heterogeneous network with multiple MDs. Each MD is assumed to connect to either MBS or small cells on the uplink direction, independently. For further assessment on the delay and energy consumption performances, the following queueing models are utilized in the network: a) $M/M/1$ queue for MDs; b) $M/M/1$ queue for APs (or equivalently, fog nodes); and c) the queue with a predetermined maximum request rate and the one at the cloud server is defined as $M/M/c$ queue. There exist a few research works related to the MEC that study such resource allocation and user association in a multiaccess MECO environment [22], [26]–[28].
2) With a focus on the execution delay and the energy consumption of IoT MDs, we provide an integrated framework in the MEC networks for both resource allocation and computation offloading. We formulate a constrained optimization MECO problem in our UD-IoT network to decrease the computation overhead in the whole network and satisfy both the delay and energy constraints.
3) A multiobjective optimization problem involving the minimization of the energy consumption and the execution delay, from the MDs perspective, is formulated where we derive the optimal transmit power, local computing frequency, and the best location for processing the tasks. We use the scalarization method to transform the above multiobjective optimization problem into a single-objective optimization case. The remaining energy of MDs is used to weigh these functions. To address this optimization problem, the successive convex approximation (SCA) method combined with an iterative search algorithm is proposed to achieve the optimal offloading decision, power allocation, local computing frequency scheduling, and computation offloading.
4) Eventually, a comprehensive analysis supported by some simulation studies shows that the proposed MECO algorithm displays substantial superior performance with comparison to other benchmark schemes.

The remainder of this article is structured as follows. Section III presents the related works on the MECO problem. In Section IV, the system model of multidevice MECO is introduced and our optimization problem related to the minimum total cost function is formulated as a mixed-integer nonlinear programming problem. Section V provides an efficient algorithm to solve the joint computation offloading and resource allocation optimization problem. Section VI presents some simulation results for different schemes to evaluate the performance of the network and confirm our analysis. Finally, in Section VII, a summary of the results and conclusions are presented.

III. Related Work

Many researches have investigated the MECO problem in recent years due to the increased popularity of this subject where the main focus of some research has been on the proposing of computation offloading algorithms in the single/multiuser single edge server scenarios [29]–[32]. In these works, the MECO problem has been solved in addition to the interference management, radio, and computational resources allocation [21], [33]. Zheng *et al.* [20] formulated the problem of multiuser MECO as a stochastic game while assuming the time-varying channel gains and the MDs' activity. They presented a multiagent stochastic learning scheme to minimize the system-wide computation overhead. Wang *et al*. [34] studied the multiuser MECO as well as the interference mitigation in cellular systems with MEC and formulate three optimization problems for the physical resource block allocation, the computation offloading decision making, and the MEC computation resource allocation.

In order to reduce the energy consumption of MDs further, some works have proposed some computation offloading algorithms through energy harvesting (EH) techniques or the integration of dynamic voltage frequency scaling in the uplink transmission of IoT-based wireless networks [35]–[37]. Mao *et al.* [36] considered an MEC system with EH devices and proposed an online Lyapunov optimization-based dynamic computation offloading algorithm. In this article, the CPU-cycle frequencies for the tasks processing in MDs, and the transmit power of MDs for the computation offloading decision are determined through the adaptation of their algorithm.

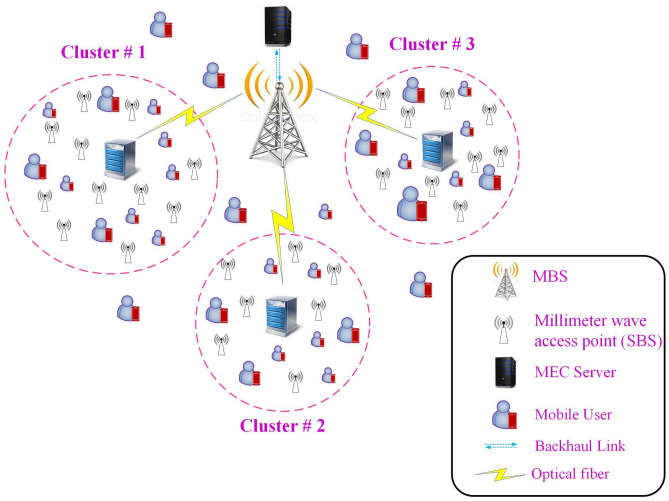

Fig. 1. NOMA-based computation offloading in ultradense IoT networks.

Some works attempted to combine the UD network (UDN) and the MEC to design the MECO problem in 5G multicell scenarios. Sardellitti *et al.* [37] studied the problem of multiuser MECO in a multicell system with the multiple antennas connected to the cloud server, and proposed an iterative algorithm based on the SCA method as the solution. Through optimizing the computation resource allocation and the communication jointly, Zhang *et al.*, presented an energy-efficient computation offloading algorithm to investigate the tradeoff between the MDs' energy consumption and the transmission delay of single/multicell networks. However, the focus of that article has been on the assumption that multiple cells are connected to a common cloud/edge server. This motivates researchers to study the MECO problem in UDNs where multiple edge servers are utilized [22]–[24]. Chen and Hao [22] studied the task offloading problem in a UDN through introducing the software-defined networks where the main objective is to jointly minimize the delay and the MDs' energy consumption in a mixed-integer nonlinear problem (MINLP).

Considering the limitations of the existing works, the main objective of this article is to investigate the computation offloading problem subject to the specified energy and delay constraints for UD-IoT networks with conventional sub-6-GHz macrocells coexisting with ultradense low-power mmWave small cells. It is assumed that each MD connects opportunistically to either macrocell or small cells via the nonorthogonal multiple access (NOMA) protocol. Unlike previous studies in [30], [31], and [33]–[37], in this article, the computation offloading techniques for MEC with mmWave communications have been merged, and the radio and computational resources are jointly assigned to IoT MDs which is modeled as an MINLP problem.

IV. Ultradense IoT Network Model

In this article, we consider a multitier heterogeneous network consisting of a macrocell working in the microwave band where the MEC server is connected to the MBS via optical fiber links, and small cells working in the mmWave band. We assume in this model that small cells are formed by mmWave APs that are directly connected to the local computing servers, also known as fog nodes. The fog nodes provide the computing capacity to offload some tasks from MDs. A collaborative MECO scenario in the UD-IoT networks is presented where collocated IoT MDs are overlaid by small cells or MBS (see Fig. 1). For such an ultradense wireless network, the total number of small cells is assumed to be more than MDs [22]–[25]. The small cells are distributed in the network in a way that they are clustered and more concentrated in dense places with a large number of users. Without loss of generality and for ease of our notations and analysis, we assume that the network is partitioned in N_c clusters where each cluster includes N users distributed in M small cells. This assumption is generic and has been applied in the pertinent MEC networks in the literature (see [34], [37], [38]). We examined our proposed algorithms numerically through deploying a more practical network model with different number of small cells and users within each cluster, and our simulation results showed that this randomness scheme has no effect on the behavior of our results in the general case. We assume that small cells in the network have a limited computational and storage capacity so that the corresponding users can use resources of the network to accomplish their processing tasks. It is supposed that all processors of small cells in each cluster are concentrated in the center of the cluster called the central processor unit. Thus, all small cells in each cluster are connected to the central processor, and this central node is connected to the macrocell through the optical fiber. Furthermore, it is assumed that N_ϕ users in the network are not in the coverage area of small cells and they are served by the MBS. For such a real network model, it is assumed that all users and IoT devices are equipped with omnidirectional antennas, and the method of user access to the radio spectrum is NOMA. In order to analyze the energy consumption and delay performances, we consider the following queue model for the network users' requests: 1) $M/M/1$ queue for MDs and APs (or equivalently, fog nodes) and 2) the queue with a defined maximum request rate and the one at the central cloud is considered as $M/M/c$ queue. We use index i_n related to the ith user in its corresponding cluster $n \in \{1, \ldots, N_c\}$ in the network. In addition, index m_n corresponds to the mth small cell of the network inside the corresponding cluster n. Furthermore, we set $n = 0$ to represent the cluster number of users that are not in the coverage area of the cluster. Hence, the whole users, index set and the processing methods of their computational tasks in the network are represented by $\mathcal{I} \triangleq \{i_n : i = 1, \ldots, N, n = 0, 1, \ldots, N_c\}$ and $\mathcal{M} \triangleq \{m_n : m = 1, \ldots, M, n = 1, \ldots, N_c\} \cup \{m_n = -1, 0\}$, respectively. Index $m_n = -1$ refers to the state that user i_n fulfills its processing task in the local mode, while $m_n = 0$ implies that user i_n intends to receive service through the MEC server in the macrocell, otherwise, the user aims to receive service from the small cell m_n. In addition, we define $\mathcal{M}' \triangleq \mathcal{M} \backslash \{-1\}$ meaning that $\mathcal{M}'$ consists of all members of $\mathcal{M}$ except -1. The task that each user i_n wants to accomplish

is expressed by $\mathcal{APP}_{i_n} = \{V_{i_n}, \theta_{i_n}, \mathcal{T}_{i_n}^{\max}\}$, in which V_{i_n} is the number of CPU cycles required for executing a computational task, θ_{i_n} indicates the size of the input data (including the program code and the input parameters), and $\mathcal{T}_{i_n}^{\max}$ is an upper bound for the execution time of each program. For each mobile device, parameter V_{i_n} is obtained using the synthetic benchmark method in [39], in which a power meter is connected to the MD's battery and makes to measure the V_{i_n} while running. Through running the synthetic benchmark multiple times and varying the runtime of the benchmark, the power meter records the number of CPU cycles that is required for each run. We assume that the computational tasks of users are atomic and indivisible to the smaller parts, where each user accomplishes its processing by the following three methods: 1) each user performs its processing task in a local mode; 2) each user offloads its processing to the fog server which is connected to the small cells with a high power of processing; and 3) each user offloads its processing to the MEC server which is connected to the macrocell. For users who are covered by small cells and intend to offload their processing tasks to the server, due to the short distance between the users and the corresponding SBSs and benefit from the high channel data rates, small cells are in higher priority than the macrocell. Since the main features of mmWave communications are the short range communications along with supporting high data rates requests, small cells should use the mmWave frequency band, while MBS should be assigned a microwave frequency band. In addition, we consider two interference concepts in our network model: 1) intercluster interference, i.e., the interference imposed on one user in cluster $n \in \{1, \ldots, N_c\}$ due to the transmission of one user in cluster $m \in \{1, \ldots, N_c\}$, $m \neq n$, when both MDs are scheduled on the same spectral resources and 2) intracluster interference, i.e., the interference caused by users within the same cluster.

In the following, we will separately compute the energy and the latency of processing for the aforementioned strategies. In addition, the problem is modeled mathematically using the key notations summarized in Table I.

TABLE I
NETWORK'S PARAMETERS

Parameter	Description
k_s	Effective switch capacitance for each MD
θ_{i_n}	Size of input data for user i_n
V_{i_n}	Number of CPU cycles required by $\mathcal{APP}_{i_n}$
$\mathcal{T}_{i_n}^{\max}$	Maximum tolerable delay for user i_n
(δ, α)	Termination accuracy and step size constants
$E_{i_n}^{\max}, P_{i_n}^{\max}$	Energy and power budget constraints for user i_n
ψ'_{i_n}	Weight of the latency function for each user i_n
N_c, N	Number of clusters and users in each cluster
c	Number of servers in the MEC
λ^{MEC}	Average request arrival rate in the MEC server
λ^{FoG,m_n}	Average request arrival rate in m_n^{th} fog server
F^{MEC}, F^{FoG,m_n}	Service rate in the MEC and fog servers
F_b^{MEC}	Sending rate of the MEC
W^{MEC}, W^{FoG}	Sub-6 GHz mmWave bandwidths of the MEC and fog servers

A. Computing Total Cost in Local Processing

We denote $f_{i_n}^{\text{loc}}$ as the computational capacity of a specific MD in the unit of CPU cycles per second. Then, the required time for local processing $\mathcal{APP}_{i_n}$ can be obtained as a function of $f_{i_n}^{\text{loc}}$ as follows:

$$\tau_{i_n}^{\text{loc}}\left(f_{i_n}^{\text{loc}}\right) = \frac{V_{i_n}}{f_{i_n}^{\text{loc}}}. \tag{1}$$

In addition, the corresponding energy consumption for local processing is given by

$$E_{i_n}^{\text{loc}}\left(f_{i_n}^{\text{loc}}\right) = k_s \times \left(f_{i_n}^{\text{loc}}\right)^2 V_{i_n} \tag{2}$$

where $k_s = 10^{-26}$ denotes the effective switch capacitance for each MD which depends on the each device chip architecture [40]. It is seen that parameter $f_{i_n}^{\text{loc}}$ affects the execution time and the energy consumption for the local execution. In this regard, two main objectives of this section are the minimization of the latency and the energy consumption in the MDs' service reception. These objective cost functions pose conflicting goals that requires making tradeoff between them.

Toward this goal, we define the total overhead for each MD in the local case as the weighted sum of the aforementioned cost functions. We utilize the percentage of the residual energy for each MD to obtain the weight of each function. More precisely, users that have lower percent of residual energy are assigned higher energy weight than the latency weight. In contrast, if a delay-sensitive user does not have any energy constraint (e.g., face recognition or fingerprint recognition applications), higher weight is deployed for the latency. Thus, we define parameter ψ_{i_n} for each user i_n as follows:

$$\psi_{i_n} \triangleq \frac{E_{i_n}^{\max}}{E_{i_n}^{\text{total}}} \tag{3}$$

where $\psi_{i_n} \in [0,\ 1]$ indicates the percentage of the residual energy for each user, $E_{i_n}^{\max}$ represents the maximum residual energy with respect to the battery of user i_n, and $E_{i_n}^{\text{total}}$ denotes the user's battery capacity in terms of Joule. Therefore, we can express the computation overhead (i.e., total cost) in the local case as follows:

$$\mathcal{O}_{i_n}^{\text{loc}}\left(f_{i_n}^{\text{loc}}\right) = \psi_{i_n}\tau_{i_n}^{\text{loc}}\left(f_{i_n}^{\text{loc}}\right) + \left(1 - \psi_{i_n}\right)E_{i_n}^{\text{loc}}\left(f_{i_n}^{\text{loc}}\right). \tag{4}$$

B. Computing Total Cost in Remote Processing

We assume that user i_n intends to receive the service from a remote server (e.g., MEC or fog server in a cluster) to develop the hybrid model of the network. In this case, the user is able to connect to the macrocell (i.e., $\mathcal{MC}$) or one of the small cells inside cluster opportunistically [41] and offload its computational task to either a fog node or a MEC server to receive the service that the user needs. In the subsequent section, we use notation $\mathcal{SC}$ to represent the set of small cells. The tolerable delay block for receiving the desired service from the

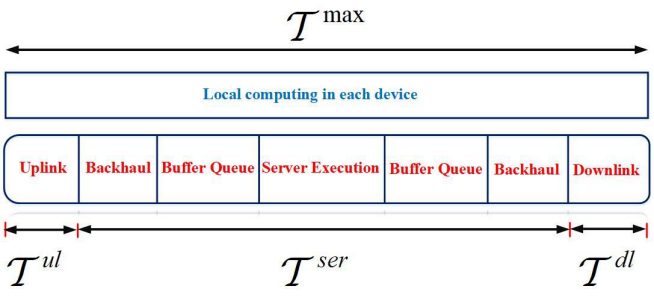

Fig. 2. Time line of offloading from an MD i_n.

remote server, denoted by $\mathcal{T}^{\max}$, is divided into three parts: 1) the time for offloading the users tasks in the uplink direction, denoted by $\mathcal{T}^{ul}$; 2) the execution time of the offloaded tasks by the servers, denoted by $\mathcal{T}^{\text{ser}}$; and 3) the time spent for sending the computation results back to the users, denoted by $\mathcal{T}^{dl}$. In fact, in the first phase with duration $\mathcal{T}^{ul}$, all users offload their computation tasks simultaneously. After collecting these bits, in the second phase with duration $\mathcal{T}^{\text{ser}}$, the desired server remotely executes the offloaded tasks on behalf of these users. Finally, in the third phase with duration $\mathcal{T}^{dl}$, the server sends the computation results back to the users. It should be noted that due to the small sizes of computation results, the time of downloading these results from the server is negligible compared to the time of mobile offloading and local computing (i.e., $\mathcal{T}^{dl} \approx 0$). Accordingly, we can define

$$\mathcal{T}^{\max} \approx \mathcal{T}^{ul} + \mathcal{T}^{\text{ser}}. \tag{5}$$

Note that the time spend in the server (i.e., $\tau_{i_n}^{\text{ser}}$) consists of three parts: 1) the latency due to the round-trip time for exchanging information between the base transceiver station (BTS) and the server through the backhaul link, represented by $\tau_{i_n}^{bh}$; 2) the queueing latency in a buffer, denoted by $\tau_{i_n}^{\text{que}}$; and 3) the latency due to the processing time in the server, represented by $\tau_{i_n}^{\text{exe}}$. In other words

$$\tau_{i_n}^{\text{ser}} \triangleq \tau_{i_n}^{bh} + \tau_{i_n}^{\text{que}} + \tau_{i_n}^{\text{exe}} \tag{6}$$

$$\tau_{i_n}^{\text{ser}} \leq \mathcal{T}^{\text{ser}}. \tag{7}$$

Fig. 2 shows the timeline of the latency experienced by each user in the network, while receiving the desirable service from the server. Our second goal is to minimize the total energy consumption for each user. Thus, we calculate only the energy that a user consumes for execution, transmitting, and receiving data. Below we will derive performance metrics as given in the above five equations in our network model.

1) Uplink Transmission: One of the intrinsic features of mmWave is using multiantenna techniques at the receiver side. Since, in the uplink transmission mode, we assume that each MD is exposed to the beam of only one specific antenna at the corresponding SBS, we use the concept of single-input–single-output (SISO) for all the subsequent SNR and achievable rate expressions in this article. We assume that all users in the network access the spectral resources through the NOMA protocol to superpose the multiple users in the power domain forming a superposition coding for the signal transmission. The transmitted signal for user i_n is denoted by χ_{i_n} such that $\mathbb{E}[|\chi_{i_n}|^2] = 1$, where $\mathbb{E}$ is the expectation operator. Furthermore, using successive interference cancelation (SIC) technique at the BS m_n (either MBS or APs in small cells), the superposition coded signal can be correctly decoded and demodulated at the receiver, which leads to reducing the interference [42]–[45]. Note that the interference in the signal of each user at the receiver side is only due to the users which have stronger channel gains related to the end user. This means that the interference caused by the weaker channels are canceled by this technique [46], [47]. We denote the sample-space received signal at the BS m_n as

$$r^{m_n} = \sum_{i_n \in \mathcal{I}} \sqrt{p_{i_n}} g_{i_n}^{m_n} \chi_{i_n} \mathbb{I}\left\{S_{i_n}^{m_n} = 1\right\} + n_k^{MC}\mathbb{I}\{n = 0\} + n_k^{SC}\mathbb{I}\{n \neq 0\} \quad \forall\, m_n \in \mathcal{M}' \tag{8}$$

where p_{i_n} denotes the transmission power allocated to user i_n which satisfies the power constraint $0 \leq p_{i_n} \leq P_{i_n}^{\max}$. Note that $P_{i_n}^{\max}$ indicates the maximum power budget in each MD. In addition, $g_{i_n}^{m_n}$ represents the channel coefficient between the user i_n and the server m_n. The first term in the right-hand side of (8) is the aforementioned received superposed signal at server m_n, and n_k^{MC} and n_k^{SC} represent the additive white Gaussian noise (AWGN) in macrowave and mmWave bands, respectively. For the indicator function $\mathbb{I}\{S_{i_n}^{m_n} = 1\}$, if user i_n offloads its processing to BS m_n, then $S_{i_n}^{m_n}$ is equal to 1, otherwise, it takes zero value.

We define $\varphi^{m_n} = \{i_n \in \mathcal{I} | S_{i_n}^{m_n} = 1\}\ \forall\ m_n \in \mathcal{M}$, for the set of users that receive their services from BS m_n. In this case, the signal-to-interference-plus-noise ratio (SINR) in the small cell m_n related to the user i_n is defined as

$$\text{SINR}_{i_n}^{m_n} = \frac{p_{i_n} H_{i_n}^{m_n}}{1 + \sum_{j_n \in \varphi^{m_n}} p_{j_n} H_{j_n}^{m_n} \mathbb{I}\left\{H_{j_n}^{m_n} \geq H_{i_n}^{m_n}\right\}}$$

where $h_{i_n}^{m_n} \triangleq |g_{i_n}^{m_n}|^2$ is the direct channel gain between the user i_n and the server m_n and $H_{i_n}^{m_n} \triangleq [(h_{i_n}^{m_n})/(\sigma_{\mathcal{K}}^2)]$, in which $\sigma_{\mathcal{K}}^2 \triangleq \mathbb{E}[|n_{\mathcal{K}}|^2]$, $\mathcal{K} \in \{\mathcal{MC}, \mathcal{SC}\}$, indicates the noise power in macrowave and mmWave bands. In addition

$$h_{i_n}^{m_n} = \tilde{h}_{i_n}^{m_n} \beta^{\mathcal{K}} G_{i_n}^{\mathcal{K}} L^{\mathcal{K}}\left(d_{i_n}^{m_n}\right)^{-1} \Big/ \sigma_{\mathcal{K}}^2 \tag{9}$$

where $L^{\mathcal{K}}(d_{i_n}^{m_n}) = (d_{i_n}^{m_n})^{\alpha_{\mathcal{K}}}$ and $\mathcal{K} \in \{\mathcal{MC}, \mathcal{SC}\}$ denotes the path loss at distance $d_{i_n}^{m_n}$ between user i_n and BS m_n. In addition, $\alpha_{\mathcal{K}}$ is the path-loss exponent where $\alpha_{\mathcal{SC}}$ is equal to $\alpha_{\mathcal{SC}}^{l}$ if the link is line-of-sight (LOS) and $\alpha_{\mathcal{SC}}^{n}$, otherwise, where both parameters $\alpha_{\mathcal{SC}}^{l}$ and $\alpha_{\mathcal{SC}}^{n}$ are different in values from the path-loss exponent for macrocell. Another difference between the channel models for mmWave small cells and macrocell is related to their near-field path-loss models at 1 m, denoted by $\beta^{\mathcal{K}}$, $\mathcal{K} \in \{\mathcal{MC}, \mathcal{SC}\}$. In fact, parameter $\beta^{\mathcal{K}}$ for both macrowave and mmWave have been considered as $\beta^{\mathcal{K}} = (f^{\mathcal{K}}/4\pi)^2$ where $f^{\mathcal{MC}} = 2$ GHz and $f^{\mathcal{SC}} = 70$ GHz. This specific channel model with path loss is used in many research works (see [41], [48], and [49]). In addition, $\tilde{h}_{i_n}^{m_n}$ and $G_{i_n}^{\mathcal{K}}$ in (9) represent the small scale Rayleigh fading channel and the antenna gain, respectively. Note that all mmWave small

cells and macrocell are equipped with directional and omnidirectional antennas, respectively. The main difference between the directional and omnidirectional antennas is related to their antenna gains. Unlike the omnidirectional antenna that has a fixed gain, directional antenna gain is a function of angle θ_{i_n} and the main lobe beam-width θ^{m_n}. Accordingly, small cell receives a signal with $G^{m_n,\max}$, if angle θ_{i_n} of user i_n with respect to the best beam alignment is within the main beam width (θ^{m_n}) of the serving cell, and $G^{m_n,\min}$ otherwise. Thus

$$G_{i_n}^{m_n}(\theta_{i_n}) = \begin{cases} G^{m_n,\max}, & \text{if } |\theta_{i_n}| \leq \frac{\theta^{m_n}}{2} \\ G^{m_n,\min}, & \text{otherwise} \end{cases} \quad \forall m_n \in \mathcal{M}\backslash\{-1,0\}$$

where m_n denotes the small cell in cluster n to which the user i_n is connected. Taking the above considerations into account, if user i_n decides to send its processing to the server indexed by $m_n \in \mathcal{M}'$, the transmission data rate of user i_n in the uplink case is expressed in terms of the number of bits per second as follows:

$$R_{i_n}^{m_n}(p_{i_n}) = W^{\mathcal{K}} \log_2\left(1 + \text{SINR}_{i_n}^{m_n}\right) \tag{10}$$

where $W^{\mathcal{K}}$, $\mathcal{K} \in \{\mathcal{MC}, \mathcal{SC}\}$, denotes the bandwidth of the link between user i_n and its corresponding base station. In addition, the latency or equivalently the required time for transmitting θ_{i_n} data bits to a MEC server in the uplink direction can be obtained as

$$\tau_{i_n}^{ul}(p_{i_n}) = \frac{\theta_{i_n}}{R_{i_n}^{m_n}(p_{i_n})}. \tag{11}$$

Finally, the consumed energy for transmitting θ_{i_n} data bits to the BS is derived as

$$E_{i_n}^{ul}(p_{i_n}) = p_{i_n} \tau_{i_n}^{ul}. \tag{12}$$

2) Offloading Process in the MEC Server in Macrocell: Let F^{MEC} denote the service rate of the MEC server in terms of millions of instructions per second (MIPS) that is unchanged during the execution of the computation task. We assume that the data generated by user i_n follows the Poisson distribution with an average generating rate of λ_{i_n}. Thus, offloaded requests in the queue of the MEC server would be a Poisson process, as well. The requests from different MDs in the system are pooled together with a total rate λ_p^{MEC}, where based on the properties of the Poisson process, λ_p^{MEC} is given by

$$\lambda_p^{\text{MEC}} = \sum_{i_n \in \mathcal{I}} \lambda_{i_n} \mathbb{I}\left\{S_{i_n}^0 = 1\right\}. \tag{13}$$

According to the analysis of $M/M/c$ queue at the MEC server and the Erlang's formula [50], we define

$$\rho^{\text{MEC}} = \frac{\lambda_p^{\text{MEC}}}{cF^{\text{MEC}}} \tag{14}$$

to calculate the average waiting time of each request at the MEC which contains the waiting time at the queue ($\tau_{i_n}^{\text{que}}$) and the execution time ($\tau_{i_n}^{\text{exe}}$) as follows [50], [51]:

$$\tau_{i_n}^{\text{MEC,que}} + \tau_{i_n}^{\text{MEC,exe}} = \frac{C(c, \rho^{\text{MEC}})}{cF^{\text{MEC}} - \lambda_p^{\text{MEC}}} + \frac{1}{F^{\text{MEC}}} \tag{15}$$

where

$$C\left(c, \rho^{\text{MEC}}\right) = \frac{\left(\frac{(c\rho^{\text{MEC}})}{c!}\right)\left(\frac{1}{1-\rho^{\text{MEC}}}\right)}{\sum_{k=0}^{c-1} \frac{(c\rho^{\text{MEC}})^k}{k!} + \left(\frac{(c\rho^{\text{MEC}})}{c!}\right)\left(\frac{1}{1-\rho^{\text{MEC}}}\right)}. \tag{16}$$

Assuming F_b^{MEC} is the transmission data rate of the MEC server, we can obtain the expected time $\tau_{i_n}^{\text{MEC},b}$ for the execution results waiting in the fog node before they are completely delivered as follows:

$$\tau_{i_n}^{\text{MEC},b} = \frac{1}{F_b^{\text{MEC}} - \lambda_p^{\text{MEC}}}. \tag{17}$$

In addition, the corresponding delay due to the backhaul link between small cell m_n and the MEC processor is obtained as

$$\tau_{i_n}^{bh} = \frac{\theta_{i_n}}{\vartheta} \tag{18}$$

where ϑ indicates the capacity in terms of bits per second of the optical fiber link.

3) Offloading Process in the FOG Server Omitted in Small Cells: Let F^{FOG,m_n} indicates the service rate of the fog server m_n in MIPS. If user i_n intends to receive service from the fog server, since the queue model for the incoming requests is considered as $M/M/1$, we can define the total rate in the fog server m_n as

$$\lambda^{\text{FOG},m_n} = \sum_{i_n \in \mathcal{I}} \lambda_{i_n} \mathbb{I}\left\{S_{i_n}^{m_n} = 1\right\} \ \forall m_n \in \mathcal{M}\backslash\{-1,0\}. \tag{19}$$

Hence, we have the following relationships in the calculation of $\tau_{i_n}^{\text{FOG,que}}$ and $\tau_{i_n}^{\text{FOG,exe}}$ [52]:

$$\tau_{i_n}^{\text{FOG,que}} + \tau_{i_n}^{\text{FOG,exe}} = \frac{V_{i_n}}{F^{\text{FOG},m_n} - \lambda^{\text{FOG},m_n}}. \tag{20}$$

In addition, due to the use of optical fiber as backhaul links between the fog processor and the access point, we can calculate $\tau_{i_n}^{bh}$ the same as (18).

4) Downlink Transmission: Since the output data has a much smaller size than the input data, i.e., $E_{i_n}^{dl} \ll E_{i_n}^{ul}$ and $\mathcal{T}^{dl} \ll \mathcal{T}^{ul}$, we ignore the time delay and the energy consumption of receiving the outcome computation result back from the servers as following the same assumptions in [53]–[55].

5) Computing Total Cost in Remote Mode: Similar to the local processing case, the total cost function in the remote processing can be expressed as

$$\mathcal{O}_{i_n}^{\mathcal{K}}(p_{i_n}) = \psi_{i_n} \tau_{i_n}^{\mathcal{K}}(p_{i_n}) + (1 - \psi_{i_n}) E_{i_n}^{\mathcal{K}}(p_{i_n}). \tag{21}$$

Remark 1: Recalling that the binary parameter $S_{i_n}^{m_n} \in \{0,1\}$ denotes the offloading decision for each user i_n which means if user i_n offloads its task to the server $m_n \in \mathcal{M}$, $S_{i_n}^{m_n} = 1$, otherwise, $S_{i_n}^{m_n} = 0$, we can obtain the overhead of user i_n as follows:

$$\begin{aligned} \mathcal{O}_{i_n}\left(p_{i_n}, S_{i_n}^{m_n}\right) = {} & S_{i_n}^{m_n} \, \mathcal{O}_{i_n}^{\text{loc}}\left(f_{i_n}^{\text{loc}}\right) \mathbb{I}\{m_n = -1\} \\ & + S_{i_n}^{m_n} \, \mathcal{O}_{i_n}^{\mathcal{K}}(p_{i_n}) \mathbb{I}\{m_n \neq -1\}. \end{aligned} \tag{22}$$

C. Optimization Problem

Now, we formulate the problem of minimizing the total cost function in (22) as the following optimization problem:

$$\boldsymbol{\mathcal{P}1)}\ \min_{\boldsymbol{S},\boldsymbol{F},\boldsymbol{P}} \sum_{i_n\in\mathcal{I}}\sum_{m_n\in\mathcal{M}} \begin{pmatrix} S_{i_n}^{m_n}\ \mathcal{O}_{i_n}^{\mathrm{loc}}\left(f_{i_n}^{\mathrm{loc}}\right)\mathbb{I}\{m_n=-1\} \\ +S_{i_n}^{m_n}\ \mathcal{O}_{i_n}^{\mathcal{K}}\left(p_{i_n}\right)\mathbb{I}\{m_n\neq -1\}\end{pmatrix}$$

$$\text{s.t. } \mathbf{C1}.\ S_{i_n}^{-1}\tau_{i_n}^{\mathrm{loc}}\left(f_{i_n}^{\mathrm{loc}}\right)+\left(1-S_{i_n}^{-1}\right)\tau_{i_n}^{ul}\leq \mathcal{T}^{\max}-\left(1-S_{i_n}^{-1}\right)\mathcal{T}^{\mathrm{ser}}$$

$$\mathbf{C2}.\ \left(1-S_{i_n}^{-1}\right)\tau_{i_n}^{\mathrm{ser}}\leq \mathcal{T}^{\mathrm{ser}}\ \forall i_n\in\mathcal{I}$$

$$\mathbf{C3}.\ S_{i_n}^{-1}E_{i_n}^{\mathrm{loc}}\left(f_{i_n}^{\mathrm{loc}}\right)+\left(1-S_{i_n}^{-1}\right)E_{i_n}^{\mathcal{K}}\left(p_{i_n}\right)\leq E_{i_n}^{\max}\ \forall i_n\in\mathcal{I}$$

$$\mathbf{C4}.\ 0\leq\left(1-S_{i_n}^{-1}\right)p_{i_n}\leq P_{i_n}^{\max}\ \forall i_n\in\mathcal{I}$$

$$\mathbf{C5}.\ f_{i_n}^{\min}\leq S_{i_n}^{-1}f_{i_n}^{\mathrm{loc}}\leq f_{i_n}^{\max}\ \forall i_n\in\mathcal{I}$$

$$\mathbf{C6}.\ \lambda_p^{\mathrm{MEC}}<cF^{\mathrm{MEC}}$$

$$\mathbf{C7}.\ \lambda_p^{\mathrm{MEC}}<F_b^{\mathrm{MEC}}$$

$$\mathbf{C8}.\ \lambda^{\mathrm{FOG},m_n}<F^{\mathrm{FOG},m_n}\ \forall m_n\in\mathcal{M}\backslash\{-1,0\}$$

$$\mathbf{C9}.\ S_{i_n}^{m_n}\in\{0,1\}\ \forall i_n\in\mathcal{I},m_n\in\mathcal{M}$$

$$\mathbf{C10}.\ \sum_{m_n\in\mathcal{M}}S_{i_n}^{m_n}=1\ \forall i_n\in\mathcal{I}$$

where $S_{i_n}^{-1}=1$ is related to the case that MD i_n executes $\mathcal{APP}_{i_n}$ locally in its own CPU, while $S_{i_n}^{-1}=0$ means that MD i_n decides to offload its task to other servers (e.g., MEC or fog server in a cluster). Constraints **C1**−**C4** represent the latency, energy, and power constraints which should be less than or equal to the maximum tolerable delay ($\mathcal{T}^{\max}$), residual energy ($E_{i_n}^{\max}$) and power budget ($P_{i_n}^{\max}$) in each MD. In addition, constraint **C5** shows the local CPU-cycle frequency restriction, **C6**−**C8** are obtained from (15), (17), and (20), respectively, and **C9** indicates that each MD executes its task by either local or remote computing. Finally, **C10** represents the constraint on the offloading decision parameter of each MD.

For convenience in mathematical expressions, we collect all optimization variables in vector $\boldsymbol{\mathcal{U}}\triangleq(\boldsymbol{S},\boldsymbol{F},\boldsymbol{P})$, where $\boldsymbol{S}\triangleq(S_{i_n}^{m_n})_{i_n\in\mathcal{I}}$, $\boldsymbol{F}\triangleq(f_{i_n}^{\mathrm{loc}})_{i_n\in\mathcal{I}}$, and $\boldsymbol{P}\triangleq(p_{i_n})_{i_n\in\mathcal{I}}$. Obviously, if $[(V_{i_n})/(f_{i_n}^{\mathrm{local}})]\leq\mathcal{T}^{\max}$ and $k_s\times(f_{i_n}^{\mathrm{local}})^2V_{i_n}\leq E_{i_n}^{\max}\ \forall i_n\in\mathcal{I}$, the feasible set of problem $\boldsymbol{\mathcal{P}1}$ is nonempty and be guaranteed that there is at least one solution for this problem. Considering nonconvexity of the objective functions and constraints, problem $\boldsymbol{\mathcal{P}1}$ is nonconvex. Since $S_{i_n}^{m_n}$ is a binary parameter and $\boldsymbol{\mathcal{P}1}$ is considered as an MINLP, finding an optimum solution for the problem is intractable. In addition, problem $\boldsymbol{\mathcal{P}1}$ is a generalization of the knapsack problem [56] and for this reason, it is considered as an NP-hard problem. Thus, we cannot find an optimum solution in the polynomial time.

V. Proposed Algorithm To Solve Problem $\boldsymbol{\mathcal{P}1}$

In order to tackle the optimization problem $\boldsymbol{\mathcal{P}1}$, we divide problem $\boldsymbol{\mathcal{P}1}$ in two separate parts: 1) the total cost in local mode ($\boldsymbol{\mathcal{P}2}$) and 2) the total cost in remote mode ($\boldsymbol{\mathcal{P}3}$). Generally, the procedure of the proposed algorithm in solving $\boldsymbol{\mathcal{P}1}$ is divided into six stages performed as follows.

A. Optimum Local Processing

In the first step, all users $i_n\in\mathcal{I}$ are supposed to accomplish their computational tasks in a local mode (i.e., $S_{i_n}^{-1}=1$), then, the optimum value of CPU-cycle frequency should be obtained such that it minimizes the local overhead. With the assumption $S_{i_n}^{-1}=1$, problem $\boldsymbol{\mathcal{P}1}$ is converted to the following optimization problem:

$$\boldsymbol{\mathcal{P}2)}\ \min_{\boldsymbol{F}}\sum_{i_n\in\mathcal{I}}\mathcal{O}_{i_n}^{\mathrm{loc}}\left(f_{i_n}^{\mathrm{loc}}\right)$$

$$\text{s.t. } \mathbf{C1}.\ \tau_{i_n}^{\mathrm{loc}}\left(f_{i_n}^{\mathrm{loc}}\right)\leq\mathcal{T}^{\max}\qquad\forall i_n\in\mathcal{I}$$

$$\mathbf{C2}.\ E_{i_n}^{\mathrm{loc}}\left(f_{i_n}^{\mathrm{loc}}\right)\leq E_{i_n}^{\max}\qquad\forall i_n\in\mathcal{I}$$

$$\mathbf{C3}.\ f_{i_n}^{\min}\leq f_{i_n}^{\mathrm{loc}}\leq f_{i_n}^{\max}\qquad\forall i_n\in\mathcal{I}.$$

Problem $\boldsymbol{\mathcal{P}2}$ is convex that is straightforward to be solved. In order to solve the optimization problem, one can find the first-order derivation of the objective function $f_{i_n}^{\mathrm{loc}}$ and set it equal to zero. Therefore, we have

$$\tilde{f}_{i_n}=\sqrt[3]{\frac{\psi_{i_n}}{2\left(1-\psi_{i_n}\right)k_s}}. \tag{23}$$

On the other hand, according to **C1** and **C2**, one can easily find upper and lower bands for $f_{i_n}^{\mathrm{loc}}$ by the following relations:

$$f^{lb}\leq f_{i_n}^{\mathrm{loc}}\leq f^{ub} \tag{24}$$

where $f^{lb}\triangleq[(V_{i_n})/(\mathcal{T}^{\max})]$ and $f^{ub}\triangleq\sqrt{[(E_{i_n}^{\max})/(k_sV_{i_n})]}$ obtained from (1) and (2). Through combining (24) and **C3**, we can modify the lower and upper bands for $f_{i_n}^{\mathrm{loc}}$ as follows:

$$f_{i_n}^{l}=\max\left\{f_{i_n}^{\min},f^{lb}\right\} \tag{25}$$

$$f_{i_n}^{u}=\min\left\{f_{i_n}^{\max},f^{ub}\right\}. \tag{26}$$

If we have $f_{i_n}^{u}<f_{i_n}^{l}$ for one of the users, then, the feasible region of problem $\boldsymbol{\mathcal{P}2}$ is empty and the problem has no solution. This means that user cannot execute its processing in the local mode and it should use either MEC or the fog server. Clearly, for this type of user, $S_{i_n}^{-1}=0$. However, for the users that $f_{i_n}^{l}\leq f_{i_n}^{u}$, and using the obtained upper and lower bounds for $f_{i_n}^{\mathrm{loc}}$, the optimum value of $f_{i_n}^{\mathrm{loc}}$ is given by

$$\hat{f}_{i_n}^{\mathrm{loc}}=\begin{cases}f_{i_n}^{l} & \tilde{f}_{i_n}\leq f_{i_n}^{l}\\ \tilde{f}_{i_n} & f_{i_n}^{l}\leq\tilde{f}_{i_n}\leq f_{i_n}^{u}\\ f_{i_n}^{u} & \tilde{f}_{i_n}>f_{i_n}^{u}.\end{cases} \tag{27}$$

In addition, the optimum local overhead for user i_n can be obtained as (see lines 1–8 of Algorithm 1)

$$\min_{\boldsymbol{F}}\sum_{i_n\in\mathcal{I}}\mathcal{O}_{i_n}^{\mathrm{loc}}\left(f_{i_n}^{\mathrm{loc}}\right)=\sum_{i_n\in\mathcal{I}}\mathcal{O}_{i_n}^{\mathrm{loc}}\left(\hat{f}_{i_n}^{\mathrm{loc}}\right). \tag{28}$$

B. Optimum Remote Processing

In the second case, we assume that each user $i_n \in \mathcal{I}$ wants to receive one service from a server (fog or MEC), then $S_{i_n}^{-1} = 0$, thus, problem $\boldsymbol{\mathcal{P}1}$ can be changed to the following optimization problem:

$$\begin{aligned}
\boldsymbol{\mathcal{P}3)}\ \min_{\boldsymbol{S},\boldsymbol{P}}\ & \sum_{i_n\in\mathcal{I}}\sum_{m_n\in\mathcal{M}'} S_{i_n}^{m_n}\mathcal{O}_{i_n}^{\mathcal{K}}\big(p_{i_n}\big)\\
\text{s.t.}\ & \mathbf{C1}.\ \tau_{i_n}^{ul} \le \mathcal{T}^{ul} \qquad \forall i_n \in \mathcal{I}\\
& \mathbf{C2}.\ \tau_{i_n}^{\text{ser}} \le \mathcal{T}^{\text{ser}} \qquad \forall i_n \in \mathcal{I}\\
& \mathbf{C3}.\ E_{i_n}^{\mathcal{K}}\big(p_{i_n}\big) \le E_{i_n}^{\max} \qquad \forall i_n \in \mathcal{I}\\
& \mathbf{C4}.\ 0 \le p_{i_n} \le P_{i_n}^{\max} \qquad \forall i_n \in \mathcal{I}\\
& \mathbf{C5}.\ \lambda_p^{\text{MEC}} < cF^{\text{MEC}}\\
& \mathbf{C6}.\ \lambda_p^{\text{MEC}} < F_b^{\text{MEC}}\\
& \mathbf{C7}.\ \lambda^{\text{FOG},m_n} < F^{\text{FOG},m_n} \qquad \forall m_n \in \mathcal{M}\backslash\{-1,0\}\\
& \mathbf{C8}.\ S_{i_n}^{m_n} \in \{0,1\} \qquad \forall i_n \in \mathcal{I}, m_n \in \mathcal{M}\\
& \mathbf{C9}.\ \sum_{m_n\in\mathcal{M}} S_{i_n}^{m_n} = 1 \qquad \forall i_n \in \mathcal{I}.
\end{aligned}$$

This problem is still nonconvex and is considered to be MINLP and NP-hard which its solution is complicated. In order to tackle this problem, we classify the users into two groups. The first group is related to the users that are connected to the macrocell, while the second group includes the users that are connected to small cells inside the cluster. As mentioned in [41], different criteria are introduced to allocate the small cells to the network users, e.g., the biased received powers. Using (9), we perform this classification algorithm based on the direct channel gain criterion as follows [41], [57]:

$$\tilde{h}_{i_n}^{m_n^*}\beta^{\mathcal{K}}G_{i_n}^{\mathcal{K}}L^{\mathcal{K}}\Big(d_{i_n}^{m_n^*}\Big)^{-1} \ge \tilde{h}_{i_n}^{m_n}\beta^{\mathcal{K}}G_{i_n}^{\mathcal{K}}L^{\mathcal{K}}\Big(d_{i_n}^{m_n}\Big)^{-1}. \quad (29)$$

Using this criterion, user i_n connects to the fog server which has the best channel condition. In addition, in (29), the asterisk superscript (i.e., $*$) for server m_n means that the best server (in terms of having the best direct channel gain condition) is selected for user i_n to start communication with this nominated server. Moreover, this selection scheme is completely independent of the interference in the network. Denoting $\mathcal{Q}_{\mathcal{MC}}$ and $\mathcal{Q}_{\mathcal{SC}}$ as the sets of users that receive service from the MEC and small cells, respectively, we can rewrite problem $\boldsymbol{\mathcal{P}3}$ as

$$\begin{aligned}
\boldsymbol{\mathcal{P}4)}\ \min_{\boldsymbol{P}}\ & \mathcal{O}^{\mathcal{K}} = \left[\sum_{i_n\in\mathcal{Q}_{\mathcal{SC}}}\mathcal{O}_{i_n}^{\mathcal{K}}\big(p_{i_n}\big) + \sum_{i_n\in\mathcal{Q}_{\mathcal{MC}}}\mathcal{O}_{i_n}^{\mathcal{K}}\big(p_{i_n}\big)\right]\\
\text{s.t.}\ & \mathbf{C1}.\ \tau_{i_n}^{ul} \le \mathcal{T}^{ul}\ \forall i_n \in \mathcal{I}\\
& \mathbf{C2}.\ \tau_{i_n}^{\text{ser}} \le \mathcal{T}^{\text{ser}}\ \forall i_n \in \mathcal{I}\\
& \mathbf{C3}.\ E_{i_n}^{\mathcal{K}}\big(p_{i_n}\big) \le E_{i_n}^{\max}\ \forall i_n \in \mathcal{I}\\
& \mathbf{C4}.\ 0 \le p_{i_n} \le P_{i_n}^{\max}\ \forall i_n \in \mathcal{I}\\
& \mathbf{C5}.\ \lambda_p^{\text{MEC}} < cF^{\text{MEC}}\\
& \mathbf{C6}.\ \lambda_p^{\text{MEC}} < F_b^{\text{MEC}}\\
& \mathbf{C7}.\ \lambda^{\text{FOG},m_n} < F^{\text{FOG},m_n}\ \forall m_n \in \mathcal{M}\backslash\{-1,0\}.
\end{aligned}$$

It is seen that problem $\boldsymbol{\mathcal{P}4}$ is nonconvex because of the nonconvexity of the objective function and constraints **C1** and **C3**, so we employ the SCA algorithm in [58] and [59] (lines 9–48 of Algorithm 1) to solve this challenging optimization problem. In this algorithm, an iterative procedure is performed to solve such a nonconvex problem where in each iteration, we obtain a convex successor of $\boldsymbol{\mathcal{P}4}$ by calculating the convex approximation for the objective function and the constraints **C1** and **C3** so that these approximations should satisfy the specific constraints in [58] and [59]. Using the obtained total cost from $\boldsymbol{\mathcal{P}4}$ with the previous iteration (i.e., $\mathcal{O}_{i_n}^{\mathcal{K}}(p_{i_n})$) and in comparison with the local cost (i.e., $\mathcal{O}_{i_n}^{\text{loc}}(\hat{f}_{i_n}^{\text{loc}})$), we can classify the users into three categories (i.e., $\mathcal{Q}_{\mathcal{L}}$, $\mathcal{Q}_{\mathcal{MC}}$, and $\mathcal{Q}_{\mathcal{SC}}$), as will be discussed shortly.

C. Convex Approximation for Objective Function

Let us denote the set $\boldsymbol{\mathcal{X}}$ as the feasible region for problem $\boldsymbol{\mathcal{P}4}$ and $\boldsymbol{P}(\upsilon)$ represents the transmit power vector calculated in the υth iteration of the algorithm. If we denote the approximation of the objective function $\mathcal{O}^{\mathcal{K}}(\boldsymbol{P};\boldsymbol{P}(\upsilon))$ around the current feasible point $\boldsymbol{P}(\upsilon)$, i.e., the vector $\boldsymbol{P} \triangleq (p_{i_n})_{i_n\in\mathcal{I}}$, by $\tilde{\mathcal{O}}^{\mathcal{K}}(\boldsymbol{P};\boldsymbol{P}(\upsilon))$, this approximation should satisfy the following conditions [58, Sec. II]:

A1: $\tilde{\mathcal{O}}^{\mathcal{K}}(\bullet;\boldsymbol{P}(\upsilon))$ is uniformly convex on $\boldsymbol{\mathcal{J}}$ with constant $c_{\tilde{\mathcal{O}}^{\mathcal{K}}} > 0$ meaning that $\forall x, z \in \boldsymbol{\mathcal{J}}$ and $\forall y \in \boldsymbol{\mathcal{X}}$, we have $c_{\tilde{\mathcal{O}}^{\mathcal{K}}}\|x-z\|^2 \le (\nabla_x\tilde{\mathcal{O}}^{\mathcal{K}}(x;y) - \nabla_x\tilde{\mathcal{O}}^{\mathcal{K}}(z;y))(x-z)^T$;

A2: $\nabla_{\boldsymbol{P}}\tilde{\mathcal{O}}^{\mathcal{K}}(\boldsymbol{P}(\upsilon);\boldsymbol{P}(\upsilon)) = \nabla_{\boldsymbol{P}}\mathcal{O}^{\mathcal{K}}(\boldsymbol{P}(\upsilon);\boldsymbol{P}(\upsilon))$, for all $\boldsymbol{P}(\upsilon) \in \boldsymbol{\mathcal{X}}$;

A3: $\nabla_{\boldsymbol{P}}\tilde{\mathcal{O}}^{\mathcal{K}}(\bullet;\bullet)$ is continuous on $\boldsymbol{\mathcal{J}} \times \boldsymbol{\mathcal{X}}$.

For the above constraints, $\nabla_x f(x;y)$ denotes the partial gradient of $f(x;\ y)$ with respect to the first argument evaluated at $(x;\ y)$, and $\boldsymbol{\mathcal{J}}$ represents a compact convex set that includes the feasible set $\boldsymbol{\mathcal{X}}$, i.e., $\boldsymbol{\mathcal{X}} \subseteq \boldsymbol{\mathcal{J}}$. The above condition emphasizes that in addition to the convexity of smoothness, the first order behavior of the approximation should be similar to the original nonconvex function. We will compute a convex approximation for the objective function so that it satisfies the aforementioned conditions A1–A3 as follows:

$$\begin{aligned}
\tilde{\mathcal{O}}^{\mathcal{K}}(\boldsymbol{P};\boldsymbol{P}(\upsilon)) = \text{MEC}\sum_{i_n\in\mathcal{I}}\tilde{\mathcal{O}}_{i_n}^{\mathcal{K}}\big(p_{i_n};\boldsymbol{P}(\upsilon)\big)\\
+\ \frac{\gamma_{\boldsymbol{P}}}{2}\|\boldsymbol{P}-\boldsymbol{P}(\upsilon)\|^2. \quad (30)
\end{aligned}$$

The second term in the right-hand side of (30) is a quadratic regularization term that is aggregated in order to make $\tilde{\mathcal{O}}^{\mathcal{K}}(\boldsymbol{P};\ \boldsymbol{P}(\upsilon))$ uniformly strongly convex and $\gamma_{\boldsymbol{P}}$ indicates the positive arbitrary constant. In addition

$$\begin{aligned}
\tilde{\mathcal{O}}_{i_n}^{\mathcal{K}}\big(p_{i_n};\boldsymbol{P}(\upsilon)\big) = \psi_{i_n}\tau_{i_n}^{\mathcal{K}}\big(p_{i_n}\big)\\
+\ \big(1-\psi_{i_n}\big)\tilde{E}_{i_n}^{\mathcal{K}}(\boldsymbol{P};\boldsymbol{P}(\upsilon)) \quad (31)
\end{aligned}$$

where

$$\begin{aligned}
\tilde{E}_{i_n}^{\mathcal{K}}(\boldsymbol{P};\boldsymbol{P}(\upsilon)) = p_{i_n}(\upsilon)\frac{\theta_{i_n}}{R_{i_n}^{m_n}\big(p_{i_n},p_{-i_n}(\upsilon)\big)}\\
+\ p_{i_n}\frac{\theta_{i_n}}{R_{i_n}^{m_n}\big(p_{i_n}(\upsilon),p_{-i_n}(\upsilon)\big)}\\
+\sum_{j_m\in\mathcal{I}}\frac{\partial E_{j_m}(\boldsymbol{P})}{\partial p_{i_n}}\times\big(p_{i_n}-p_{i_n}(\upsilon)\big). \quad (32)
\end{aligned}$$

Algorithm 1 Proposed Algorithm for Joint Access and Resource Management

Initialization:
Classified MDs sets: $\mathcal{Q}_{\mathcal{L}} = \mathcal{Q}_{\mathcal{MC}} = \mathcal{Q}_{\mathcal{SC}} = \phi$.
Set $\Omega_0 = \Omega_1 = \Omega_2 = \phi$ and $\Omega \triangleq \mathcal{I}$.
Set $\mu_{low} = 0$, $\mu_{up} = numel(\Omega)$.
Set $j = 1$, $\gamma(0) \in (0, 1]$ and $\alpha \in \left(0, \frac{1}{\gamma(0)}\right)$.
for each $i_n \in \mathcal{I}$ **do**
  Obtain $\mathcal{APP}_{i_n} = \left\{V_{i_n}, \theta_{i_n}, \mathcal{T}_{i_n}^{\max}\right\}$.
  Calculate ψ_{i_n} according to (3).
  Calculate $f_{i_n}^{l}$ and $f_{i_n}^{u}$ according to (25) and (26).
  Calculate $\tilde{f}_{i_n}$ and $\tilde{f}_{i_n}^{loc}$ according to (23) and (27).
  Determine $\mathcal{O}_{i_n}^{loc}(f_{i_n}^{loc})$ according to (32).
  Perform cell association according to (29).
  Update $\mathcal{Q}_{\mathcal{SC}}$ and $\mathcal{Q}_{\mathcal{MC}}$.
end for
while $\mu_{up} - \mu_{low} > 1$ **do**
  Compute $\tilde{\mathcal{O}}^{\mathcal{K}}(\boldsymbol{P}; \boldsymbol{P}(\upsilon))$ according to (30) and (32).
  Compute $\tilde{\tau}_{i_n}^{\mathcal{K}}(p_{i_n})$ according to (35).
  Compute $\bar{E}_{i_n}^{\mathcal{K}}(p_{i_n})$ according to (38).
  Obtain $\boldsymbol{\mathcal{P}5}$ as convex successor of $\boldsymbol{\mathcal{P}4}$.
  Set $\upsilon = 0$.
  Select $\boldsymbol{P}(0) \in \boldsymbol{\mathcal{X}}$.
  Solve convex problem $\boldsymbol{\mathcal{P}5}$ to obtain $\hat{\boldsymbol{P}}$.
  calculate $\mathcal{O}_{i_n}^{\mathcal{K}}(\hat{p}_{i_n})$ and $\mathcal{O}^{\mathcal{K}}(\hat{\boldsymbol{P}})$ according to (21) and (22), respectively.
  while $\left|\mathcal{O}^{\mathcal{K}}\left(\hat{\boldsymbol{P}}(\upsilon+1)\right) - \mathcal{O}^{\mathcal{K}}\left(\hat{\boldsymbol{P}}(\upsilon)\right)\right| \leq \delta$ **do**
    Set $\boldsymbol{P}(\upsilon+1) = \boldsymbol{P}(\upsilon) + \gamma(\upsilon)(\hat{\boldsymbol{P}}(\upsilon) - \boldsymbol{P}(\upsilon))$.
    $\upsilon \leftarrow \upsilon + 1$, and return to step 16.
    $\gamma(\upsilon) = \gamma(\upsilon-1)\mathbb{I}\{\upsilon > 0\}(1 - \alpha\gamma(\upsilon-1)) + \gamma(0)\mathbb{I}\{\upsilon = 0\}$.
  end while
  $\boldsymbol{P}^{opt} \leftarrow \hat{\boldsymbol{P}}(\upsilon+1)$.
  for each $i_n \in \Omega$ **do**
    if $j == 1$ && $\mathcal{O}_{i_n}^{\mathcal{K}}(p_{i_n}^{opt}) < \mathcal{O}_{i_n}^{loc}(f_{i_n}^{loc})$ **then**
      $\Omega_1 \leftarrow \Omega_1 \cup i_n$.
      Set $S_{i_n}^{-1} = 0$.
    else if $j > 1$&&$\left(\mathcal{O}^{\mathcal{K}}(\boldsymbol{P}^{opt})\right)^{j} < \left(\mathcal{O}^{\mathcal{K}}(\boldsymbol{P}^{opt})\right)^{j-1}$
      $\mu_{low} \leftarrow \lambda$.
    else if $j > 1$&&$\left(\mathcal{O}^{\mathcal{K}}(\boldsymbol{P}^{opt})\right)^{j} > \left(\mathcal{O}^{\mathcal{K}}(\boldsymbol{P}^{opt})\right)^{j-1}$
      $\mu_{up} \leftarrow \lambda$.
    else $j == 1$ && $\mathcal{O}_{i_n}^{\mathcal{K}}(p_{i_n}^{opt}) > \mathcal{O}_{i_n}^{loc}(f_{i_n}^{loc})$
      Set $S_{i_n}^{-1} = 1$.
    end if
  end for
  if $j == 1$ **then**
    $\Omega_2 \triangleq \Omega \backslash \Omega_1$.
  end if
  Sort Ω_1 and Ω_2 according to the total cost of the remote in descending order.
  Set $\Omega_0 \leftarrow \Omega_1 + \Omega_2$.
  Set $j \leftarrow j + 1$.
  Set $\lambda \leftarrow numel(\Omega_1) + \left\lfloor \frac{\mu_{low}+\mu_{up}}{2} \right\rfloor$.
  Set $\Omega \leftarrow \Omega_0\{1, ..., \lambda\}$.
end while
$\lambda^* = \mu_{low}$.
  if $i_n \in \Omega_0\{1, ..., \lambda^*\}$ **do**
  Set $S_{i_n}^{-1} = 0$.
  else if $i_n \in \Omega_0\{\lambda^* + 1, ..., numel(\Omega_0)\}$ **do**
  Set $S_{i_n}^{-1} = 1$.
  end if
Output: $\mathcal{Q}_{\mathcal{L}}, \mathcal{Q}_{\mathcal{MC}}, \mathcal{Q}_{\mathcal{SC}}$ and $\boldsymbol{\mathcal{U}}^{opt} = (\boldsymbol{S}^{opt}, \boldsymbol{F}^{opt}, \boldsymbol{P}^{opt})$.

Note that the first two phrases on the right-hand side of (32) are the desired convexification of $E_{i_n}^{\mathcal{K}}(\boldsymbol{P}; \boldsymbol{P}(\upsilon))$, while the third part denotes the linearization term, furthermore, $p_{-i_n}(\upsilon) = (p_{j_n}(\upsilon))_{i \neq j=1}^{N}$.

D. Convex Approximation for Constraints C1 and C3

If we denote the convex approximation of this function in the υth iteration of the SCA algorithm by $\tilde{\tau}_{i_n}^{ul}(p_{i_n})$, this approximation should satisfy the following conditions [58, Sec. II]:

B1: $\tilde{\tau}_{i_n}^{ul}(\bullet; \boldsymbol{P}(\upsilon))$ is uniformly convex on $\boldsymbol{\mathcal{J}}$;

B2: $\nabla_{\boldsymbol{P}}\tilde{\tau}_{i_n}^{ul}(\boldsymbol{P}(\upsilon); \boldsymbol{P}(\upsilon)) = \nabla_{\boldsymbol{P}}\tau_{i_n}^{ul}(\boldsymbol{P}(\upsilon); \boldsymbol{P}(\upsilon))$, for all $\boldsymbol{P}(\upsilon) \in \boldsymbol{\mathcal{X}}$;

B3: $\nabla_{\boldsymbol{P}}\tilde{\tau}_{i_n}^{ul}(\bullet; \bullet)$ is continuous on $\boldsymbol{\mathcal{J}} \times \boldsymbol{\mathcal{X}}$;

B4: $\tilde{\tau}_{i_n}^{ul}(\boldsymbol{P}; \boldsymbol{P}(\upsilon)) \geq \tau_{i_n}^{ul}(\boldsymbol{P}; \boldsymbol{P}(\upsilon))$, for all $\boldsymbol{P}(\upsilon) \in \boldsymbol{\mathcal{X}}$ and $\boldsymbol{P} \in \boldsymbol{\mathcal{J}}$;

B5: $\tilde{\tau}_{i_n}^{ul}(\boldsymbol{P}(\upsilon); \boldsymbol{P}(\upsilon)) = \tau_{i_n}^{ul}(\boldsymbol{P}(\upsilon); \boldsymbol{P}(\upsilon))$, for all $\boldsymbol{P}(\upsilon) \in \boldsymbol{\mathcal{X}}$;

B6: $\tilde{\tau}_{i_n}^{ul}(\bullet; \bullet)$ is continuous on $\boldsymbol{\mathcal{J}} \times \boldsymbol{\mathcal{X}}$.

In order to calculate this approximation, we first rewrite (10) as the difference between two concave functions as shown in (33), at the bottom of the next page, where functions $R_{i_n}^{m_n{}^+}(p_{i_n})$ and $R_{i_n}^{m_n{}^-}(p_{i_n})$ are concave functions. To compute the above concave approximation of the function, we use the first-order Taylor's expansion approximation of the function $R_{i_n}^{m_n{}^-}(p_{i_n})$, i.e.,

$$\tilde{R}_{i_n}^{m_n}(p_{i_n}) = R_{i_n}^{m_n{}^+}(p_{i_n}) - R_{i_n}^{m_n{}^-}(p_{i_n}(\upsilon)) - \sum_{j_m \in \mathcal{I}} \frac{\partial R_{j_m}^{m_n{}^-}(p_{j_m}(\upsilon))}{\partial p_{j_m}} \times (p_{j_m} - p_{j_m}(\upsilon)). \quad (34)$$

Finally, by substituting the concave approximation in (11), we can claim that the convex upper bound for the constraint **C1** in problem $\boldsymbol{\mathcal{P}4}$: $\tilde{\tau}_{i_n}^{ul}(p_{i_n}) \leq \mathcal{T}^{\max}\ \forall i_n \in \mathcal{I}$ can be obtained by

$$\tilde{\tau}_{i_n}^{ul}(p_{i_n}) = \frac{\theta_{i_n}}{\tilde{R}_{i_n}^{m_n}(p_{i_n})}. \quad (35)$$

In the next step, constraint **C3** is manipulated as follows:

$$p_{i_n}\frac{\theta_{i_n}}{R_{i_n}^{m_n}(p_{i_n})} \leq E_{i_n}^{\max} \quad (36)$$

or equivalently

$$p_{i_n}\theta_{i_n} - E_{i_n}^{\max}R_{i_n}^{m_n}(p_{i_n}) \leq 0. \quad (37)$$

Therefore, the convex approximation for the constraint **C3** would be obtained by substituting (34) in (37) as follows:

$$\bar{E}_{i_n}^{\mathcal{K}}(p_{i_n}) \triangleq p_{i_n}\theta_{i_n} - E_{i_n}^{\max}\tilde{R}_{i_n}^{m_n}(p_{i_n}) \leq 0. \quad (38)$$

Remark 2: It can be easily demonstrated that the obtained convex approximations satisfy the constraints mentioned in A1-A3 and B1-B6.

E. Convex Successor of Problem $\mathcal{P}4$

After computing convex approximations of the objective function [i.e., $\tilde{\mathcal{O}}^{\mathcal{K}}(\boldsymbol{P}; \boldsymbol{P}(\upsilon))$ in (30)] and the constraints **C1** and **C3** around the current iterate $\boldsymbol{P}(\upsilon) \in \boldsymbol{\mathcal{X}}$, the following problem can be solved instead of $\boldsymbol{\mathcal{P}4}$ by conventional methods such as interior point methods

$$\begin{aligned}
\boldsymbol{\mathcal{P}5}) \ \min_{\boldsymbol{P}} \ & \sum_{i_n \in \mathcal{Q}_{\mathcal{SC}}} \tilde{\mathcal{O}}_{i_n}^{\mathcal{K}}(p_{i_n}) + \sum_{i_n \in \mathcal{Q}_{\mathcal{MC}}} \tilde{\mathcal{O}}_{i_n}^{\mathcal{K}}(p_{i_n}) \\
\text{s.t.} \ & \mathbf{C1}.\ \tilde{\tau}_{i_n}^{ul}(p_{i_n}) \leq \mathcal{T}^{\max}\ \forall i_n \in \mathcal{I} \\
& \mathbf{C2}.\ \tau_{i_n}^{\text{ser}} \leq \mathcal{T}^{\text{ser}}\ \forall i_n \in \mathcal{I} \\
& \mathbf{C3}.\ \bar{E}_{i_n}^{\mathcal{K}}(p_{i_n}) \leq 0\ \forall i_n \in \mathcal{I} \\
& \mathbf{C4}.\ 0 \leq p_{i_n} \leq P_{i_n}^{\max}\ \forall i_n \in \mathcal{I} \\
& \mathbf{C5}.\ \lambda_p^{\text{MEC}} < cF^{\text{MEC}}
\end{aligned}$$

$$\textbf{C6}.\ \lambda_p^{\mathrm{MEC}} < F_b^{\mathrm{MEC}}$$
$$\textbf{C7}.\ \lambda^{\mathrm{FOG},m_n} < F^{\mathrm{FOG},m_n}\ \forall m_n \in \mathcal{M}\backslash\{-1,0\}.$$

Note that we use lines 13–22 of Algorithm 1 to achieve the optimal solution. For this part of the algorithm, $\boldsymbol{P}(0)$ determines the initial point which is selected from the feasible region of the problem [i.e., $\boldsymbol{P}(0) \in \boldsymbol{\mathcal{X}}$] ,and $\hat{\boldsymbol{P}}(\upsilon)$ indicates the unique solution of $\boldsymbol{\mathcal{P}5}$ in the υth iteration of the algorithm. Parameter γ in line 20 indicates the step size in the algorithm which is obtained as $\gamma(\upsilon) = \gamma(\upsilon-1)(1-\alpha\gamma(\upsilon-1))$ where $\gamma(0) \in (0,1]$ and $\alpha \in (0, [1/\gamma(0)])$. Furthermore, other step size rules can be employed. The algorithm is terminated when $|\mathcal{O}^{\mathcal{K}}(\hat{\boldsymbol{P}}(\upsilon+1)) - \mathcal{O}^{\mathcal{K}}(\hat{\boldsymbol{P}}(\upsilon))| \le \delta$, in which δ determines the algorithm accuracy and $\boldsymbol{P}^{\mathrm{opt}} \triangleq (p_{i_n}^{\mathrm{opt}})_{i_n\in\mathcal{I}}$ represents the optimal solution that satisfies this condition.

F. Users Classification

So far, we have derived the optimum value of total overhead in the local and remote modes. In this step, we obtain the optimal decision between the local and remote modes. Recalling from (22), problem $\boldsymbol{\mathcal{P}3}$, that specifically focuses on deciding $S_{i_n}^{-1}$ for each user, turns to the following form:

$$\boldsymbol{\mathcal{P}6})\ \min_{S} \sum_{i_n\in\Omega} \left(\begin{array}{l} S_{i_n}^{-1}\mathcal{O}_{i_n}^{\mathrm{loc}}\left(f_{i_n}^{\mathrm{loc}}\right) \\ \quad + \left(1 - S_{i_n}^{-1}\right)\mathcal{O}_{i_n}^{\mathcal{K}}\left(p_{i_n}\right) \end{array} \right)$$
$$\text{s.t.}\ \ \textbf{C1}.\ S_{i_n}^{-1} \in \{0,1\}\ \forall i_n \in \mathcal{I}.$$

In order to obtain the binary coefficients $S_{i_n}^{-1}$, we can use lines 25–36 of Algorithm 1. In the first iteration of the algorithm, $\mathcal{O}_{i_n}^{\mathcal{K}}(p_{i_n})$ is compared to $\mathcal{O}_{i_n}^{\mathrm{loc}}(f_{i_n}^{\mathrm{loc}})$. For user i_n that $\mathcal{O}_{i_n}^{\mathcal{K}}(p_{i_n}) < \mathcal{O}_{i_n}^{\mathrm{loc}}(f_{i_n}^{\mathrm{loc}})$, we are sure that $S_{i_n}^{-1} = 0$ and collect this user index in Ω_1, while Ω_2 represents other user indexes (i.e., $\mathcal{O}_{i_n}^{\mathcal{K}}(p_{i_n}) > \mathcal{O}_{i_n}^{\mathrm{loc}}(f_{i_n}^{\mathrm{loc}})$) showing an optimal decision for members of this collection. Note that $\Omega = \Omega_1 + \Omega_2$. After that, users who are in the Ω_1 and Ω_2 vectors are sorted according to the total cost of the remote in the descending order of the value of Ω_2 as $\Omega_{2,\pi_1} \le \Omega_{2,\pi_2} \le \cdots \le \Omega_{2,\pi_{\mathrm{numel}(\Omega_2)}}$, where π is a permutation of $\mathrm{numel}(\Omega_2)$. Next, we concatenate the sorted vectors Ω_1 and Ω_2 to form Ω_0. By defining parameter λ as an index, in each iteration, first users up to λ are selected as users who receive service from the server (i.e., Ω). We define parameter j as the number of iterations in Algorithm 1. In the subsequent iterations of the algorithm (i.e., $j > 1$), the above cost is compared to the previous iteration, and by the manner which has been described in lines 25–51 of Algorithm 1, we achieve the optimal decision for users in Ω_2.

It is worth mentioning that the aforementioned stages in subsections *B-E* must operate iteratively until the convergence criterion $(\mu_{\mathrm{up}} - \mu_{\mathrm{low}}) > 1$ is satisfied. Then, for the indices first up to λ^* in the set Ω_0 (i.e., $i_n \in \Omega_0 = \{1,\ldots,\lambda^*\}$ in Algorithm 1), we set $S_{i_n}^{-1} = 0$ and for other indices, we set $S_{i_n}^{-1} = 1$. Accordingly, all users can be classified into three categories $\mathcal{Q}_{\mathcal{L}}$, $\mathcal{Q}_{\mathcal{MC}}$, and $\mathcal{Q}_{\mathcal{SC}}$. We summarize the aforementioned steps of our procedure as pseudocode in Algorithm 1. In addition, we will refer to our proposed joint access and resource allocation in the UD-IoT scheme as JARA-UDIoT in the next proposition.

Proposition 1 (Complexity Analysis): The computational complexity of JARA-UDIoT Algorithm 1 is of order $\mathcal{O}(\sum_{l=1}^{j}\mathcal{C}_l\Theta_l^2\Xi_l)$, where $\mathcal{C}_l$ represents the number of iterations needed for the convergence of SCA, Θ_l and Ξ_l denote the total number of IOT-MDs and offloading decision choices in the lth iteration, respectively.

Proof: Regarding to the proposed JARA-UDIoT in Algorithm 1, there are totally two-tier loops, containing an outer "for" loop (i.e., steps 1–9) and outer "while" loop (i.e., steps 10–51). The running time required for the first outer "for" loop in steps 1–9 is of order $\mathcal{O}(|\mathcal{I}|)$ and runs only one time for each user $i_n \in \mathcal{I}$, where $|\bullet|$ denotes the cardinality operator. The outer "while" loop contains two more main inner loops, including the inner "while" loop in steps 19–23, and the inner "for" loop in steps 25–36. The running time for the outer "while" loop mainly depends on the iteration times of steps 11–24, related to solving problem $\boldsymbol{\mathcal{P}5}$, where we employed the SCA algorithm. By this method, each case of problem $\boldsymbol{\mathcal{P}5}$ can be solved with the complexity of order $\mathcal{O}(\max\{x_j^3, x_j^2 y_j\})$ through employing interior points methods (IPM) [60, Ch. 1]. The subscript j indicates the iteration number of the algorithm, x_j is the total number of optimization variables, and y_j is the total number of constraints, namely, for $j=1$, $x_1 = |\mathcal{I}|$, $y_1 = 3|\mathcal{I}| + |\mathcal{M}|$. Hence, the corresponding computational complexity for $\boldsymbol{\mathcal{P}5}$ is of order $\mathcal{O}(\max\{|\mathcal{I}|^3, |\mathcal{I}|^2(3|\mathcal{I}| + |\mathcal{M}|)\}) = \mathcal{O}(|\mathcal{I}|^3 + |\mathcal{I}|^2|\mathcal{M}|)$. Denoting the total number of users and the total number of their offloading decisions in the lth iteration of the algorithm as Θ_l and Ξ_l, we have

$$\begin{aligned}\mathcal{O}\left(\max\left\{x_l^3, x_l^2 y_l\right\}\right) &= \mathcal{O}\left(\max\left\{\Theta_l^3, \Theta_l^2(3\Theta_l + \Xi_l)\right\}\right) \\ &= \mathcal{O}\left(\max\left\{\Theta_l^3, 3\Theta_l^3 + \Theta_l^2\Xi_l\right\}\right).\end{aligned}$$

$$\begin{aligned} R_{i_n}^{m_n}(p_{i_n}) &= W^{\mathcal{K}}\log_2\left(1 + \frac{p_{i_n}H_{i_n}^{m_n}}{1 + \sum_{j_n\in\varphi^{m_n}}\left(p_{j_n}H_{j_n}^{m_n}\mathbb{I}\left\{H_{j_n}^{m_n} \ge H_{i_n}^{m_n}\right\}\right)}\right) \\ &= \underbrace{W^{\mathcal{K}}\log_2\left(1 + \sum_{j_n\in\varphi^{mn}}\left(p_{j_n}H_{j_n}^{m_n}\mathbb{I}\left\{H_{j_n}^{m_n} \ge H_{i_n}^{m_n}\right\}\right) + p_{i_n}H_{i_n}^{m_n}\right)}_{R_{i_n}^{m_n+}(p_{i_n})} - \underbrace{W^{\mathcal{K}}\log_2\left(1 + \sum_{j_n\in\varphi^{m_n}}\left(p_{j_n}H_{j_n}^{m_n}\mathbb{I}\left\{H_{j_n}^{m_n} \ge H_{i_n}^{m_n}\right\}\right)\right)}_{R_{i_n}^{m_n-}(p_{i_n})} \end{aligned} \tag{33}$$

For the proposed UDN $\Xi_l > \Theta_l$, so the computational complexity of this part in the lth iteration can be more simplified as $\mathcal{O}(\Theta_l^2 \Xi_l)$. In addition, the running time of inner *"for"* loop (i.e., steps 25–36) is of order $\mathcal{O}(\Theta_l)$. Moreover, the process from steps 37 to 51 is run in an $\mathcal{O}(1)$ time. Finally, let us define $\mathcal{C}_l$ as the set of iterations needed for the convergence of the SCA in the inner "while" loop in the lth iteration, and accordingly, we define $\mathcal{C} \triangleq \{\mathcal{C}_1, \ldots, \mathcal{C}_j\}$. Therefore, the total running time of the JARA-UDIoT algorithm can be calculated by adding the aforementioned complexity terms, i.e., $\mathcal{O}(\sum_{l=1}^{j}(\mathcal{C}_l \Theta_l^2 \Xi_l + \Theta_l + 1) + |\mathcal{I}|) = \mathcal{O}(\sum_{l=1}^{j} \mathcal{C}_l \Theta_l^2 \Xi_l)$. ■

VI. Simulation Results

In this section, we verify the performance of our proposed NOMA-based offloading design algorithm in the UD-IoT networks, through the extensive simulations, and compare that with other benchmark methods.

1) *NOMA-Based Offloading:* We assume that all IoT MDs use the NOMA protocol to access the radio spectrum, where this case can be divided into the following three modes.
   a) *NOMA-Based Offloading (Multifrequency Band):* Small cells and macrocell work in the mmWave and microwave frequency band, respectively.
   b) *NOMA-Based Offloading (Microwave Frequency Band):* Small cells and macrocell work in the microwave frequency band.
   c) *NOMA-Based MEC Offloading (Microwave Frequency Band):* In this case, we consider a wireless network without any small cell where its users receive the service from the MEC server.
2) *OMA-Based Offloading:* We assume that all IoT MDs in each cluster use the frequency-division multiple access (FDMA) protocol to access the radio spectrum. In any frequency band (i.e., mmWave or microwave frequency band), the total available bandwidth $W^{\mathcal{K}}$, $\mathcal{K} \in \{\mathcal{MC}, \mathcal{SC}\}$, is equally and orthogonally assigned to its corresponding IoT MDs for task offloading. This case can be divided into three modes similar to the cases of NOMA-based offloading scenario. We name these cases as the OMA-based offloading (multifrequency band), OMA-based offloading (Microwave frequency band), and OMA-based MEC offloading (Microwave frequency band) in our simulations.
3) *Local Computing:* In this case, all users in the network execute their computational tasks only on their own device (i.e., $S_{i_n}^{-1} = 1 \ \forall i_n \in \mathcal{I}$). In this scheme, total overhead of the network is obtained by solving problem $\boldsymbol{\mathcal{P}2}$ according to (28).

We consider a centralized MEC scenario in an ultradense heterogeneous network which is covered by an MBS located at the center with 500 m in diameter. Furthermore, M SBSs with 100 m in radius are randomly scattered over the network. The MBS is equipped with an MEC server, whose computation capability is 4 GHz/s. MDs are randomly distributed over the coverage area. The input data size of the computation offloading and the total number of CPU cycles are uniformly distributed within $[300 - 800]$ KB and $[100 - 1000]$ megacycles, respectively. The detailed simulation parameters adopted in our performance evaluation, unless mentioned otherwise, are summarized in Table II.

TABLE II
Simulation Parameters

Parameter	Value
k_s	10^{-26}
θ_{i_n}	uniform $[300 - 800]$ KB
V_{i_n}	uniform $[0.1 - 1]$ Gigacycles
$\mathcal{T}_{i_n}^{\max}$	uniform $[0.5 - 5]$ sec
(δ, α)	$(10^{-3}, 10^{-5})$
$P_{i_n}^{\max}$	23 dBm
$E_{i_n}^{\max}$	5 Joule
N_c	2
c	4
W^{MEC}, W^{FoG}	2 MHz , 1 GHz
$\alpha_{\mathcal{MC}}$	3
$\alpha_{\mathcal{SC}}^{l}, \alpha_{\mathcal{SC}}^{n}$	4 , 2
$f^{\mathcal{MC}}, f^{\mathcal{SC}}$	2 GHZ , 70 GHZ
$G^{m_n,\max}$, $G^{m_n,\min}$	18 dBi, -2 dBi
$G^{\mathcal{MC}}$	0 dBi
$\beta^{\mathcal{K}}$	$\left(\frac{f^{\mathcal{K}}}{4\pi}\right)^2$
λ^{FoG,m_n}	1.5 MIPS
F^{MEC}, F^{FoG,m_n}	10 MIPS, 2 MIPS
F_b^{MEC}	10 MIPS

In order to validate the superiority of our proposed JARA-UDIoT algorithm, we compare latency, energy consumption, and the total overhead versus the number of IoT MDs with different benchmark schemes in Fig. 5. We assume that the number of IoT MDs in the network ranges from 9 to 48. It can be observed that the total execution delay, energy consumption, and total overhead increase in all schemes with an increment of the number of MDs in the network. Based on the results depicted in Fig. 5, we found that more cost is expended when all tasks are performed locally. It is also shown that the performance of the network is improved by employing the NOMA-based partial offloading scheme when compared to the OMA-based case. Through combining NOMA and the capabilities of mmWave for small cell users and also, employing the proposed JARA-UDIoT algorithm, the amount of the cost function significantly diminishes in comparison to other schemes. For example, for $N = 36$, the proposed scheme reduces the total cost of the network with the value of $20\%, 46\%, 76\%, 79\%, 81\%,$ and 90% for the schemes NOMA-based offloading (microwave frequency band), NOMA-based MEC offloading (microwave frequency band), OMA-based offloading (multifrequency band), OMA-based offloading (microwave frequency band) and OMA-based MEC offloading (microwave frequency band), and local computing, respectively. By comparing the results in Fig. 5, we can

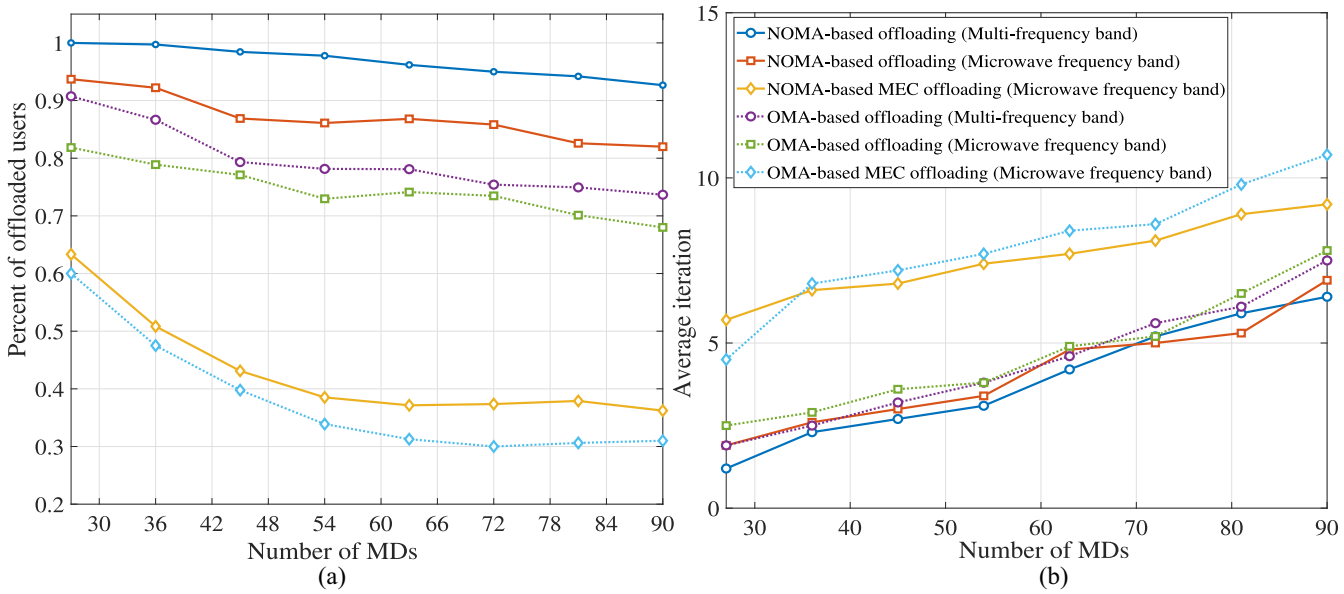

Fig. 3. Comparison of the performance of the proposed user classification algorithm for different schemes. (a) Percentage of IoT MDs who receive the service from the server. (b) Average algorithm iteration to achieve the desired solution (i.e., $\bar{j}$) versus the number of users.

conclude that the proposed JARA-UDIoT algorithm able to attain superior performance in comparison to other methods.

Furthermore, we investigate the performance of the proposed JARA-UDIoT algorithm from another perspective in Fig. 3. The percentage of offloaded users who are able to receive the service from each server[1] to accomplish their processing tasks has been illustrated versus the number of IoT MDs for different schemes in Fig. 3(a). In addition, the average number of iterations, denoted by $\bar{j}$, to achieve these results during simulation implementation is represented in Fig. 3(b). From Fig. 3(a), it is found that the number of users receiving the service from the server is the largest among all other schemes. In addition, it is seen from OMA-based MEC offloading (Microwave frequency band) that the percentage of users who able to offload their computational tasks to the server is the lowest between all schemes. More precisely, most of the users' requests to receive the service from the server are rejected. In order to reduce the overall cost of the network, it is cost effective for users to accomplish their processing locally on their own devices. It is worth mentioning that the main purpose of this article is minimizing the total cost, not maximizing the number of applicant users at any cost. In Fig. 3(a), the advantage of employing NOMA and mmWave jointly is clearly visible. Accordingly, the NOMA-based computation offloading has the best performance in comparison with its counterpart (OMA-based) among the studied schemes. In addition, it is observed from Fig. 3(a) that the percentage of users receiving the service from the server decreases in all the schemes by increasing the number of network's users. This is completely reasonable, since the number of requests sent to the server increases by the increment of MDs in the network, leading to an increase in the interference. Obviously, increment of the interference reduces the transmission data rates. Hence, users should spend more power for transmitting data to the server, resulting in an increase in the total cost of the network. In addition, the interference increases the latency of the user's task processing, due to the limitation of computing resources on the servers. Accordingly, for reducing the total cost of the network, it is advantageous that some users process their computational tasks locally on their devices.

We also examined the typical convergence speed of the proposed JARA-UDIoT algorithm in Fig. 3(b). For this purpose, we independently simulated ten times for various schemes and illustrated the average number of iterations to obtain the desired solutions (i.e., satisfying the termination criteria). From Fig. 3(b), we find that the proposed JARA-UDIoT algorithm displays the desirable solution with the least average repetitions when compared to the other schemes which indicate the fastest convergence time and the best performance of the proposed JARA-UDIoT algorithm.

Fig. 4 shows the impact of the chosen coefficients for weighting the energy consumption and execution delay on total overhead in the network under the different number of IoT MDs. For this purpose, we compare the applied weighting method (i.e., $\psi_{i_n} = \psi'_{i_n}\eta^E_{i_n}$) with the random weighting case. As mentioned earlier, to improve the network performance, we used the remaining energy available in each device to weigh the overhead for each user. Accordingly, for the user with smaller energy (i.e., more limitation is considered on energy consumption), the weight of the energy function is greater than the delay. In contrast, users without any energy limitation and with a high charge percentage, are assigned more weight to latency. By this procedure, we can weigh the overhead function for each user. As shown in Fig. 4, for $\psi_{i_n} = 0.9$ and $\psi_{i_n} = 0.1$, the amount of the network's latency and consumed energy have the lowest and highest values, respectively. This is reasonable since, with increasing the weight of the network delay

[1]From now on, the generic term "server" refers to the MBS and APs in small cells.

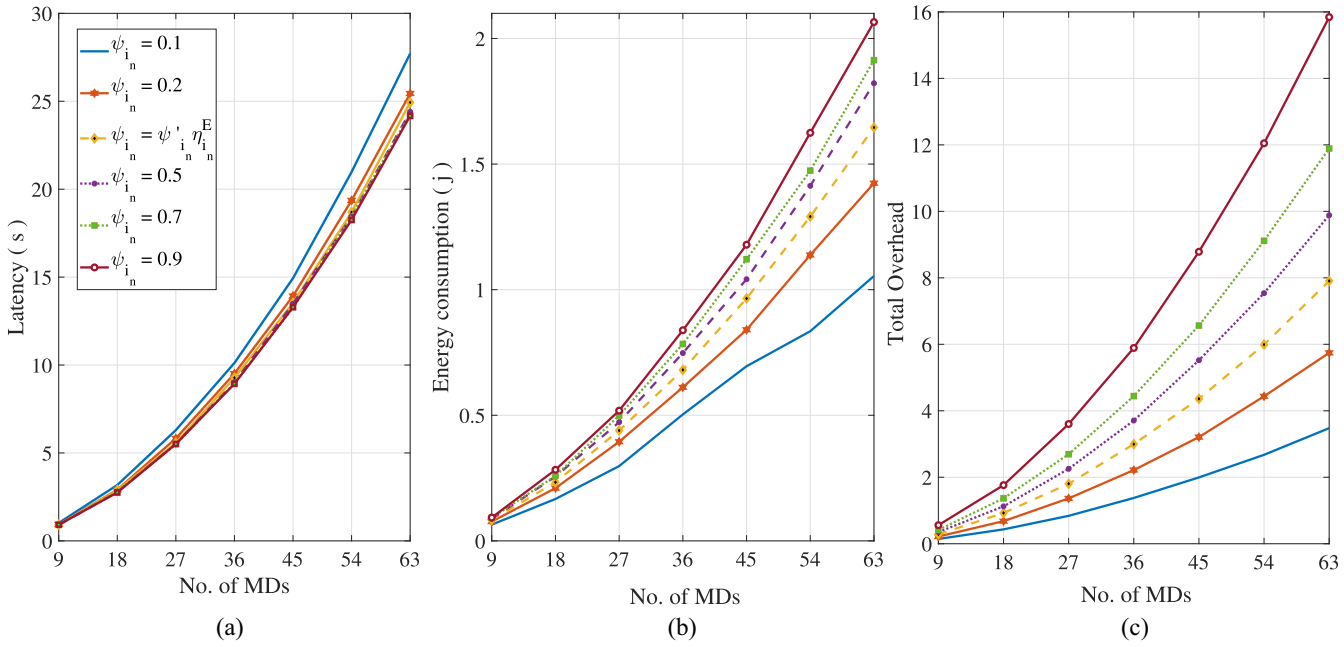

Fig. 4. Impact of weightings on latency, energy consumption, and total overhead.

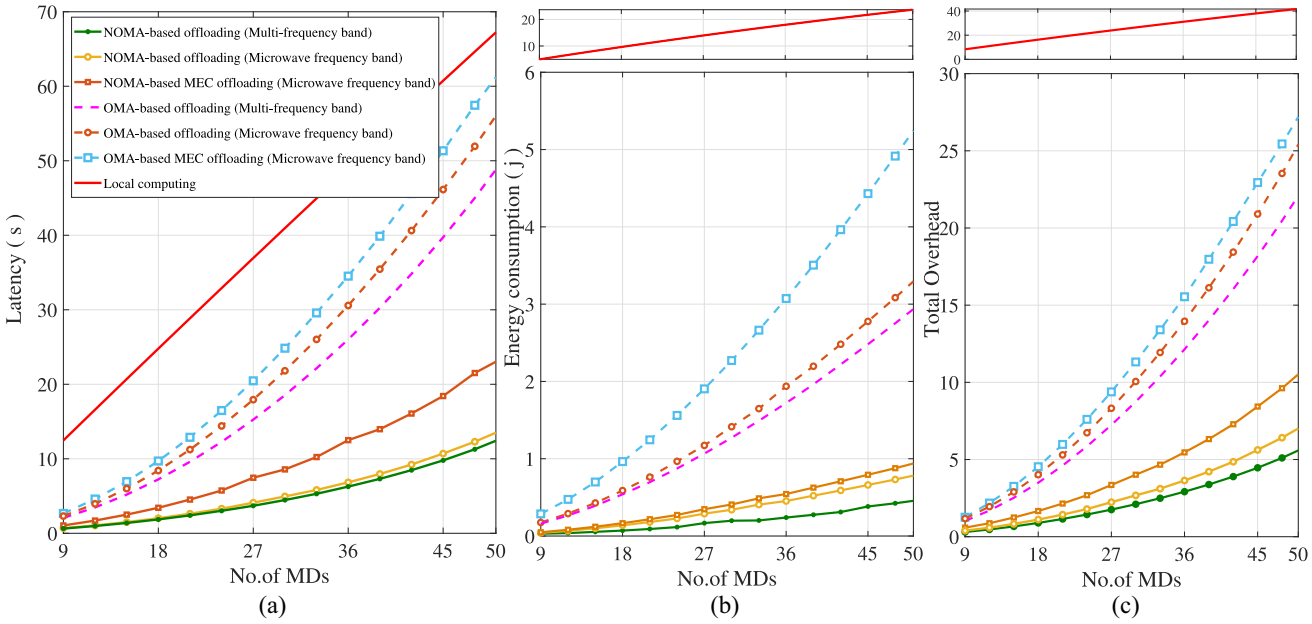

Fig. 5. Comparison results of the latency, energy consumption and total overhead versus the number of MDs by adopting various offloading schemes.

function, the proposed JARA-UDIoT algorithm proceeds in a way to minimize the network's latency as much as possible. By increasing the weight of the delay function, the amount of consumed energy naturally increases due to the lower weight of energy. This may lead to an increase in the power consumption of each user sending the data in the uplink direction. From Fig. 4, it is observed that this type of weighting procedure outperforms the random weighting method. For example, it is seen that the performance of this type of weighting is better than the weights of $\psi_{i_n} = 0.5, 0.7,$ and 0.9.

Fig. 6 depicts the effect of employing the proposed user classification algorithm on the total overhead. For this purpose, we evaluate the effect of the proposed JARA-UDIoT algorithm in NOMA-based offloading (multifrequency band) by calculating the total network cost versus the number of MDs in two cases. In the first case in NOMA-based offloading performed on the multifrequency band, it is assumed that the user classification is not applied for calculating the total overhead. In this case, we solve problem $\boldsymbol{\mathcal{P}1}$ by the combination of the SCA method and the interior penalty function. However, in the second case and using the proposed user classification algorithm (i.e., proposed JARA-UDIoT scheme) for NOMA-based offloading (multifrequency band), we can reduce significantly the overall cost of the network. It is worth mentioning that the aforementioned cases 1 and 2 have been marked in Fig. 6 as "Offloading without the proposed algorithm" and "Offloading

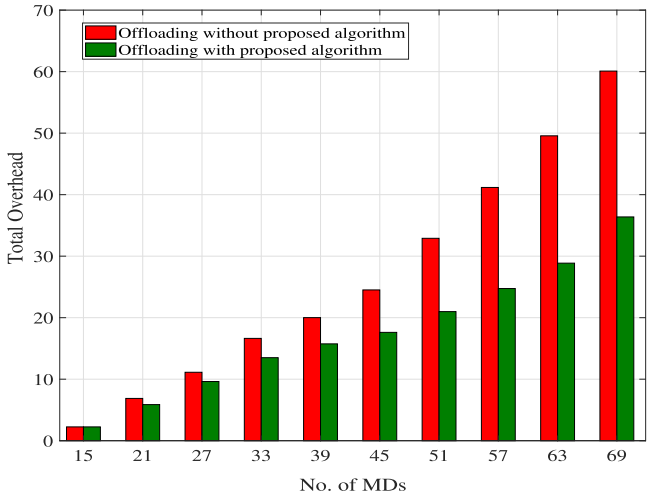

Fig. 6. Comparison the results of total overhead by adopting proposed algorithm versus the number of MDs.

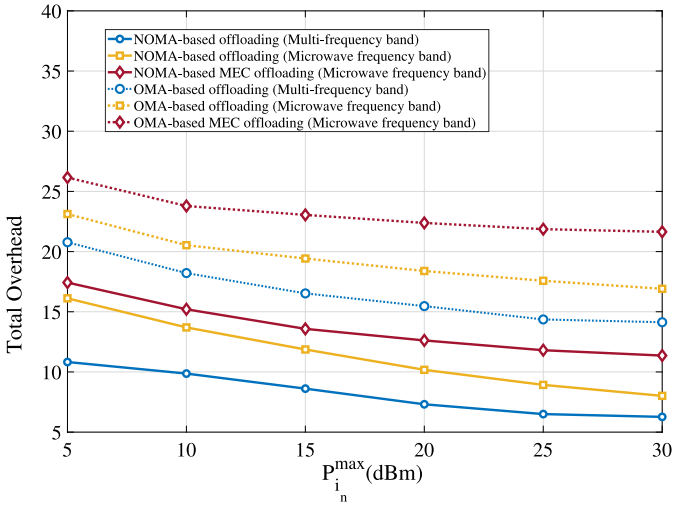

Fig. 7. Impact of maximum transmission power on total overhead.

with the proposed algorithm," respectively. We see in Fig. 6 that for $N = 15$, the performances of the two schemes (i.e., with and without using the proposed JARA-UDIoT algorithm) are exactly similar. This is mainly due to the fact that when there are few applicant's users in the IoT network, there is sufficient radio resource for the offloading tasks. Thereby, because of the lower interference in the network, users will spend less energy to offload their tasks to the server. Besides, for a few number of network's users, the amount of available computational resources on the servers, distributed among the applicant's users, is lower. This leads to a reduction in processing delays. Therefore, processing on the server is less costly than local processing. Consequently, it is advantageous for the entire users to employ the computing resources that are available on the server for processing. However, as mentioned earlier, the amount of energy and execution delays will increase with increment of the network's users due to increment of the interference and limitation in computing resources on servers. Since the main purpose is to reduce the total cost of the network, it can reduce the total cost by rejecting the requests of some network users. With increasing the number of MDs, the energy consumption and processing delays of some users in the remote mode will be greater than local processing, hence the rejection of users' requests will be occurred. As shown in Fig. 6, the proposed JARA-UDIoT algorithm shows a suitable performance in which more improvement can be achieved by increasing the number of MDs in the network. For example, for $N = 57$, it is seen that by using the proposed JARA-UDIoT algorithm in the network, the total network cost is reduced by 40%.

Finally, in Fig. 7, we investigate the impact of the maximum available power per device (in dBm) for different schemes. It is seen that the total cost of the network for different schemes is reduced by increasing $P_{i_n}^{\max}$. In fact, by the increment of $P_{i_n}^{\max}$, the transmission data rate to the server increases which causes a reduction in both energy and delay in the uplink direction. We find that the proposed algorithm can attain superior performance in total overhead by joint optimization of computation and communication resources. This indicates the comprehensiveness and verification of the proposed JARA-UDIoT algorithm.

Proposition 2 (Optimality Analysis): If the solving accuracy is set to zero (i.e., $\delta = 0$), the proposed JARA-UDIoT algorithm converges to the global optimal solution, otherwise, i.e., $\delta \gtrsim 0$, the JARA-UDIoT scheme converges to the near-optimal solution.

Proof: Since, $S_{i_n}^{m_n}$ is a binary parameter and the nonconvex optimization problem $\boldsymbol{\mathcal{P}1}$ is considered as an MINLP belonging to NP-hard problems, we cannot find an optimum solution in the polynomial time. Hence, we adopt the decomposition method to decompose the original problem $\boldsymbol{\mathcal{P}1}$ into two subproblems as: 1) the total cost in local mode (i.e., $\boldsymbol{\mathcal{P}2}$), and 2) the total cost in remote mode (i.e., $\boldsymbol{\mathcal{P}4}$). The aim of applying the decomposition method is to reduce the computational complexity and the number of iterations for converging to the optimal solution of the problem. It is straightforward to show that problem $\boldsymbol{\mathcal{P}2}$ is a convex optimization problem and global optimal solution for parameter $\boldsymbol{F}$ is achieved at either the stationary point of the objective function or one of the boundary points. In this way, the optimal solution for this problem has been obtained in a closed-form expression according to the (23)–(27). On the other hand, problem $\boldsymbol{\mathcal{P}4}$ is nonconvex. We employed an iterative search algorithm, namely, SCA, where a convex approximation of $\boldsymbol{\mathcal{P}4}$ should be obtained in each iteration and solve this convex problem using the IPM [60, Ch. 1]. Besides, we adopted the Frank and Wolf (FW) method [61] to obtain a desired convex approximation. It should be noted that the IPM converges to the stationary solution for nonconvex problems, but for the convex problem, the obtained stationary solution is equivalent to the global solution. According to the above arguments, by applying FW and IPM methods and setting $\delta = 0$, we can iteratively search the optimal solution for problem $\boldsymbol{\mathcal{P}4}$ [62]–[66]. There is only one parameter left to be optimized that is offloading decision vector $\boldsymbol{S}$. We have generalized the well-known bisection method for our problem to calculate the binary decision parameter $\boldsymbol{S}$ for the network's users. It is proved in [60] and [67] that the bisection method converges to the global solution. Taking the above considerations into account, we claimed that the solution obtained with the proposed algorithm converges to

the optimal global solution. In addition, it can be proved that for $\delta \gtrsim 0$, the JARA-UDIoT scheme converges to the near-optimal solution. ■

VII. Conclusion

The main contribution of this article was to analyze the tradeoff between the execution delay of computational tasks and the energy consumption of MDs in hybrid UD-IoT networks. In particular, we focused on a two-tier heterogeneous network with sub-6-GHz macrocells coexisting with dense mmWave small cells, where each MD opportunistically connects to one of them via the NOMA protocol. We merged the computation offloading scheme for MEC with mmWave backhaul connections and jointly optimized the computation and communication resources in an energy-aware MEC system. With the focus on minimizing the total cost function of MDs, we defined the total weighted energy and delay as the performance metric, in which the weighting factor was introduced according to known residual energy of MDs. Due to the fact that the outcoming problem was intractable nonconvex and MINLP, we proposed the JARA-UDIoT algorithm that employed the SCA scheme to search iteratively for the best solution under the energy and delay restrictions. The original optimization problem was simplified by the proposed algorithm that decreased the computation complexity, even though the final solutions were suboptimal. The extensive performance evaluation has been performed to illustrate the effectiveness of the proposed JARA-UDIoT algorithm by trace-driven simulations which proved the superior performance of our scheme when compared to the existing works in the literature review.

References

[1] J. Xu, J. Yao, L. Wang, Z. Ming, K. Wu, and L. Chen, "Narrowband Internet of Things: Evolutions, technologies, and open issues," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1449–1462, Jun. 2018.

[2] T. Gomes, F. Salgado, S. Pinto, J. Cabral, and A. Tavares, "A 6LoWPAN accelerator for Internet of Things endpoint devices," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 371–377, Feb. 2018.

[3] S. Zhang, N. Zhang, S. Zhou, J. Gong, Z. Niu, and X. Shen, "Energy-sustainable traffic steering for 5G mobile networks," *IEEE Commun. Mag.*, vol. 55, no. 11, pp. 54–60, Nov. 2017.

[4] L. P. Qian, Y. Wu, H. Zhou, and X. Shen, "Dynamic cell association for non-orthogonal multiple-access V2S networks," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 10, pp. 2342–2356, Oct. 2017.

[5] J. Zheng, Y. Wu, N. Zhang, H. Zhou, Y. Cai, and X. Shen, "Optimal power control in ultra-dense small cell networks: A game-theoretic approach," *IEEE Trans. Wireless Commun.*, vol. 16, no. 7, pp. 4139–4150, Jul. 2017.

[6] H. Ma, L. Liu, A. Zhou, and D. Zhao, "On networking of Internet of Things: Explorations and challenges," *IEEE Internet Things J.*, vol. 3, no. 4, pp. 441–452, Aug. 2016.

[7] H. Guo, J. Liu, Z. Fadlullah, and N. Kato, "On minimizing energy consumption in FiWi enhanced LTE-A HetNets," *IEEE Trans. Emerg. Topics Comput.*, vol. 6, no. 4, pp. 579–591, Oct. 2018.

[8] H. Guo, J. Liu, and L. Zhao, "Big data acquisition under failures in FiWi enhanced smart grid," *IEEE Trans. Emerg. Topics Comput.*, vol. 7, no. 3, pp. 420–432, Jul./Sep. 2019.

[9] T. Soyata, R. Muraleedharan, C. Funai, M. Kwon, and W. Heinzelman, "Cloud-vision: Real-time face recognition using a mobile-cloudlet-cloud acceleration architecture," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Jul. 2012, pp. 59–66.

[10] L. Liu, Z. Chang, X. Guo, S. Mao, and T. Ristaniemi, "Multiobjective optimization for computation offloading in fog computing," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 283–294, Feb. 2018.

[11] Z. Su, Q. Qi, Q. Xu, S. Guo, and X. Wang, "Incentive scheme for cyber physical social systems based on user behaviors," *IEEE Trans. Emerg. Topics Comput.*, to be published.

[12] F. Cicirelli *et al.*, "Edge computing and social Internet of Things for large-scale smart environments development," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2557–2571, Aug. 2018.

[13] J. Zheng, Y. Cai, Y. Wu, and X. Shen, "Dynamic computation offloading for mobile cloud computing: A stochastic game-theoretic approach," *IEEE Trans. Mobile Comput.*, vol. 18, no. 4, pp. 771–786, Apr. 2019.

[14] N. Nouri, A. Entezari, J. Abouei, M. Jaseemuddin, and A. Anpalagan, "Dynamic power-latency trade-off for mobile edge computation offloading in NOMA-based networks," *IEEE Internet Things J.*, to be published.

[15] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: A survey," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 450–465, Feb. 2018.

[16] G. Tanganelli, C. Vallati, and E. Mingozzi, "Edge-centric distributed discovery and access in the Internet of Things," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 425–438, Feb. 2018.

[17] D. López-Pérez, M. Ding, H. Claussen, and A. H. Jafari, "Towards 1 Gbps/ue in cellular systems: Understanding ultra-dense small cell deployments," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2078–2101, 4th Quart., 2015.

[18] T. S. Rappaport *et al.*, "Millimeter wave mobile communications for 5G cellular: It will work!" *IEEE Access*, vol. 1, pp. 335–349, 2013.

[19] A. Ghosh *et al.*, "Millimeter-wave enhanced local area systems: A high-data-rate approach for future wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1152–1163, Jun. 2014.

[20] J. Zheng, Y. Cai, Y. Wu, and X. S. Shen, "Stochastic computation offloading game for mobile cloud computing," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC)*, Jul. 2016, pp. 1–6.

[21] J. Zhang *et al.*, "Energy-latency tradeoff for energy-aware offloading in mobile edge computing networks," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2633–2645, Aug. 2018.

[22] M. Chen and Y. Hao, "Task offloading for mobile edge computing in software defined ultra-dense network," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 587–597, Mar. 2018.

[23] A.-C. Pang, W.-H. Chung, T.-C. Chiu, and J. Zhang, "Latency-driven cooperative task computing in multi-user fog-radio access networks," in *Proc. IEEE 37th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2017, pp. 615–624.

[24] Y.-Y. Shih, W.-H. Chung, A.-C. Pang, T.-C. Chiu, and H.-Y. Wei, "Enabling low-latency applications in fog-radio access networks," *IEEE Netw.*, vol. 31, no. 1, pp. 52–58, Jan./Feb. 2017.

[25] T. Yang, H. Zhang, H. Ji, and X. Li, "Computation collaboration in ultra dense network integrated with mobile edge computing," in *Proc. IEEE 28th Annu. Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, Oct. 2017, pp. 1–5.

[26] H. Guo, J. Liu, J. Zhang, W. Sun, and N. Kato, "Mobile-edge computation offloading for ultradense IoT networks," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4977–4988, Dec. 2018.

[27] J. Zhou, X. Zhang, and W. Wang, "Joint resource allocation and user association for heterogeneous services in multi-access edge computing networks," *IEEE Access*, vol. 7, pp. 12272–12282, 2019.

[28] Y. Liu, F. R. Yu, X. Li, H. Ji, H. Zhang, and V. C. M. Leung, "Joint access and resource management for delay-sensitive transcoding in ultra-dense networks with mobile edge computing," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2018, pp. 1–6.

[29] A. Kiani and N. Ansari, "Toward hierarchical mobile edge computing: An auction-based profit maximization approach," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 2082–2091, Dec. 2017.

[30] T. G. Rodrigues, K. Suto, H. Nishiyama, and N. Kato, "Hybrid method for minimizing service delay in edge cloud computing through VM migration and transmission power control," *IEEE Trans. Comput.*, vol. 66, no. 5, pp. 810–819, May 2017.

[31] J. Liu, H. Guo, Z. M. Fadlullah, and N. Kato, "Energy consumption minimization for FiWi enhanced LTE-A HetNets with UE connection constraint," *IEEE Commun. Mag.*, vol. 54, no. 11, pp. 56–62, Nov. 2016.

[32] N. Nouri, P. Rafiee, and A. Tadaion, "NOMA-based energy-delay trade-off for mobile edge computation offloading in 5G networks," in *Proc. IEEE 9th Int. Symp. Telecommun. (IST)*, Dec. 2018, pp. 522–527.

[33] H. Guo and J. Liu, "Collaborative computation offloading for multiaccess edge computing over fiber–wireless networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4514–4526, May 2018.

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

[35] W. Sun and J. Liu, "Coordinated multipoint-based uplink transmission in Internet of Things powered by energy harvesting," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2585–2595, Aug. 2018.
[36] Y. Mao, J. Zhang, and K. B. Letaief, "Dynamic computation offloading for mobile-edge computing with energy harvesting devices," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3590–3605, Dec. 2016.
[37] 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, no. 2, pp. 89–103, Jun. 2015.
[38] 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, no. 12, pp. 12313–12325, Dec. 2018.
[39] E. Cuervo *et al.*, "MAUI: Making smartphones last longer with code offload," in *Proc. 8th Int. Conf. Mobile Syst. Appl. Services*, Jun. 2010, pp. 49–62.
[40] W. Zhang, Y. Wen, K. Guan, D. Kilper, H. Luo, and D. O. Wu, "Energy-optimal mobile cloud computing under stochastic wireless channel," *IEEE Trans. Wireless Commun.*, vol. 12, no. 9, pp. 4569–4581, Sep. 2013.
[41] H. Elshaer, M. N. Kulkarni, F. Boccardi, J. G. Andrews, and M. Dohler, "Downlink and uplink cell association with traditional macrocells and millimeter wave small cells," *IEEE Trans. Wireless Commun.*, vol. 15, no. 9, pp. 6244–6258, Sep. 2016.
[42] K. Higuchi and Y. Kishiyama, "Non-orthogonal access with random beamforming and intra-beam SIC for cellular MIMO downlink," in *Proc. IEEE 78th Veh. Technol. Conf.*, Sep. 2013, pp. 1–5.
[43] Y. Endo, Y. Kishiyama, and K. Higuchi, "Uplink non-orthogonal access with MMSE-SIC in the presence of inter-cell interference," in *Proc. Int. Symp. Wireless Commun. Syst. (ISWCS)*, Aug. 2012, pp. 261–265.
[44] J. Umehara, Y. Kishiyama, and K. Higuchi, "Enhancing user fairness in non-orthogonal access with successive interference cancellation for cellular downlink," in *Proc. IEEE Int. Conf. Commun. Syst. (ICCS)*, Nov. 2012, pp. 324–328.
[45] N. Otao, Y. Kishiyama, and K. Higuchi, "Performance of non-orthogonal access with SIC in cellular downlink using proportional fair-based resource allocation," in *Proc. IEEE Int. Symp. Wireless Commun. Syst. (ISWCS)*, Aug. 2012, pp. 476–480.
[46] F. Fang, H. Zhang, J. Cheng, and V. C. M. Leung, "Energy-efficient resource allocation for downlink non-orthogonal multiple access network," *IEEE Trans. Commun.*, vol. 64, no. 9, pp. 3722–3732, Sep. 2016.
[47] P. Sedtheetorn and T. Chulajata, "Uplink spectral efficiency for non-orthogonal multiple access in Rayleigh fading," in *Proc. IEEE 18th Int. Conf. Adv. Commun. Technol. (ICACT)*, Jan. 2016, pp. 751–754.
[48] M. Shi, K. Yang, Z. Han, and D. Niyato, "Coverage analysis of integrated sub-6GHz-mmWave cellular networks with hotspots," *IEEE Trans. Commun.*, vol. 67, no. 11, pp. 8151–8164, Nov. 2019.
[49] R. Li, X. Liu, K. Luo, T. Jiang, and S. Jin, "Decoupled access in HetNets with backhaul constrained small base stations," *IEEE Access*, vol. 6, pp. 27028–27038, 2018.
[50] B. Ngo and H. Lee, "Analysis of a pre-emptive priority *M/M/c* model with two types of customers and restriction," *Electron. Lett.*, vol. 26, no. 15, pp. 1190–1192, Jul. 1990.
[51] M. Jia, J. Cao, and W. Liang, "Optimal cloudlet placement and user to cloudlet allocation in wireless metropolitan area networks," *IEEE Trans. Cloud Comput.*, vol. 5, no. 4, pp. 725–737, Oct./Dec. 2017.
[52] Y. Wang, X. Lin, and M. Pedram, "A nested two stage game-based optimization framework in mobile cloud computing system," in *Proc. IEEE 7th Int. Symp. Service Oriented Syst. Eng.*, Mar. 2013, pp. 494–502.
[53] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Trans. Netw.*, vol. 24, no. 5, pp. 2795–2808, Oct. 2016.
[54] Y. Yu, J. Zhang, and K. B. Letaief, "Joint subcarrier and CPU time allocation for mobile edge computing," in *Proc. IEEE Glob. Commun. Conf. (GLOBECOM)*, Dec. 2016, pp. 1–6.
[55] K. Zhang *et al.*, "Energy-efficient offloading for mobile edge computing in 5G heterogeneous networks," *IEEE Access*, vol. 4, pp. 5896–5907, 2016.
[56] K. Zhu and E. Hossain, "Virtualization of 5G cellular networks as a hierarchical combinatorial auction," *IEEE Trans. Mobile Comput.*, vol. 15, no. 10, pp. 2640–2654, Oct. 2016.
[57] J. Abouei, A. Bayesteh, M. Ebrahimi, and A. K. Khandani, "Virtual cooperation for throughput maximization in distributed large-scale wireless networks," *EURASIP J. Adv. Signal Process.*, vol. 2011, p. 2, Oct. 2011.
[58] G. Scutari, F. Facchinei, and L. Lampariello, "Parallel and distributed methods for constrained nonconvex optimization: Part I: Theory," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 1929–1944, Apr. 2017.
[59] G. Scutari, F. Facchinei, L. Lampariello, S. Sardellitti, and P. Song, "Parallel and distributed methods for constrained nonconvex optimization—Part II: Applications in communications and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 1945–1960, Apr. 2017.
[60] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[61] P. Apkarian and H. D. Tuan, "Robust control via concave minimization local and global algorithms," *IEEE Trans. Autom. Control*, vol. 45, no. 2, pp. 299–305, Feb. 2000.
[62] D. T. Ngo, S. Khakurel, and T. Le-Ngoc, "Joint subchannel assignment and power allocation for OFDMA femtocell networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 1, pp. 342–355, Jan. 2014.
[63] H. H. Kha, H. D. Tuan, and H. H. Nguyen, "Fast global optimal power allocation in wireless networks by local DC programming," *IEEE Trans. Wireless Commun.*, vol. 11, no. 2, pp. 510–515, Feb. 2011.
[64] M. Chiang, C. W. Tan, D. P. Palomar, D. O'neill, and D. Julian, "Power control by geometric programming," *IEEE Trans. Wireless Commun.*, vol. 6, no. 7, pp. 2640–2651, Jul. 2007.
[65] J. Papandriopoulos and J. S. Evans, "SCALE: A low-complexity distributed protocol for spectrum balancing in multiuser DSL networks," *IEEE Trans. Inf. Theory*, vol. 55, no. 8, pp. 3711–3724, Aug. 2009.
[66] S. Rezvani, S. Parsaeefard, N. Mokari, M. R. Javan, and H. Yanikomeroglu, "Cooperative multi-bitrate video caching and transcoding in multicarrier NOMA-assisted heterogeneous virtualized MEC networks," *IEEE Access*, vol. 7, pp. 93511–93536, 2019.
[67] K. Sikorski, "Bisection is optimal," *Numerische Mathematik*, vol. 40, no. 1, pp. 111–117, 1982.

Nima Nouri (S'17) received the B.Sc. degree in communication engineering from the Shahid Bahonar University of Kerman, Kerman, Iran, in 2014, and the M.Sc. degree in communication systems engineering from Yazd University, Yazd, Iran, in 2017.

Since 2017, he has been a Research Assistant with the WINEL Lab, Yazd University. His main research interests include Internet of Things, 5G communication systems, edge/fog computing, resource allocation, and nonconvex optimization.

Jamshid Abouei (S'05–M'11–SM'13) received the B.Sc. degree in electronics engineering and the M.Sc. degree (Highest Hons.) in communication systems engineering from the Isfahan University of Technology, Isfahan, Iran, in 1993 and 1996, respectively, and the Ph.D. degree in electrical engineering from the University of Waterloo, Waterloo, ON, Canada, in 2009.

He joined the Department of Electrical Engineering, Yazd University, Yazd, Iran, in 1996, as a Lecturer, where he was promoted to an Assistant Professor in 2010 and an Associate Professor in 2015. From 1998 to 2004, he served as a Technical Advisor and Design Engineer with the Research and Development Center and Cable Design Department, SGCC, Tehran, Iran. From 2009 to 2010, he was a Postdoctoral Fellow with the Multimedia Lab, Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada, and worked as a Research Fellow with the Self-Powered Sensor Networks (ORF-SPSNs) consortium. He was an Associate Researcher with the Department of Electrical, Computer and Biomedical Engineering, Ryerson University, Toronto. He was the International Relations Chair in 27th ICEE2019 Conference, Iran, in 2019. He currently directs the Research Group with the Wireless Networking Laboratory, Yazd University. His research interests are in the next generation of wireless networks (5G) and wireless sensor networks, with a particular emphasis on PHY/MAC layer designs, including the energy efficiency and optimal resource allocation in cognitive cell-free massive MIMO networks, multiuser information theory, mobile edge computing, and femtocaching.

Dr. Abouei is a recipient of the Best Paper Award for the IEEE Iranian Conference on Electrical Engineering in 2018. He has received several awards and scholarships, including FOE and IGSA awards for excellence in research in University of Waterloo, MSRT Ph.D. Scholarship from the Ministry of Science, Research and Technology, Iran, in 2004, Distinguished Researcher Award in province of Yazd, Iran, in 2011, and Distinguished Researcher Award in Electrical Engineering Department, Yazd University in 2013. He is a member of the IEEE Information Theory.

Muhammad Jaseemuddin (M'98) received the B.E. degree from N.E.D. University, Karachi, Pakistan, the M.S. degree from the University of Texas at Arlington, Arlington, TX, USA, and the Ph.D. degree from the University of Toronto, Toronto, ON, Canada.

He worked with Advanced IP group and Wireless Technology Lab, Nortel Networks, Ottawa, ON, Canada. He is a Professor and the Program Director of Computer Networks Program, Ryerson University, Toronto. His research interests include network automation, caching in 5G and ICN networks, context-aware mobile middleware and mobile cloud, localization, power-aware MAC and routing for sensor networks, heterogeneous wireless networks, and IP routing and traffic engineering.

Alagan Anpalagan (S'98–M'01–SM'04) received the B.A.Sc., M.A.Sc., and Ph.D. degrees in electrical engineering from the University of Toronto, ON, Canada.

He joined with the ELCE Department, Ryerson University, Toronto, in 2001, where he was promoted to a Full Professor in 2010, and served in administrative positions as the Associate Chair, the Program Director for Electrical Engineering, and the Graduate Program Director. He was a Visiting Professor with Asian Institute of Technology, Nueng, Thailand, and a Visiting Researcher with Kyoto University, Kyoto, Japan. He directs a Research Group working on radio resource management and radio access and networking areas within the WINCORE Laboratory, Ryerson University. He has coauthored four edited books and two books in wireless communication and networking areas. His industrial experience includes working for three years with Bell Mobility, Mississauga, ON, Canada, Nortel Networks, Ottawa, ON, and IBM, Armonk, NY, USA.

Dr. Anpalagan was a recipient of the IEEE Canada J.M. Ham Outstanding Engineering Educator Award in 2018, the YSGS Outstanding Contribution to Graduate Education Award in 2017, the Deans Teaching Award in 2011, the Faculty Scholastic, Research and Creativity Award thrice from the Ryerson University, the IEEE M.B. Broughton Central Canada Service Award in 2016, the Exemplary Editor Award from IEEE ComSoc in 2013 and the Editor-in-Chief Top10 Choice Award in Transactions on Emerging Telecommunications Technology in 2012, and the coauthor of a paper that received IEEE SPS Young Author Best Paper Award in 2015. He served as an Editor for the IEEE COMMUNICATIONS SURVEYS AND TUTORIALS from 2012 to 2014, IEEE COMMUNICATIONS LETTERS from 2010 to 2013, and *EURASIP Journal of Wireless Communications and Networking* from 2004 to 2009. He also served as Guest Editor for six special issues published in IEEE, IET, and ACM. He served as the TPC Co-Chair for IEEE VTC Fall 2017, and the TPC Co-Chair for IEEE INFOCOM'16: Workshop on Green and Sustainable Networking and Computing, IEEE Globecom15: SAC Green Communication and Computing, and IEEE PIMRC'11: Cognitive Radio and Spectrum Management. He served as the Vice Chair for IEEE SIG on Green and Sustainable Networking and Computing with Cognition and Cooperation from 2015 to 2018, IEEE Canada Central Area Chair from 2012 to 2014, IEEE Toronto Section Chair from 2006 to 2007, ComSoc Toronto Chapter Chair from 2004 to 2005, and IEEE Canada Professional Activities Committee Chair from 2009 to 2011. He is a Registered Professional Engineer in the province of Ontario, Canada. He is a fellow of the Institution of Engineering and Technology and the Engineering Institute of Canada.