Hybrid NOMA-FDMA Assisted Dual Computation Offloading: A Latency Minimization Approach

Yang Li, Yuan Wu, *Senior Member, IEEE*, Minghui Dai, Bin Lin, *Senior Member, IEEE*, Weijia Jia, *Fellow, IEEE*, and Xuemin Shen, *Fellow, IEEE*

***Abstract*—Edge computing has been considered as a promising solution for enabling computation-intensive yet latency-sensitive applications at resource-constrained wireless devices (WDs). In this paper, exploiting the advanced small-cell dual connectivity (DC), we investigate a paradigm of dual computation offloading in which a WD can simultaneously offload partial workloads to a cloudlet-server co-located at the macro base station (MBS) and an edge-server (ES) co-located at a small-cell based station (SBS). To facilitate the multi-user dual computation offloading, we exploit a hybrid model of non-orthogonal multiple access (NOMA) and frequency division multiple access (FDMA). Specifically, due to the SBSs' limited channel resources, we consider that the WDs form different NOMA-groups for offloading their respective workloads to different SBSs, which improves the spectrum efficiency. Meanwhile, all WDs use FDMA for offloading their workloads to the MBS, which avoids the WDs' co-channel interference. We formulate a joint optimization of the WDs' partial offloading decisions, their FDMA transmission to the MBS, different NOMA-groups' transmission to the SBSs, as well as the computing-rate allocation of the ESs and the cloudlet-server, with the objective of minimizing the overall latency for completing all WDs' workloads. Despite the strict non-convexity of the joint optimization problem, we propose a layered yet cell-based distributed algorithm for obtaining the optimal dual offloading solution. Based on the optimal dual offloading solution, we further investigate how to properly group WDs into different NOMA-groups for offloading workloads to the corresponding SBSs, and propose a cross-entropy based learning algorithm for finding the optimal NOMA grouping scheme. Numerical results are finally provided to validate the effectiveness and efficiency of our proposed algorithms.**

***Index Terms*—Dual computation offloading, hybrid NOMA-FDMA transmission, joint computation offloading and resource allocation.**

Manuscript received September 27, 2021; revised April 15, 2022; accepted May 14, 2022. Date of publication May 23, 2022; date of current version September 9, 2022. This work was supported in part by the Intergovernmental International Cooperation in Science and Technology Innovation Program under Grant 2019YFE0111600, in part by FDCT-MOST Joint Project under Grant 0066/2019/AMJ, in part by the National Natural Science Foundation of China under Grant 62072490, in part by the Science and Technology Development Fund of Macau SAR under Grants 0162/2019/A3 and 0060/2019/A1, in part by FDCT SKL-IOTSC(UM)-2021-2023 in part by Guangdong Basic and Applied Basic Research Foundation under Grant 2022A1515011287, in part by the Research Grant of University of Macau under Grant MYRG2020-00107-IOTSC, and in part by the Natural Sciences and Engineering Research Council of Canada. Recommended for acceptance by Dr. Yan Zhang. *(Corresponding author: Yuan Wu)*

Yang Li and Minghui Dai are with the State Key Laboratory of Internet of Things for Smart City and Department of Computer and Information Science, University of Macau, Taipa, Macau, China (e-mail: yb87469@um.edu.mo; minghuidai@um.edu.mo).

Yuan Wu is with the State Key Laboratory of Internet of Things for Smart City and Department of Computer and Information Science, University of Macau, Taipa, Macau, China, also with the Department of Computer Information Science, University of Macau, Taipa, China, and also with Zhuhai-UM Science and Technology Research Institute, Zhuhai 519013, China (e-mail: yuanwu@um.edu.mo).

Bin Lin is with the Department of Communication Engineering, Dalian Maritime University, Dalian 116026, China, and also with the Network Communication Research Centre, Peng Cheng Laboratory, Shenzhen 518052, China (e-mail: binlin@dlmu.edu.cn).

Weijia Jia is with the Guangdong Key Lab of AI and Multi-Modal Data Processing, BNU-UIC Institute of Artificial Intelligence and Future Networks, Beijing Normal University (BNU), BNU-HKBU United International College, Zhuhai, Guangdong 519087, China (e-mail: jiawj@uic.edu.cn).

Xuemin Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo ON N2L 3G1, Canada (e-mail: sshen@uwaterloo.ca).

Digital Object Identifier 10.1109/TNSE.2022.3176924

I. Introduction

MOBILE edge computing (MEC), a paradigm that deploys computation resources at the edge of radio access networks (RANs), has provided a promising approach for enabling computation-intensive yet latency-sensitive applications [1], [2]. Thanks to the computation offloading technology, resource-constrained wireless devices (WDs) can offload their tasks to nearby edge-servers (ESs) via a short-distance wireless transmission. In addition, uploading tasks to edge-servers consumes less energy and latency than uploading the tasks to the remote cloud center. The benefits of computation offloading in MEC have motivated lots of research efforts from both academia and industries. The single cell scenario in which a set of WDs offload their tasks to one base station was studied in [3], [4]. The multi-cell computation offloading scenario in which WDs can select one of cells to offload their tasks to different ESs was studied in [5], [6]. In addition, the multi-cell multi-tier offloading scenario was studied in [7], [8], in which each ES can further upload its received workloads to the remote cloud center via high-speed backhaul link. However, the explosive growth in edge traffic and the number of connected WDs still imposes a great pressure on the task transmission from the WDs and the ESs.

The emerging multi-access MEC provides a promising solution for addressing the above issue [9]. The WDs in multi-access MEC are allowed to offload their tasks to different ESs simultaneously by exploiting the feature of multi-homing in heterogeneous RANs. The small-cell dual connectivity (DC) [10], [11] provides a promising approach for enabling the multi-access edge computing. The DC allows each WD to be simultaneously associated with a macro base station (MBS) and a small-cell

base station (SBS) and thus to schedule its traffic towards the MBS and SBS flexibly [12], [13]. Such a DC assisted MEC can accommodate the dual computation offloading to improve the offloading efficiency for multiple WDs. Cui *et. al.* [14] studied a DC assisted offloading scheme in which a user terminal connects with a SBS and a MBS simultaneously via orthogonal multiple access (OMA). Li *et. al.* [15] investigated a similar DC assisted computation offloading via non-orthogonal multiple access (NOMA) transmission. NOMA has been envisioned as a highly-efficient multiple access scheme for enabling massive connectivity in future wireless networks [16]–[19]. Exploiting NOMA, a group of WDs can simultaneously offload their workloads to an ES over the same frequency channel, which thus improves the efficiency of workloads transmission. However, few works investigate a hybrid NOMA-OMA assisted dual computation offloading scheme, which is highly flexible but complicated for determining dual offloading decisions and resource allocation.

In this work, we thus consider a hybrid NOMA and frequency division multiple access (FDMA) assisted dual computation offloading scheme with the objective of minimizing the overall latency for completing all WDs' tasks. Thanks to DC, each WD can execute a *dual computation offloading*, i.e., simultaneously offloading its partial workloads to one of ESs through the SBS and another partial workloads to the cloudlet-server[1] through the MBS. To enable this dual computation offloading, we propose a hybrid NOMA-FDMA transmission scheme. The MBS has sufficient channel resources, and thus it adopts FDMA to accommodate a large number of WDs' offloading transmission, which can avoid the WDs' co-channel interference. Each SBS has limited channel resources, and it thus adopts NOMA to accommodate a small number of WDs' offloading transmission, which improves the spectrum efficiency but suffers from the WDs' co-channel interference. Although the proposed hybrid NOMA-FDMA assisted dual computation offloading scheme can improve the offloading efficiency and provide a highly flexible computing service, it is challenging to determine the dual offloading decisions and resource allocation. In particular, the WDs' co-channel interference in NOMA may degrade the performance of workloads transmission, which makes how to group WDs into different NOMA-groups for offloading workloads to ESs an important issue. Moreover, such issue becomes more complicated in dual computation offloading scenario, because the fact that all WDs share the cloudlet-server's computing-rate makes WDs' dual offloading decisions coupled. To address these technical challenges, we decompose the formulated problem and propose three algorithms to solve the decomposed problem, respectively. Our key contributions can be summarized as follows.

- We firstly consider that the WDs' NOMA-groupings are given in advance, and investigate a joint optimization of all WDs' dual offloading decisions, the hybrid NOMA-FDMA transmission duration, and the computing-rate allocation, with the objective of minimizing the overall latency for completing all WDs' workloads.
- Despite the non-convexity of the joint optimization problem, we exploit its layered structure and decompose it into a top-problem for optimizing the overall latency and the consequent subproblem for optimizing the WDs' partial offloading decisions, the computing and communication resource allocation. In particular, regarding the subproblem, we propose a cell-based distributed algorithm, in which each SBS can individually optimize its own decision-variables without being coupled with the other SBSs. Based on the solution of the subproblem, we propose a two-layer hybrid search algorithm to solve the top-problem for finding the minimum overall latency.
- After obtaining the optimal dual computation offloading solution under a given WDs' NOMA-grouping, we further investigate how to properly group the WDs into NOMA-groups for offloading their workloads to different ESs, with the objective of further minimizing the overall latency. In particular, by exploiting our previous layered algorithm as a subroutine, we adopt the cross-entropy based learning algorithm [22] to determine the optimal WDs' NOMA-grouping for further minimizing the overall latency.
- We provide extensive numerical results to validate the effectiveness and efficiency of our proposed algorithms. Simulation results demonstrate that, under the given WDs' NOMA-groupings, our layered algorithm can achieve the nearly minimum latency in comparison with the globally optimal solution provided by LINGO [23]. Meanwhile, our proposed algorithms can effectively reduce the computing time compared to LINGO. Moreover, the numerical results also validate that the cross-entropy based learning algorithm can find the optimal WDs' NOMA-grouping and outperform some benchmark grouping schemes.

The remainder of this paper is organized as follows. We review the related studies in Section II. We present the system model and problem formulation in Section III. The decomposition of the problem is illustrated in Section IV, and our layered algorithms are shown in Section V. We demonstrate the performance evaluations in Section VI. We further investigate the WDs' NOMA-groupings in Section VII. We finally conclude this work in Section VIII and discuss the future directions.

II. Literature Review

We review the related studies in this section, by focusing on (i) the joint computation offloading and resource allocation for MEC, and (ii) the NOMA assisted computation offloading.

Achieving the benefits of MEC necessitates a joint management of the computation offloading and communication/computing resource allocation, which thus has attracted lots of research efforts from various fields [24]. For instance, in [25],

[1] The cloudlet is a small-sized cloud-server comprised of powerful multi-core computing units, and it can be deployed at the edge of networks for facilitating computation offloading with a low latency [20], [21].

Wang *et. al.* studied an intelligent dynamic computation offloading for smart Internet of Things (IoT) system. In [26], Zhang *et. al.* proposed an vehicular task offloading scheme in vehicular MEC to distributively schedule computation task offloading and edge resource allocation. In [27], Lim *et. al.* investigated a joint resource allocation and incentive mechanism design for edge intelligence. Different performance matrices have been considered to evaluate the joint computation offloading and resource allocation schemes. In [5], [7], [28], energy minimization has been studied by jointly optimizing multi-user computation offloading and user association. In [29], [30], the sum of weighted energy and latency minimization has been considered by jointly optimizing the binary/partial offloading decision and resource allocation. Several works [25], [31], [32] considered the time-varying environments (e.g., the channel condition, the number of the arriving tasks, and the capability of available resource) and designed the corresponding dynamic edge resource allocation schemes. To further improve the flexibility of computation offloading, there have been many studies investigating how to jointly exploit the resources at the ESs and those at the cloud servers [21]. In particular, to facilitate a flexible usage of the computation-resources at the ESs and cloud, [33], [34] investigated a two-tier edge-cloud framework with the hierarchical offloading scheme, i.e., a mobile user can offload its computation workloads to an ES, and the ES can further offload part of its received workloads to the cloud or other ESs. Different from the hierarchical offloading scheme, the binary task offloading scheme has been investigated in [35]–[37], in which each task can be either processed locally, or offloaded to the ES, or offloaded to the cloud.

Thanks to its advantage in accommodating massive connectivity with ultra-high spectrum-efficiency, NOMA has been regarded as one of the key technologies for enabling the next generation multiple access [16]–[19]. A comprehensive review of leveraging NOMA for future 6G networks has been proposed in [16]. In [17], Wei *et. al.* investigated the ergodic sum-rate gain of NOMA over the uplink cellular network and demonstrated the potential gain compared to that of OMA. In [18], Liu *et. al.* proposed a quality of service (QoS)-guarantee resource allocation scheme for multi-beam satellite industrial Internet of Things by using NOMA transmission. In [19], a NOMA assisted federated learning via wireless power transfer has been proposed. The benefit of NOMA has attracted many studies exploiting NOMA for offloading transmission [38]–[41]. In [38], the authors exploited uplink NOMA to enable the multi-user' task offloading, with the objective of minimizing a total cost including the completion time of the tasks and all users' total energy consumption. In [39], a joint power and time allocation scheme has been proposed for users' computation offloading via NOMA transmission. In [40], the authors studied a NOMA assisted multi-access computation offloading with the objective of minimizing the overall delay of all users to complete their tasks. In [41], the authors investigated a NOMA-assisted MEC for multi-user secure computation offloading in the presence of an eavesdropper. In addition to solely rely on NOMA, there are growing interests in exploring the hybrid NOMA and OMA for multi-user transmission. In [42], a hybrid NOMA-OMA transmission scheme has been proposed for computation offloading in MEC systems. However, few studies consider the hybrid transmission assisted dual computation offloading scenario, which is highly flexible but complicated for determining dual offloading decisions and resource allocation.

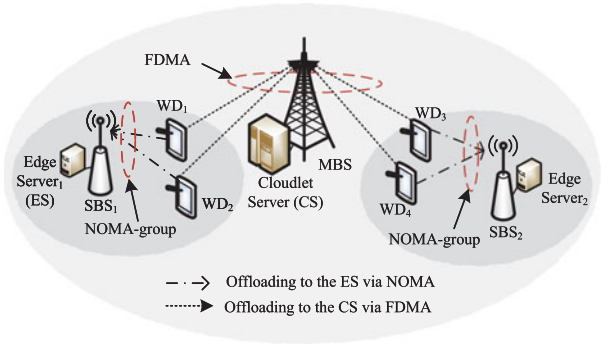


Fig. 1. A two-tier RAN with one MBS and two of SBSs. Each WD offloads parts of its computation-workloads to the MBS and one of the SBSs simultaneously.

In this work, we consider a DC enabled dual computation offloading, in which each WD is simultaneously associated with a SBS and the MBS, and thus can jointly exploit the computation-resources at the SBS and those at the MBS. In particular, to facilitate the multi-user dual offloading, we consider that the MBS adopts FDMA to accommodate different WDs' offloaded workloads, which thus avoids their co-channel interference, and each SBS adopts NOMA to accommodate the WDs' offloaded workloads, which improves the spectrum-efficiency but incurs the WDs' co-channel interference. This dual computation offloading scheme via hybrid NOMA and FDMA differs our work from many existing studies which focus on either the hierarchical edge-cloud offloading [33], [34] or the selective edge/cloud offloading scheme in [35]–[37]. Based on our dual offloading model, we thus investigate the joint optimization of the WDs' partial offloading, NOMA and FDMA transmission, and computing-rate allocation to minimize the overall task-completion latency for all WDs.

III. System Model and Problem Formulation

A. System Model

As shown in Fig. 1, we consider a two-tier RAN with one MBS and a group of SBSs denoted by $\mathcal{K} = \{1, 2, \ldots, K\}$. The MBS is equipped with a cloudlet-server (CS) with a maximum computing-rate denoted by U_0 (in the following of this work, we use subscript "0" to denote the MBS). Each SBS k is equipped with an ES with a maximum computing-rate denoted by U_k. Meanwhile, there exist a group of the WDs denoted by $\mathcal{I} = \{1, 2, \ldots, I\}$, with each WD i having a total computation-workload S_i^{tot} to be completed. We use u_i^{loc} to denote WD i's local computing-rate. Thanks to DC, each WD can execute the dual computation offloading to the MBS and one of the selected SBSs. As illustrated before, each SBS uses NOMA to accommodate the associated WDs' offloading. Similar to

TABLE I
NOTATIONS

Notations	Definitions
Ω_k	SBS k's NOMA-group.
S_i^{tot}	WD i's workloads.
u_i^{loc}	WD i's local computing-rate.
W_k	SBS k's bandwidth.
W_0	The bandwidth for each WD in FDMA.
$G_{i,k}$	The channel gain from WD i to SBS k.
$p_{i,k}^{\text{req}}$	The required transmit-power for WD i uploading workloads to SBS k.
$P_i^{\max}$	The maximum transmit-power of WD i.
U_k	The maximum computing-rate of SBS k.
U_0	The maximum computing-rate of the MBS.
t_i^{ove}	The overall latency of completing WD i's task.
$s_{i,k}$	Offloading decision from WD i to SBS k.
$s_{i,0}$	Offloading decision from WD i to the MBS.
t_k^{up}	Transmission duration of SBS k's NOMA-group.
t_0^{up}	Transmission duration of the MBS's FDMA.
$u_{i,k}$	Allocated computing-rate from SBS k to WD i.
$u_{i,0}$	Allocated computing-rate from the MBS to WD i.

many existing studies [39], [42], we assume that two WDs form a NOMA-group, which is aligned with the 3GPP standard for mitigating the long decoding latency due to the successive interference cancellation (SIC). In addition, in order to provide a clear system model and problem formulation, and enable an effective algorithmic design, similar to [35], [36], we assume a perfect channel state information in the hybrid NOMA-FDMA transmission. The key notations used in this paper are provided in Table I.

For the sake of clear modeling, we use $\Omega_k = \{i, j\}$ to denote the set of two selected WDs which are associated with SBS k, namely, WD i and WD j form a NOMA-group to offload their respective workloads $s_{i,k} \in [0, S_i^{\text{tot}}]$ and $s_{j,k} \in [0, S_j^{\text{tot}}]$ to SBS k. Furthermore, assuming that SBS k decodes WD j's offloaded data prior to WD i's offloaded data,[2] then according to the Shannon's formula and SIC operations, we can express the offloading rate from WD j to SBS k as

$$r_{j,k} = W_k \log_2\left(1 + \frac{p_{j,k}G_{j,k}}{p_{i,k}G_{i,k} + N_k}\right), j \in \Omega_k, \tag{1}$$

and express the offloading rate from WD i to SBS k as

$$r_{i,k} = W_k \log_2\left(1 + \frac{p_{i,k}G_{i,k}}{N_k}\right), i \in \Omega_k, \tag{2}$$

where parameter W_k denotes the channel bandwidth of SBS k. We use $p_{i,k}$ and $p_{j,k}$ to denote WD i's and WD j's transmission powers. Parameter $G_{j,k}$ denotes the channel power gain from WD j to SBS k, and $G_{i,k}$ denotes the channel power gain from WD i to SBS k. Parameter N_k denotes the power of the background noise.

[2] It is noticed that the uplink NOMA can accommodate an arbitrary decoding order. Thus, our modelings and proposed algorithms are also applicable to the other decoding order by slightly changing the WDs' indices.

Furthermore, we use t_k^{up} to denote the transmission duration for this NOMA-group $\Omega_k = \{i, j\}$, which thus leads to the following two conditions

$$r_{i,k}t_k^{\text{up}} \geq s_{i,k}, i \in \Omega_k, \tag{3}$$

$$r_{j,k}t_k^{\text{up}} \geq s_{j,k}, j \in \Omega_k, \tag{4}$$

namely, WD i should complete sending its offloaded workloads $s_{i,k}$ with duration t_k^{up}, and so does WD j.

With (3) and (4), we can obtain the required minimum NOMA transmission powers of WD i and WD j as follows:

$$p_{i,k}^{\text{req}}\left(s_{i,k}, t_k^{\text{up}}\right) = \frac{N_k}{G_{i,k}}\left(2^{\frac{s_{i,k}}{W_k t_k^{\text{up}}}} - 1\right), i \in \Omega_k, \tag{5}$$

$$p_{j,k}^{\text{req}}(s_{i,k}, s_{j,k}, t_k^{\text{up}}) = \frac{N_k}{G_{j,k}} 2^{\frac{s_{i,k}}{W_k t_k^{\text{up}}}}\left(2^{\frac{s_{j,k}}{W_k t_k^{\text{up}}}} - 1\right), j \in \Omega_k. \tag{6}$$

In addition to offloading its workloads to its associated SBS, each WD i can also offload part of its workloads denoted by $s_{i,0} \in [0, S_i^{\text{tot}}]$ to the MBS. The MBS uses FDMA to accommodate different WDs' offloaded data, and we express the offloading rate from WD i to the MBS as

$$r_{i,0} = W_0 \log_2\left(1 + \frac{P_{i,0}G_{i,0}}{N_0}\right), \forall i \in \mathcal{I}, \tag{7}$$

where parameter W_0 denotes the orthogonal channel bandwidth for each WD. Parameter $P_{i,0}$ denotes WD i's transmission-power to the MBS, and $G_{i,0}$ denotes the channel gain from WD i to the MBS. In particular, we use t_0^{up} to denote all WDs' FDMA-transmission duration to the MBS. Since each WD i should complete sending its data $s_{i,0}$ with duration t_0^{up}, we have the following constraint

$$t_0^{\text{up}} r_{i,0} \geq s_{i,0}, \forall i \in \mathcal{I}. \tag{8}$$

The latency for completing WD i's workloads can be expressed as follows. First, WD i needs to locally process its remaining workloads $S_i^{\text{tot}} - s_{i,k} - s_{i,0}$, and the local processing time is $\frac{S_i^{\text{tot}} - s_{i,k} - s_{i,0}}{u_i^{\text{loc}}}$. Second, from WD i's perspective, its processing time at ES k is $\frac{s_{i,k}}{u_{i,k}}$, where $u_{i,k}$ is the computing-rate allocated by ES k for WD i (notice that $u_{i,k} = 0$ and $s_{i,k} = 0$ if WD i is not associated with SBS k). Third, the processing time at the CS is $\frac{s_{i,0}}{u_{i,0}}$, where $u_{i,0}$ is the computing-rate allocated by the CS for WD i. In summary, we can express the latency for completing WD i's workloads as

$$t_i^{\text{ove}} = \max\left\{\frac{S_i^{\text{tot}} - s_{i,k} - s_{i,0}}{u_i^{\text{loc}}}, t_k^{\text{up}} + \frac{s_{i,k}}{u_{i,k}}, t_0^{\text{up}} + \frac{s_{i,0}}{u_{i,0}}\right\}, \forall i \in \mathcal{I}. \tag{9}$$

B. Problem Formulation

We first consider that all WDs' NOMA-groupings $\Omega = \{\Omega_k\}_{k \in \mathcal{K}}$ are given and formulate an optimization problem to minimize the overall latency for completing all WDs' workloads, by jointly optimizing the offloading decisions $\mathbf{s} =$

$\{\{s_{i,k}\}_{\forall i\in\mathcal{I},k\in\mathcal{K}},\{s_{i,0}\}_{\forall i\in\mathcal{I}}\}$, the transmission duration $\mathbf{t}^{\mathrm{up}}=\{t_0^{\mathrm{up}},\{t_k^{\mathrm{up}}\}_{\forall k\in\mathcal{K}}\}$, and the computing-rate allocation $\mathbf{u}=\{\{u_{i,k}\}_{\forall i\in\mathcal{I},k\in\mathcal{K}},\{u_{i,0}\}_{\forall i\in\mathcal{I}}\}$. The formulated problem is shown as follows ("OLM" stands for "Overall Latency Minimization"):

$$(\mathrm{OLM}):\quad d_\Omega=\min_{\mathbf{s},\mathbf{t}^{\mathrm{up}},\mathbf{u}}\max_{\forall i\in\mathcal{I}}\{t_i^{\mathrm{ove}}\}$$

$$\text{Subject to}:\quad p_{i,k}^{\mathrm{req}}(s_{i,k},t_k^{\mathrm{up}})\le P_i^{\max},i\in\Omega_k, \tag{10}$$

$$p_{j,k}^{\mathrm{req}}(s_{i,k},s_{j,k},t_k^{\mathrm{up}})\le P_j^{\max},j\in\Omega_k, \tag{11}$$

$$W_0\log_2\left(1+\frac{P_{i,0}G_{i,0}}{N_0}\right)t_0^{\mathrm{up}}\ge s_{i,0},\forall i\in\mathcal{I}, \tag{12}$$

$$0\le s_{i,k}+s_{i,0}\le S_i^{\mathrm{tot}},\forall i\in\Omega_k, \tag{13}$$

$$\sum_{i\in\Omega_k}u_{i,k}\le U_k,\forall k\in\mathcal{K}, \tag{14}$$

$$\sum_{i\in\mathcal{I}}u_{i,0}\le U_0. \tag{15}$$

$$\text{Variables}:\quad \mathbf{s},\ \mathbf{t}^{\mathrm{up}}\text{and }\mathbf{u}.$$

Constraints (10) and (11) ensure that in a NOMA-group Ω_k (associated with SBS k), both WD i's and WD j's transmission powers cannot exceed their respective capacities denoted by $P_i^{\max}$ and $P_j^{\max}$. Constraint (12) ensures that the WDs offload their workloads $\{s_{i,0}\}_{i\in\mathcal{I}}$ to the MBS with duration t_0^{up}. Constraint (13) ensures that each WD i's offloaded workloads cannot exceed its total workloads. Constraint (14) ensures that the total allocated computation-resources by SBS k cannot exceed its capacity U_k. Constraint (15) ensures that the total allocated computing-rate by the MBS for the WDs cannot exceed its capacity U_0.

IV. Decomposition for Problem (OLM)

However, Problem (OLM) is a non-convex optimization problem, which is difficult to solve directly. To address this difficulty, we adopt an approach of decomposition, i.e., separating Problem (OLM) into a top-problem for optimizing the overall latency, and the consequent subproblem for optimizing the WDs' partial offloading decisions, the computing and communication resources allocation. The details are as follows.

A. Equivalent Transformation

Before decomposing Problem (OLM), we introduce a variable $d_\Omega^{\mathrm{sub}}=\max_{\forall i\in\mathcal{I}}\{t_i^{\mathrm{ove}}\}$ to denote the overall latency and rewrite the objective function of Problem (OLM) as minimizing d_Ω^{sub}

$$d_\Omega=\min_{\mathbf{s},\mathbf{t}^{\mathrm{up}},\mathbf{u}}d_\Omega^{\mathrm{sub}}. \tag{16}$$

Correspondingly, each item in (9) should not be greater than d_Ω^{sub}, which leads to the following constraints:

$$S_i^{\mathrm{tot}}-s_{i,k}-s_{i,0}\le u_i^{\mathrm{loc}}d_\Omega^{\mathrm{sub}},\forall i\in\Omega_k,k\in\mathcal{K}, \tag{17}$$

$$t_k^{\mathrm{up}}u_{i,k}+s_{i,k}\le u_{i,k}d_\Omega^{\mathrm{sub}},\forall i\in\Omega_k,k\in\mathcal{K}, \tag{18}$$

$$t_0^{\mathrm{up}}u_{i,0}+s_{i,0}\le u_{i,0}d_\Omega^{\mathrm{sub}},\forall i\in\mathcal{I}. \tag{19}$$

With (16)-(19), we can rewrite an equivalent form of Problem (OLM) as follows:

$$(\mathrm{OLM\text{-}E}):\quad d_\Omega=\min_{\mathbf{s},\mathbf{t}^{\mathrm{up}},\mathbf{u}}d_\Omega^{\mathrm{sub}},$$

$$\text{Subject to}:\quad \text{constraints: }(10)-(15),(17)-(19),$$

$$\text{Variables}:\quad d_\Omega^{\mathrm{sub}},\mathbf{s},\ \mathbf{t}^{\mathrm{up}},\ \text{and }\mathbf{u}.$$

A keen observation is that Problem (OLM) aims at finding the minimum feasible d_Ω^{sub} subject to constraints $(10)-(15)$ and $(17)-(19)$. Therefore, assuming that d_Ω^{sub} is given in advance, we check whether the above constraints can yield a non-empty region under the given d_Ω^{sub}. If the region is non-empty, we can decrease the current value of d_Ω^{sub} in further. Otherwise, the currently given d_Ω^{sub} is infeasible, and we need to increase d_Ω^{sub}. Based on this rationale, we decompose Problem (OLM-E) into two subproblems in the next subsections.

B. Subproblem Under a Given d_Ω^{sub}

Assuming d_Ω^{sub} is given in advance, we aim at finding the minimum total computation-resources required from the CS. The problem is expressed in the following Problem (OLM-E-Sub) with U_Ω^{req} denoting the optimal value.

$$(\mathrm{OLM\text{-}E\text{-}Sub}):\quad U_\Omega^{\mathrm{req}}=\min_{\mathbf{s},\mathbf{t}_{\mathrm{k}}^{\mathrm{up}},\mathbf{u}}\sum_{i\in\mathcal{I}}u_{i,0}$$

$$\text{Subject to}:\quad \text{constraints: }(10)-(14),(17)-(19),$$

$$\text{Variables}:\quad \mathbf{s},\ \mathbf{t}_{\mathrm{k}}^{\mathrm{up}},\ \text{and }\mathbf{u},$$

where vector $\mathbf{t}_{\mathrm{k}}^{\mathrm{up}}=\{t_k^{\mathrm{up}}\}_{\forall k\in\mathcal{K}}$ (i.e., $\mathbf{t}_{\mathrm{k}}^{\mathrm{up}}=\mathbf{t}^{\mathrm{up}}\setminus t_0^{\mathrm{up}}$). Note that we assume that t_0^{up} is also given in Problem (OLM-E-Sub) for facilitating our following algorithm design, which will be explained by Proposition 3 in Section V-A4. It can be observed that constraint (15) in Problem (OLM-E) is transformed into the objective function of Problem (OLM-E-Sub) (i.e., the minimum total computation-resources U_Ω^{req} required at the CS under the given d_Ω^{sub}). The other constraints remain unchanged.

C. Top-Problem for Optimizing d_Ω^{sub}

In the top-problem, we aim at finding the minimum d_Ω^{sub} and the corresponding t_0^{up} such that the U_Ω^{req} in Problem (OLM-E-Sub) is not greater than U_0. Thus, the top-problem can be expressed as follows:

$$\begin{aligned}(\mathrm{OLM\text{-}E\text{-}Top}):&\quad d_\Omega=\min d_\Omega^{\mathrm{sub}}\\ \text{Subject to}:&\quad U_\Omega^{\mathrm{req}}\le U_0,\\ \text{Variables}:&\quad d_\Omega^{\mathrm{sub}}\text{ and }t_0^{\mathrm{up}}.\end{aligned} \tag{20}$$

If the minimum U_Ω^{req} satisfies constraint (15), i.e., $U_\Omega^{\mathrm{req}}\le U_0$, the currently given d_Ω^{sub} is feasible for Problem (OLM-E). Otherwise, the currently given d_Ω^{sub} is infeasible for Problem (OLM-E).

The above decomposition approach has the following properties.

Proposition 1: The result of Problem (OLM-E-Sub) (i.e., U_Ω^{req}) is non-increasing with respect to d_Ω^{sub}.

Proof: Problem (OLM-E-Sub) is to minimize the required total computation-resources from the CS under a given d_Ω^{sub}. Note that we define $d_\Omega^{\text{sub}} = \max_{i\in\mathcal{I}}\{t_i^{\text{ove}}\}$, where t_i^{ove} is defined in (9). While the currently given d_Ω^{sub} is less than the value of $\max_{i\in\mathcal{I}}\{t_i^{\text{ove}}\}$, there exist some WDs which cannot complete their workloads within the currently given d_Ω^{sub}. In other words, if these WDs' workloads have to be completed within the given d_Ω^{sub}, more computation-resources are required. Since the total allocated computation-resources from the CS is unconstrained in Problem (OLM-E-Sub), U_Ω^{req} will increase. On the other hand, if the currently given d_Ω^{sub} is greater than the value of $\max_{i\in\mathcal{I}}\{t_i^{\text{ove}}\}$, it means that the latency is loose and the computation-resources at the CS can be reduced. Since Problem (OLM-E-Sub) is to minimize the total computation-resources required at the CS, U_Ω^{req} thus decreases. Therefore, U_Ω^{req} is non-increasing with respect to d_Ω^{sub}. ■

Proposition 2: U_Ω^{req} approaches the value of U_0 under the minimum d_Ω^{sub}.

Proof: According to Proposition 1, there exist two points $\overline{d}_\Omega^{\text{sub}}$ and $\underline{d}_\Omega^{\text{sub}}$, and $\overline{d}_\Omega^{\text{sub}} > \underline{d}_\Omega^{\text{sub}}$, such that $U_\Omega^{\text{req}} < U_0$ under $\overline{d}_\Omega^{\text{sub}}$ and $U_\Omega^{\text{req}} > U_0$ under $\underline{d}_\Omega^{\text{sub}}$. Moreover, there must be $\tilde{d}_\Omega^{\text{sub}} \in \left[\underline{d}_\Omega^{\text{sub}}, \overline{d}_\Omega^{\text{sub}}\right]$ such that U_Ω^{req} approximates to U_0 under $\tilde{d}_\Omega^{\text{sub}}$. If $\tilde{d}_\Omega^{\text{sub}}$ is not the minimum and we decrease it further, U_Ω^{req} will be greater than U_0 and become infeasible for Problem (OLM). Thus, $\tilde{d}_\Omega^{\text{sub}}$ is the minimum. Meanwhile, U_Ω^{req} approaches the value of U_0 under $\tilde{d}_\Omega^{\text{sub}}$. ■

As a summary of the above decomposition approach, Problem (OLM-E) is decomposed into two subproblems, i.e., Problem (OLM-E-Sub) and Problem (OLM-E-Top). Problem (OLM-E-Sub) checks the feasibility of a given d_Ω^{sub} by jointly optimizing the WDs' dual offloading decisions, NOMA-groups' transmission duration, and computation-resources allocation. Problem (OLM-E-Top) finds the minimum feasible d_Ω^{sub} and the corresponding t_0^{up} such that the result of Problem (OLM-E-Sub) is not greater than U_0.

V. Proposed Algorithms for Problem (OLM-E-Sub) and Problem (OLM-E-Top)

A. Cell-Based Distributed Decision (CDD) Algorithm for Problem (OLM-E-Sub)

1) Variable Substitution: With constraint (19), we can obtain the lower bound of $u_{i,0}$ as follows:

$$u_{i,0} \geq \frac{s_{i,0}}{d_\Omega^{\text{sub}} - t_0^{\text{up}}}, \forall i \in \mathcal{I}. \tag{21}$$

Since Problem (OLM-E-Sub) is to minimize the sum of $\{u_{i,0}\}_{\forall i\in\mathcal{I}}$, replacing $u_{i,0}$ with its lower bound will not affect the result of Problem (OLM-E-Sub). Thus, the equivalent form of the objective function of Problem (OLM-E-Sub) is

$$\min_{\mathbf{s},\mathbf{t}_{\text{k}}^{\text{up}},\mathbf{u}} \sum_{i\in\mathcal{I}} u_{i,0} = \min_{\mathbf{s},\mathbf{t}_{\text{k}}^{\text{up}},\mathbf{u}} \sum_{i\in\mathcal{I}} \frac{s_{i,0}}{d_\Omega^{\text{sub}} - t_0^{\text{up}}}. \tag{22}$$

Similarly, with constraint (18), the lower bound of $u_{i,k}$ is

$$u_{i,k} \geq \frac{s_{i,k}}{d_\Omega^{\text{sub}} - t_k^{\text{up}}}, i \in \Omega_k. \tag{23}$$

With (23), constraint (14) can be transformed into (24), i.e.,

$$s_{i,k} + s_{j,k} \leq U_k(d_\Omega^{\text{sub}} - t_k^{\text{up}}), i, j \in \Omega_k. \tag{24}$$

The above transformations help us reduce the number of variables in Problem (OLM-E-Sub) without causing any changes to the optimal solution for Problem (OLM-E-Sub). Moreover, the non-linear coupling between t_k^{up} and $u_{i,k}$ in constraint (18) can also be decomposed.

2) Decoupling Offloading Decisions in the Uplink NOMA: A challenge for solving Problem (OLM-E-Sub) is the coupling effect among the WDs' offloading decisions according to constraint (11), which is non-linear and non-convex. The expression of (11) is shown as follows:

$$2^{\frac{s_{i,k}+s_{j,k}}{t_k^{\text{up}}W_k}} \leq \frac{P_j^{\max}G_{j,k}}{N_k} + 2^{\frac{s_{i,k}}{t_k^{\text{up}}W_k}}. \tag{25}$$

To address this difficulty, we introduce a group of auxiliary variables $\{\beta_k\}_{k\in\mathcal{K}}$

$$\beta_k = \frac{P_j^{\max}G_{j,k}}{N_k} + 2^{\frac{s_{i,k}}{t_k^{\text{up}}W_k}}, i, j \in \Omega_k, k \in \mathcal{K}. \tag{26}$$

Thus, (11) can be transformed to

$$2^{\frac{s_{i,k}+s_{j,k}}{t_k^{\text{up}}W_k}} \leq \beta_k, i, j \in \Omega_k, k \in \mathcal{K}. \tag{27}$$

Meanwhile, with (10), we can obtain

$$2^{\frac{s_{i,k}}{t_k^{\text{up}}W_k}} \leq 1 + \frac{P_i^{\max}G_{i,k}}{N_k}. \tag{28}$$

While $s_{i,k} = 0$, β_k equals to $1 + \frac{P_j^{\max}G_{j,k}}{N_k}$ according to (26), i.e., the lower bound of β_k. While the equation of (28) is met, β_k equals to $\frac{P_j^{\max}G_{j,k}+P_i^{\max}G_{i,k}}{N_k} + 1$ according to the sum of (26) and (28), i.e., the upper bound of β_k. Thus, with (26) and (28), we can obtain the range of β_k

$$\beta_k \in \left[1 + \frac{P_j^{\max}G_{j,k}}{N_k}, \frac{P_j^{\max}G_{j,k} + P_i^{\max}G_{i,k}}{N_k} + 1\right], \forall k \in \mathcal{K}. \tag{29}$$

3) Distributed Form of Problem (OLM-E-Sub): After the above transformations, Problem (OLM-E-Sub) can be rewritten into Problem (Sub-β) as follows:

$$
\begin{aligned}
(\text{Sub-}\ \beta):\quad & U_{\Omega}^{\text{req}} = \min \sum_{i\in\mathcal{I}} \frac{s_{i,0}}{d_{\Omega}^{\text{sub}} - t_0^{\text{up}}} \\
\text{Subject to}:\quad & 2^{\frac{s_{i,k}}{t_k^{\text{up}} W_k}} \le 1 + \frac{P_i^{\max} G_{i,k}}{N_k}, i \in \Omega_k, \forall k \in \mathcal{K}, \\
& 2^{\frac{s_{i,k}+s_{j,k}}{t_k^{\text{up}} W_k}} \le \beta_k, i, j \in \Omega_k, \forall k \in \mathcal{K}, \\
& W_0 \log_2\left(1+\frac{P_{i,0}G_{i,0}}{N_0}\right) t_0^{\text{up}} \ge s_{i,0}, \forall i \in \mathcal{I}, \\
& 0 \le s_{i,k} + s_{i,0} \le S_i^{\text{tot}}, i \in \Omega_k, \forall k \in \mathcal{K}, \\
& S_i^{\text{tot}} - s_{i,k} - s_{i,0} \le u_i^{\text{loc}} d_{\Omega}^{\text{sub}}, i \in \Omega_k, \forall k \in \mathcal{K}, \\
& s_{i,k} + s_{j,k} \le U_k (d_{\Omega}^{\text{sub}} - t_k^{\text{up}}), i, j \in \Omega_k, \forall k \in \mathcal{K}, \\
& \beta_k \in \left[1+\frac{P_j^{\max} G_{j,k}}{N_k}, \frac{P_j^{\max} G_{j,k} + P_i^{\max} G_{i,k}}{N_k} + 1\right], \\
\text{Variables}:\quad & \mathbf{s},\ \mathbf{t}_{\mathbf{k}}^{\text{up}},\ \text{and}\ \{\beta_k\}_{k\in\mathcal{K}}.
\end{aligned}
$$

Problem (Sub-β) can be divided into K optimization problems for individual cells in which each cell individually optimizes its own decision variables including the associated WDs' partial offloading decisions, NOMA transmission and β_k. This problem can be expressed as Problem (sub-β-cell) bellow.

$$
\begin{aligned}
(\text{Sub-}\ \beta\text{-cell}):\ & U_{\Omega_k}^{\text{req}} = \min \frac{s_{i,0} + s_{j,0}}{d_{\Omega}^{\text{sub}} - t_0^{\text{up}}} \\
\text{Subject to}:\ & 2^{\frac{s_{i,k}}{t_k^{\text{up}} W_k}} \le 1 + \frac{P_i^{\max} G_{i,k}}{N_k}, i \in \Omega_k, \\
& 2^{\frac{s_{i,k}+s_{j,k}}{t_k^{\text{up}} W_k}} \le \beta_k, i, j \in \Omega_k, \\
& W_0 \log_2\left(1+\frac{P_{i,0}G_{i,0}}{N_0}\right) t_0^{\text{up}} \ge s_{i,0}, \forall i \in \Omega_k, \\
& 0 \le s_{i,k} + s_{i,0} \le S_i^{\text{tot}}, \forall i \in \Omega_k, \\
& S_i^{\text{tot}} - s_{i,k} - s_{i,0} \le u_i^{\text{loc}} d_{\Omega}^{\text{sub}}, \forall i \in \Omega_k, \\
& s_{i,k} + s_{j,k} \le U_k (d_{\Omega}^{\text{sub}} - t_k^{\text{up}}), \forall i \in \Omega_k, \\
& \beta_k \in \left[1 + \frac{P_j^{\max} G_{j,k}}{N_k}, \frac{P_j^{\max} G_{j,k} + P_i^{\max} G_{i,k}}{N_k} + 1\right], \\
\text{Variables}:\ & s_{i,0}, s_{i,k}, s_{j,0}, s_{j,k}, t_k^{\text{up}}, \text{and}\ \ \beta_k.
\end{aligned}
$$

By solving Problem (Sub-β-cell) for each SBS, we can obtain the result of Problem (Sub-β) as

$$
U_{\Omega}^{\text{req}} = \sum_{k\in\mathcal{K}} \sum_{i\in\Omega_k} \frac{s_{i,0}}{d_{\Omega}^{\text{sub}} - t_0^{\text{up}}} = \sum_{k\in\mathcal{K}} U_{\Omega_k}^{\text{req}}. \tag{30}
$$

Notice that, since Problem (Sub-β-cell) is independent from each other, we can solve Problem (Sub-β-cell) for each SBS k in parallel.

4) Cell-Based Distributed Decision: To solve Problem (OLM-E-Sub), we identify the following significant feature.

Proposition 3: Under the given $\{\beta_k, d_{\Omega}^{\text{sub}}, t_0^{\text{up}}\}$, Problem (Sub-$\beta$-cell) is a strictly convex problem with respect to variables $\{t_k^{\text{up}}, s_{i,k}, s_{i,0}, s_{j,k}, s_{j,0}\}$.

Algorithm 1: Cell-based Distributed Decision Algorithm (i.e., CDD algorithm) for Solving Problem (OLM-E-Sub).

1: **Input:** $d_{\Omega}^{\text{sub}}, t_0^{\text{up}}$.
2: **Initialize:** β_k's upper bound $\beta_k^{\text{ub}} = \frac{P_i^{\max} G_{i,k} + P_j^{\max} G_{j,k}}{N_0} + 1$, β_k's lower bound $\beta_k^{\text{lb}} = \frac{P_j^{\max} G_{j,k}}{N_0} + 1$.
3: Generate Problem (Sub-β-cell) for each cell based on Problem (Sub-β) and Solve Problem (Sub-β-cell) by executing the following steps.
4: Update $\beta_k = \beta_k^{\text{ub}}$.
5: **while** $\beta_k \ge \beta_k^{\text{lb}}$ **do**
6: Use interior point method to solve Problem (Sub-β-cell) under β_k and obtain the optimal solutions $\{s_{i,k}^*, s_{j,k}^*, s_{i,0}^*, s_{j,0}^*, t_k^{\text{up}*}\} = \arg\min \frac{s_{i,0}+s_{j,0}}{d_{\Omega}^{\text{sub}} - t_0^{\text{up}}}$.
7: **if** $N_k 2^{\frac{s_{i,k}^*}{W_k t_k^{\text{up}*}}} (2^{\frac{s_{j,k}^*}{W_k t_k^{\text{up}*}}} - 1) \le P_j^{\max} G_{j,k}$ **then**
8: $\{s_{i,k}^*, s_{j,k}^*, s_{i,0}^*, s_{j,0}^*, t_k^{\text{up}*}\}$ is also the optimal solution for Problem (OLM-E-Sub) according to Proposition 4, and compute $U_{\Omega_k}^{\text{req}} = \frac{s_{i,0}^* + s_{j,0}^*}{d_{\Omega}^{\text{sub}} - t_0^{\text{up}}}$.
9: Break.
10: **else**
11: Update $\beta_k \leftarrow \beta_k - \Delta\beta_k$, where $\Delta\beta_k$ is the step size.
12: **end if**
13: **end while**
14: Compute the result of Problem (OLM-E-Sub): $U_{\Omega}^{\text{req}} = \sum_{k\in\mathcal{K}} U_{\Omega_k}^{\text{req}}$.
15: **Output:** U_{Ω}^{req}.

Proof: While d_{Ω}^{sub} and t_0^{up} are given, the objective function and constraints of Problem (Sub-β-cell) are convex [43] except constraint (27). If we enumerate β_k in the feasible region according to (29), constraint (27) becomes affine. Thus, Problem (Sub-β-cell) is a convex optimization problem which can be solved by the convex optimization algorithms under the given β_k, d_{Ω}^{sub}, and t_0^{up}. ■

Exploiting Proposition 3, we propose the cell-based distributed decision (CDD) algorithm to obtain the optimal solution of Problem (Sub-β) by solving Problem (Sub-β-cell) for each SBS in parallel, and further check the feasibility of the obtained solution for Problem (OLM-E-Sub) according to Proposition 4. The pseudo code of our proposed CDD algorithm is presented in Algorithm 1. The key steps are explained as follows.

- (*Step 5 to Step 6*): In Problem (Sub-β-cell) and WDs $i, j \in \Omega_k$, we first enumerate β_k linearly in descending order at the top level. Then, we use the Interior Point Method (IPM) [44] to find the correspondingly optimal solution under the given β_k, which is denoted as

$$
\left\{s_{i,k}^*, s_{j,k}^*, s_{i,0}^*, s_{j,0}^*, t_k^{\text{up}*}\right\} = \arg\min U_{\Omega_k}^{\text{req}}. \tag{31}
$$

- (*Step 7 to Step 13*): Exploiting Proposition 4, we check the feasibility of $\{s_{i,k}^*, s_{j,k}^*, s_{i,0}^*, s_{j,0}^*, t_k^{\text{up}*}\}$ for Problem (OLM-E-Sub) according to the following condition

$$N_k 2^{\frac{s^*_{i,k}}{W_k t_k^{\text{up}*}}} \left(2^{\frac{s^*_{j,k}}{W_k t_k^{\text{up}*}}} - 1 \right) \leq P_j^{\max} G_{j,k}. \tag{32}$$

If condition (32) is met, $U_{\Omega_k}^{\text{req}} = \frac{s^*_{i,0}+s^*_{j,0}}{d_\Omega^{\text{sub}} - t_0^{\text{up}}}, i, j \in \Omega_k$. Otherwise, we should decrease β_k in further and return to Step 5 until condition (32) holds.

- (*Step 14*): While the optimal solutions of all SBSs satisfy condition (32), Problem (OLM-E-Sub) is solved. Thus, we obtain the result of Problem (OLM-E-Sub) $U_\Omega^{\text{req}} = \sum_{k \in \mathcal{K}} U_{\Omega_k}^{\text{req}}$.

We identify the following proposition regarding the relationship between Problem (OLM-E-Sub) and Problem (Sub-β).

Proposition 4: If the optimal solution for Problem (Sub-β) is feasible for Problem (OLM-E-Sub), this solution is also the optimal solution for Problem (OLM-E-Sub).

Proof: We discuss an extreme case of β_k to prove this proposition, since the key difference between Problem (OLM-E-Sub) and Problem (Sub-β) is that constraint (11) is transformed into constraint (27) by introducing variable β_k. In particular, supposing that we set β_k as its upper bound, we can obtain the correspondingly optimal $t_k^{\text{up}*}$, $s^*_{i,k}$, and $s^*_{j,k}$ under the given β_k for Problem (Sub-β). However, while the equality in constraint (28) does not hold, i.e., $s^*_{i,k} < t_k^{\text{up}*} W_k \log_2(1 + \frac{P_i^{\max} G_{i,k}}{N_k})$, it means that the feasible region of Problem (OLM-E-Sub) is relaxed by introducing β_k. Note that β_k substitutes the right part of constraint (25). Accordingly, $t_k^{\text{up}*}$, $s^*_{i,k}$ and $s^*_{j,k}$ may lie outside the feasible region of Problem (OLM-E-Sub). Thus, while $t_k^{\text{up}*}$, $s^*_{i,k}$ and $s^*_{j,k}$ are within the feasible region of Problem (OLM-E-Sub), the optimal solution for Problem (Sub-β) is also the optimal solution for Problem (OLM-E-Sub). To check whether the optimal solution for Problem (Sub-β) is feasible for Problem (OLM-E-Sub), we can input this optimal solution to constraint (11). If constraint (11) is met, this optimal solution is the optimal solution for Problem (OLM-E-Sub). Otherwise, we decrease the value of β_k until constraint (11) is met. ■

Now, we solve Problem (OLM-E-Sub) by our proposed CDD algorithm and obtain the required minimum computation-resources U_Ω^{req} from CS under the given d_Ω^{sub} and t_0^{up}. Next, we propose a two-layer hybrid search algorithm to optimize d_Ω^{sub} and t_0^{up} based on the result of Problem (OLM-E-Sub).

B. Two-Layer Hybrid Search (TLHS) Algorithm for Solving Problem (OLM-E-Top)

The goal of Problem (OLM-E-Top) is to optimize two variables, i.e., d_Ω^{sub} and t_0^{up}, to find the minimum d_Ω^{sub} and the corresponding t_0^{up}. We propose a two-layer hybrid search (TLHS) algorithm consisting of a bisection search on d_Ω^{sub} in the top layer and a linear search on t_0^{up} in the bottom layer. The pseudo code of our proposed TLHS algorithm is presented in Algorithm 2 above, whose details are explained as follows.

- In the top layer, we adopt bisection search on d_Ω^{sub} based on Proposition 1. In particular, the non-increasing relationship of U_Ω^{req} with respect to d_Ω^{sub} provides an idea for optimizing d_Ω^{sub}. While U_Ω^{req} exceeds the CS's computation-resources U_0 under a given d_Ω^{sub}, it means that the given d_Ω^{sub} is infeasible for Problem (OLM). Then, we can increase the value of d_Ω^{sub} such that U_Ω^{req} will be reduced. Similarly, if U_Ω^{req} exceeds the CS's computation-resources U_0 under a given d_Ω^{sub}, we can decrease d_Ω^{sub} such that U_Ω^{req} will increase. Thus, exploiting the non-increasing relationship between U_Ω^{req} and d_Ω^{sub}, we can determine to decrease or increase d_Ω^{sub} according to whether U_Ω^{req} exceeds the CS's computation-resources U_0 or not under the given U_Ω^{req}.
- In the bottom layer, under a given d_Ω^{sub}, we use a linear search to enumerate t_0^{up} and compute the corresponding U_Ω^{req} until $U_\Omega^{\text{req}} \leq U_0$. Then, we break the step of linear search and reduce d_Ω^{sub}. If there is no t_0^{up} that yields $U_\Omega^{\text{req}} \leq U_0$, it means that the current d_Ω^{sub} is infeasible for Problem (OLM) and we should increase d_Ω^{sub}. We repeat the above operations until the gap between d_Ω^{sub}'s upper and lower bounds is less than a predefined threshold. In this case, the current d_Ω^{sub} reaches the minimum.

Algorithm 2: Two-layer Hybrid Search Algorithm (i.e., TLHS algorithm) for Solving Problem (OLM-E-Top).

Initialize: Initialize d_Ω^{sub}'s upper bound $d_\Omega^{\text{ub}} = \max\{S_i^{\text{tot}} / u_i^{\text{loc}}\}_{\forall i \in \mathcal{I}}$, d_Ω^{sub}'s lower bound $d_\Omega^{\text{lb}} = 0$, t_0^{up}'s lower bound $t_0^{\text{lb}} = 0$.
while $d_\Omega^{\text{lb}} < d_\Omega^{\text{ub}}$ **do**
  Update $d_\Omega^{\text{sub}} \leftarrow \frac{1}{2}(d_\Omega^{\text{lb}} + d_\Omega^{\text{ub}})$.
  Update t_0^{up}'s upper bound: $t_0^{\text{ub}} = d_\Omega^{\text{sub}}$.
  Update $t_0^{\text{up}} = t_0^{\text{lb}}$.
  while $t_0^{\text{up}} \leq t_0^{\text{ub}}$ **do**
    Solve Problem (OLM-E-Sub) by Algorithm 1 to obtain U_Ω^{req}.
    if $U_\Omega^{\text{req}} \leq U_0$ **then**
      Current d_Ω^{sub} and t_0^{up} are feasible for Problem (OLM).
      Update the lower bound of t_0^{up}, i.e., $t_0^{\text{lb}} = t_0^{\text{up}}$.
      Break Linear Search Loop.
    else
      Update $t_0^{\text{up}} \leftarrow t_0^{\text{up}} + \Delta t_0^{\text{up}}$, where Δt_0^{up} is the step size.
    end if
  end while
  if Finding the feasible t_0^{up} (i.e., $U_\Omega^{\text{req}} \leq U_0$) **then**
    Update d_Ω^{sub}'s upper bound, i.e., $d_\Omega^{\text{ub}} = d_\Omega^{\text{sub}}$.
  else
    Update d_Ω^{sub}'s lower bound, i.e., $d_\Omega^{\text{lb}} = d_\Omega^{\text{sub}}$.
  end if
end while
Find the minimum d_Ω^{sub}, then $d_\Omega = d_\Omega^{\text{sub}}$.
Output: $d_\Omega, t_0^{\text{up}}$.

Until now, we have obtained the optimal solution of Problem (OLM) by solving Problem (OLM-E-Sub) and Problem (OLM-E-Top).

C. The Optimality and Complexity Analysis of the Proposed Algorithms for Problem (OLM)

To address the formulated non-convex optimization problem, i.e., Problem (OLM), we equivalently decompose it into

two subproblems, i.e., Problem (OLM-E-Sub) and Problem (OLM-E-Top). To solve Problem (OLM-E-Sub), we propose the CDD algorithm to transform it into K optimization problems by introducing a group of auxiliary variables $\{\beta_k\}_{k\in\mathcal{K}}$. The optimization problem for each individual cell, i.e., Problem (Sub-β-cell), determines its own offloading decision and resource allocation. According to Proposition 3, Problem (Sub-β-cell) is proved to be a convex optimization problem that can be solved by many existing methods, e.g., the interior point method. In addition, we discuss the equivalence between Problem (Sub-β) and Problem (OLM-E-Sub) in Proposition 4. Specifically, if the optimal solution of Problem (Sub-β) is feasible to Problem (OLM-E-Sub), this solution is also the optimal solution of Problem (OLM-E-Sub). The proposed CDD algorithm uses a linear search to find β_k until the optimal solution of Problem (Sub-β) is feasible to Problem (OLM-E-Sub). Thus, the proposed CDD algorithm can find the optimal solution of Problem (OLM-E-Sub). The complexity of the CDD algorithm is linear (from step 5 to step 13 in Algorithm 1), depending on the searching step-size $\Delta\beta_k$ and the gap between the upper bound β_k^{ub} and the lower bound β_k^{lb} of β_k. β_k^{ub} and β_k^{lb} are given in equation (29). Thus, the complexity of the proposed CDD algorithm is $\mathcal{O}\left(\frac{\beta_k^{\mathrm{ub}}-\beta_k^{\mathrm{lb}}}{\Delta\beta_k}\right)$.

To solve Problem (OLM-E-Top), by exploiting the non-increasing relationship between the result of Problem (OLM-E-Sub) and d_Ω^{sub}, the proposed TLHS algorithm uses a bisection search algorithm to find the minimum d_Ω^{sub} according to the feasibility of Problem (OLM-E-Sub) with d_Ω^{sub}. Meanwhile, under a given d_Ω^{sub}, the TLHS uses a linear search to find a feasible t_0^{up} such that Problem (OLM-E-Sub) has feasible solutions. Thus, the proposed Algorithm 2 can find the minimum d_Ω^{sub}. In addition, the complexity of the bisection search is $\mathcal{O}\left(\log_2\left(\frac{d_\Omega^{\mathrm{ub}}-d_\Omega^{\mathrm{lb}}}{\varepsilon}\right)\right)$, where d_Ω^{ub} and d_Ω^{lb} are the upper bound and the lower bound of d_Ω^{sub}, respectively. ε is the preset stopping threshold of bisection search. The complexity of the linear search is $\mathcal{O}\left(\frac{t_0^{\mathrm{ub}}-t_0^{\mathrm{lb}}}{\Delta t_0}\right)$, where t_0^{ub} is the upper bound of t_0^{up} and t_0^{lb} is the lower bound of t_0^{up}. Δt_0 is the searching step-size of linear search. Thus, the complexity of Algorithm 2 is $\mathcal{O}\left(\frac{t_0^{\mathrm{ub}}-t_0^{\mathrm{lb}}}{\Delta t_0}\log_2\left(\frac{d_\Omega^{\mathrm{ub}}-d_\Omega^{\mathrm{lb}}}{\varepsilon}\right)\right)$.

In summary, the proposed method (i.e., the combination of the CDD algorithm and the TLHS algorithm) can find the optimal solution of Problem (OLM), and the complexity of solving Problem (OLM) is the product of the complexity of the CDD algorithm and the TLHS algorithm, i.e., $\mathcal{O}\left(\frac{(t_0^{\mathrm{ub}}-t_0^{\mathrm{lb}})(\beta_k^{\mathrm{ub}}-\beta_k^{\mathrm{lb}})}{\Delta t_0\Delta\beta_k}\log_2\left(\frac{d_\Omega^{\mathrm{ub}}-d_\Omega^{\mathrm{lb}}}{\varepsilon}\right)\right)$.

VI. Numerical Results for Optimal Offloading Under the Given WDs-Groupings

This section presents numerical results to validate our proposed CDD algorithm and TLHS algorithm for solving Problem (OLM). To this end, we consider a scenario of one MBS and a group of SBSs (the detailed number of the SBSs will be specified in the corresponding illustrations). Since our proposed algorithms are applicable to arbitrary WDs-groupings, we use a fixed WD-grouping scheme as $\Omega_k = \{2k-1, 2k\}, \forall k \in \mathcal{K}$. To evaluate the performance of our proposed algorithms, we adopt the parameter-settings according to [35]. In particular, the MBS's channel frequency is set as $W_0 = 10$ Mbps, and each SBS's channel frequency is set as $W_k = 3$ Mbps, $\forall k \in \mathcal{K}$. The maximum transmit-power of each WD is set as 0.5 W. The power of the background noise is $N_0 = 10^{-10}$. All the results are obtained with a PC of Intel (R) Core(TM) i7-9700 K CPU@3.6 GHz.

A. The Results of Problem (OLM)

Fig. 2 shows the results of our proposed CDD algorithm for solving Problem (OLM-E-Sub) under different values of d_Ω^{sub}. In particular, we test the CDD algorithm on three scenarios, i.e., i) $K = 3, I = 6$, and $U_0 = 10$ in Subplot 2(a), ii) $K = 4, I = 8$, and $U_0 = 40$ in Subplot 2(b), and iii) $K = 5, I = 10$, and $U_0 = 60$ in Subplot 2(c). In each subplot, we vary d_Ω^{sub}, and present the corresponding result ($\{U_{\Omega_k}^{\mathrm{req}}\}_{k\in\mathcal{K}}$) of Problem (Sub-$\beta$-cell) for each cell and the consequent result (U_Ω^{req}) of Problem (OLM-E-Sub). Notice that $U_{\Omega_k}^{\mathrm{req}}$ is the required CS computing-rate of each individual SBS k's NOMA-group, and U_Ω^{req} is the total required CS computing-rate of all SBSs' NOMA-groups, i.e., $U_\Omega^{\mathrm{req}} = \sum_{k\in\mathcal{K}} U_{\Omega_k}^{\mathrm{req}}$ (which is the result of Problem (OLM-E-Sub)) according to eq. (30) before. For the clear illustration of the rationale behind of our proposed algorithms, we also mark the value of U_0 (which is the horizontal line) and the intersection point where $U_\Omega^{\mathrm{req}} = U_0$ holds. It can be observed from the three subplots in Fig. 2 that U_Ω^{req} decreases as d_Ω^{sub} increases, which verifies our Proposition 1. Moreover, Proposition 2 proves that d_Ω^{sub} is the minimum while U_Ω^{req} approaching U_0. Exploiting Proposition 2, we can confirm the optimal solution of Problem (OLM-E-Sub) by the value of U_Ω^{req}. Thus, the intersection points between U_0 and U_Ω^{req} in subplots are the optimal solutions of different scenarios, which validates the effectiveness of our proposed CDD algorithm under all the tested cases.

With the same parameter-settings in the three subplots in Fig. 2, Fig. 3 further illustrates the rationale behind our TLHS algorithm for solving Problem (OLM). In each subplot in Fig. 3, we show the variations d_Ω^{sub} and U_Ω^{req} during the iterations of bisection search in the TLHS algorithm. In particular, it can be observed that U_Ω^{req} gradually converges to U_0, at which d_Ω^{sub} converges to its correspondingly minimum value. The results in Fig. 3 again verify Proposition 2, and validate the rationale of our proposed algorithms (i.e., the CDD algorithm and the TLHS algorithm) for solving Problem (OLM).

B. The Effectiveness of the Proposed Algorithms

Fig. 4 demonstrates the accuracy of our TLHS algorithm (with its subroutine CDD algorithm) for solving Problem (OLM). For the purpose of comparison, we also use LINGO's global-solver [23], block coordinate descent (BCD) [45], and heuristic equal-division (HED) to solve our problem. In particular, the

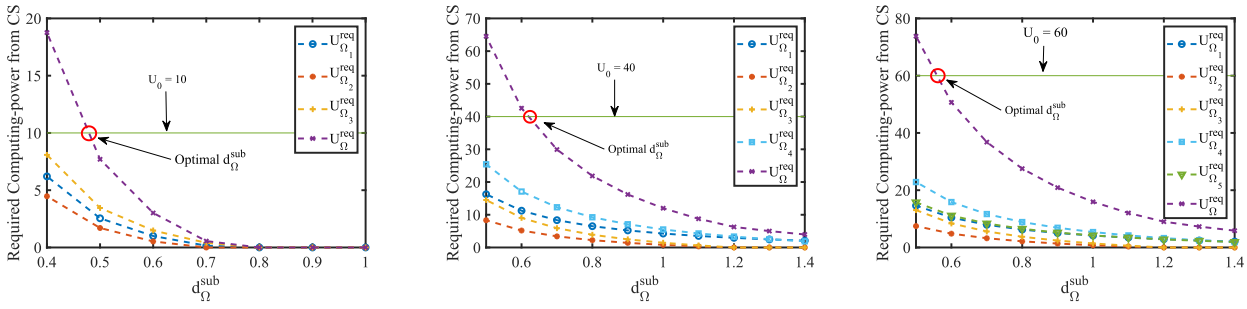


Fig. 2. Illustration of our CDD algorithm: optimal values of Problem (Sub-β-cell) and Problem (OLM-E-Sub) under different values of d_Ω^{sub}. (a)$K = 3, I = 6, U_0 = 10$. (b) $K = 4, I = 8, U_0 = 40$. (c) $K = 5, I = 10, U_0 = 60$.

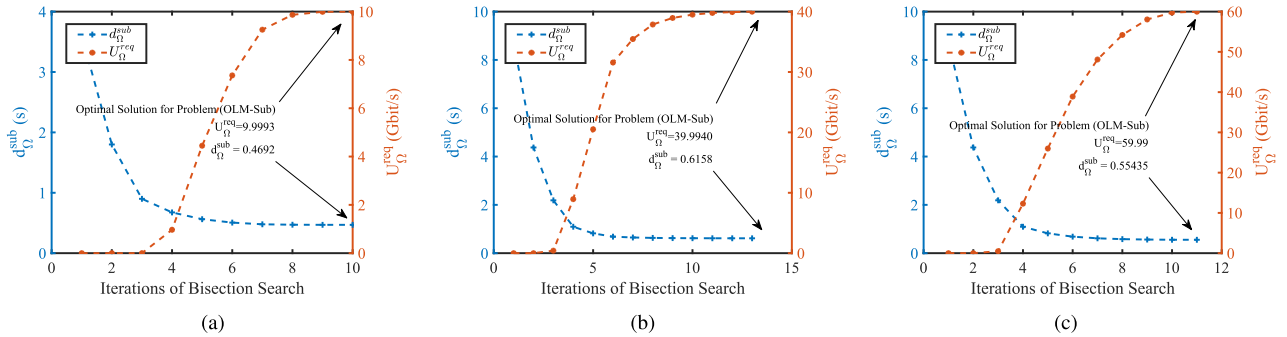


Fig. 3. Illustration of our TLHS algorithm: variations of U_Ω^{req} and d_Ω^{sub} during the iterations of the bisection search. (a) $K = 3, I = 6, U_0 = 10$. (b) $K = 4, I = 8, U_0 = 40$. (c) $K = 5, I = 10, U_0 = 60$.

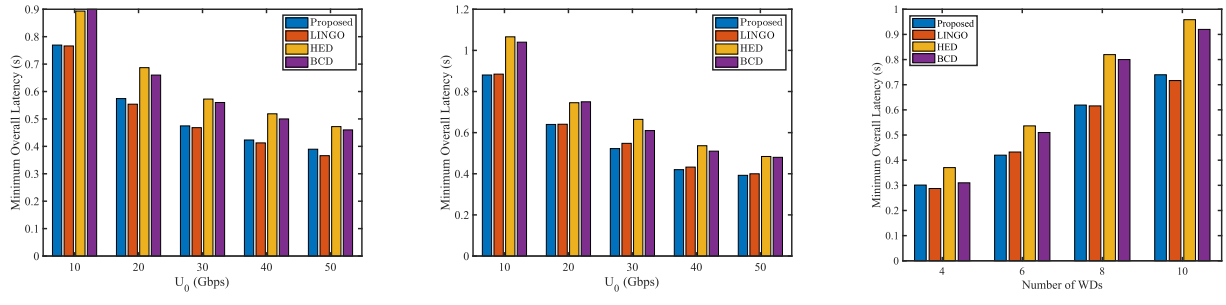


Fig. 4. Performance comparisons among our proposed algorithm, LINGO, HED, and BCD. (a) $\{U_1, U_2, U_3\} = \{6, 4, 5\}, K = 3, I = 6$. (b) $\{U_1, U_2, U_3\} = \{5, 2, 3\}, K = 3, I = 6$. (c) $U_0 = 40$.

LINGO's global-solver overcomes the weakness of suboptimal through methods of range bounding (e.g., interval analysis and convex analysis) and range reduction techniques (e.g., linear programming and constraint propagation) within a branch-and-bound framework to find the global solutions to non-convex optimization models. The BCD method is an iterative algorithm, which can find the suboptimal solution for non-convex optimization. It sequentially minimizes the objective function in each block coordinate while the other coordinates are fixed. The HED is a heuristic scheme to achieve a suboptimal solution. In this scheme, the CS's total computation-resources U_0 are equally divided among all WDs, while all the other variables are optimized according to those in Problem (OLM).

Fig. 4 shows the minimum overall latency achieved by the above four algorithms under different parameter-settings. Fig. 4(a) and Fig. 4(b) show the comparison results under different values of U_0, with Fig. 4(a) using a 6-WDs scenario with $\{U_1, U_2, U_3\} = \{6, 4, 5\}$Gbps and Fig. 4(b) using a 6-WDs scenario but with $\{U_1, U_2, U_3\} = \{5, 2, 3\}$Gbps. Fig. 4(c) shows the comparison results under different numbers of the WDs, with $U_0 = 40$Gbps. The results in all subplots demonstrate that our proposed solution can achieve the approximately same latency as the optimal solution provided by LINGO's global solver. Meanwhile, the latency achieved by our proposed solution is less than the other two suboptimal solutions provided by the BCD algorithm and HED algorithm, respectively.

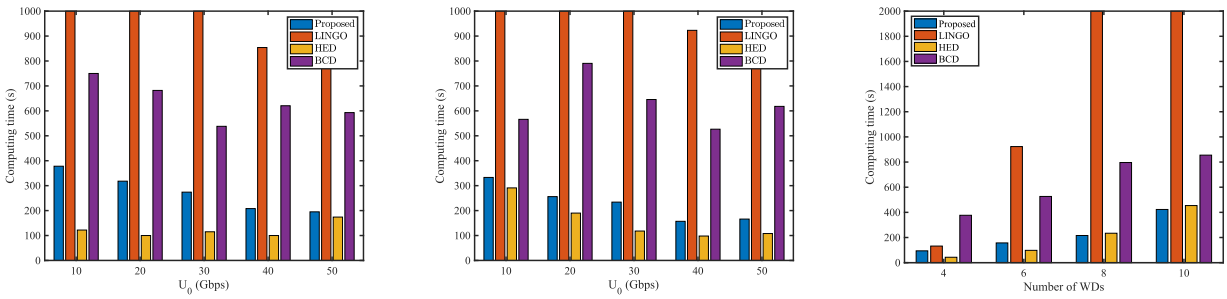


Fig. 5. Computing time comparisons among our proposed algorithm, LINGO, HED, and BCD. (a) $\{U_1, U_2, U_3\} = \{6, 4, 5\}, K = 3, I = 6$. (b) $\{U_1, U_2, U_3\} = \{5, 2, 3\}, K = 3, I = 6$. (c) $U_0 = 40$.

Followed by the parameter-settings in Fig. 4, Fig. 5 further illustrates the computation efficiency of TLHS algorithm (with CCD algorithm as its subroutine), in comparison with LINGO's global-solver, HED, and BCD. Thanks to the advantage of cell-based distributed optimization of each individual SBS and the efficiency bisection search on d_{Ω}^{sub}, all the tested cases in Fig. 5 validate that our proposed TLHS algorithm consumes a significantly less computation-time compared to LINGO's global-solver (which uses the conventional approach of branch-and-bound to solve the non-convex optimization problems numerically). In addition, our solution's computing time is also less than the BCD method in which the block coordinate descent operation consumes much time. Although the HED consumes less time than our solution while the number of WDs is small, the latency achieved by the HED is far worse than our solution (Fig. 4). In addition, due to the advantage of cell-based distributed optimization and bisection search of our solution, the gap between computing-time consumed by the HED and that of our solution becomes small when the number of WDs increases.

VII. Optimal WDs' NOMA-Groupings

Our CDD and TLHS algorithms in Section V solve Problem (OLM) and obtain the minimum overall latency under a given WDs' NOMA-groupings. With CDD and TLHS algorithms as the subroutines, we can continue to optimize the WDs' NOMA-groupings for further minimizing the overall latency of all WDs. This is the focus of this section.

A. NOMA-Grouping Model

We use a binary variable $a_{i,k}$ to denote the association between the WD i and SBS k. Let $a_{i,k} = 1$ if WD i is associated with the SBS k, and $a_{i,k} = 0$ otherwise. Since each WD is associated with one SBS and only two WDs form a NOMA-group at one time, variable $a_{i,k}$ is constrained by

$$\sum_{k \in \mathcal{K}} a_{i,k} = 1, \forall i \in \mathcal{I}. \tag{33}$$

$$\sum_{i \in \mathcal{I}} a_{i,k} = 2, \forall k \in \mathcal{K}. \tag{34}$$

Thus, for SBS k, we can denote $\mathbf{\Omega}_k = \{i | a_{i,k} = 1, \forall i \in \mathcal{I}\}, \forall k \in \mathcal{K}$ (as explained in footnote 2 before, we consider that the decoding order of these WDs in one NOMA-group is fixed according to their respective indices in $\mathcal{I}$). In this section, we consider that the number of the WDs is twice than the number of SBSs, i.e., $I = 2\ K$.[3] Correspondingly, we have

$$\sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{I}} a_{i,k} = I. \tag{35}$$

Accordingly, the optimization problem is expressed as

$$\begin{aligned} (\text{Group}): &\min_{\mathbf{\Omega}} d_{\mathbf{\Omega}} \\ \text{Subject to}: &\text{constraints:} (33), (34), (35), \\ \text{Variables}: &\mathbf{\Omega} = \{a_{i,k} \in \{0, 1\}\}_{i \in \mathcal{I}, k \in \mathcal{K}}. \end{aligned}$$

As a combinatorial optimization problem, the key feature of Problem (Group) is that its objective function cannot be analytically expressed. As a result, conventional gradient based approach is not applicable. The intuitive method to enumerate all possible $\{\mathbf{\Omega}_k\}_{k \in \mathcal{K}}$ also suffers from a significant complexity, especially when the number of the WDs is large. For instance, there exist more than one hundred thousand different combinations of $\{\mathbf{\Omega}_k\}_{k \in \mathcal{K}}$ when $I = 10$. To address this difficulty, in this paper, we adopt a CE-based learning algorithm which is an efficient methodology for addressing combinatorial optimization problems. Interested readers can refer to [22] for the details regarding the principle of the CE algorithm, and there have been several studies exploiting CE for solving different combinatorial problems [46]–[48]. The details of our CE-based learning algorithm are illustrated in the next subsection.

B. CE-Based Learning Algorithm for Problem (Group)

In our CE-based learning algorithm, we treat the values of $\{a_{i,k}\}_{i \in \mathcal{I}, k \in \mathcal{K}}$ as random variables. Each $a_{i,k}$ follows a Bernoulli distribution of parameter $\eta_{i,k}$. To find the optimal solution of Problem (Group), we use a CE-based learning algorithm to update $\{\eta_{i,k}\}_{i \in \mathcal{I}, k \in \mathcal{K}}$. The details are shown in Algorithm 3, and we explain important steps as follows.

- *(Step 4):* Let the number of solutions participating in each iteration be a constant denoted by M, and we generate a subset of feasible solutions $\overline{\mathbf{A}}(\tau) = \{A_1(\tau), A_2(\tau), \ldots,$

[3] Our following analysis and algorithm are also applicable by adding an appropriate number of virtual WDs (or SBSs) when $I \neq 2\ K$.

Algorithm 3: CE-based Algorithm for Problem (Group).

1: **Initialize:** Initialize $\{\eta_{i,k}(0)\}_{\forall i\in\mathcal{I},k\in\mathcal{K}}$, the learning rate $\alpha(0)$, and the stopping threshold ϵ_η.
2: Iteration $\tau = 1$.
3: **while** 1 **do**
4: According to the current $\{\eta_{i,k}(\tau)\}_{\forall i\in\mathcal{I},k\in\mathcal{K}}$, randomly generate M feasible solutions $\{A_m(\tau)\}_{m=1,2,...,M}$. Each $A_m(\tau)$ denotes a profile of $\{a^m_{i,k}(\tau)\}_{\forall i\in\mathcal{I},k\in\mathcal{K}}$ which form a feasible solution for Problem (Group).
5: For each feasible solution $A_m(\tau)$, input the corresponding $\{a^m_{i,k}(\tau)\}_{\forall i\in\mathcal{I},k\in\mathcal{K}}$ into Problem (OLM) and invoke TLHS algorithm to obtain the minimum overall latency $L(A_m(\tau)) = d_{A_m(\tau)}$.
6: Re-order $\{A_m(\tau)\}_{m=1,2,...,M}$ according to the order from the smallest $L(A_m(\tau))$ to the biggest $L(A_m(\tau))$.
7: Let $\hat{L}(\tau)$ be the ρ best solutions' quantile based on equation (38), and calculate quality coefficient of these best solutions to obtain $\{C_{i,k}(\tau+1)\}_{\forall i\in\mathcal{I},k\in\mathcal{K}}$.
8: Update probability distributions $\eta_{i,k}(\tau+1) = (1-\alpha(\tau))\eta_{i,k}(\tau) + \alpha(\tau)C_{i,k}(\tau+1), \forall i\in\mathcal{I}, k\in\mathcal{K}$ for next round of iteration.
9: **if** $\sum_{i\in\mathcal{I}}\sum_{k\in\mathcal{K}}\eta_{i,k}(\tau+1) - \eta_{i,k}(\tau) \le \epsilon_\eta$ **then**
10: Determine the optimal Bernoulli-distribution of $a_{i,k}$, $\eta^*_{i,k} = \eta_{i,k}(\tau+1), \forall i\in\mathcal{I}, k\in\mathcal{K}$.
11: Stop the WHILE-LOOP.
12: **else**
13: Update $\tau = \tau + 1$.
14: **end if**
15: **end while**
16: Generate A^* according to the $\{\eta^*_{i,k}\}_{i\in\mathcal{I},k\in\mathcal{K}}$.
17: **Output:** A^* is the optimal solution for Problem (Group).

$A_M(\tau)\}$ for Problem (Group) at iteration τ according to probabilities distribution $\{\eta_{i,k}(\tau)\}_{i\in\mathcal{I},k\in\mathcal{K}}$.

- *(Step 5 and Step 6):* We input each feasible solution $A_m(\tau)\in\overline{\mathbf{A}}(\tau)$ into Problem (OLM) to compute the corresponding latency $d_{A_m(\tau)}$

$$L(A_m(\tau)) = d_{A_m(\tau)}, \tag{36}$$

and re-order $\{L(A_m(\tau))\}_{\forall m\in M}$ from the smallest to the biggest

$$L(A_1(\tau)) \le L(A_2(\tau)) \le \ldots \le L(A_M(\tau)). \tag{37}$$

- *(Step 7 to Step 8):* We evaluate the quality coefficient $C_{i,k}(\tau+1)$ for the next round of iteration. Specifically, let $\hat{L}(\tau)$ be the ρ best solutions' quantile at iteration τ

$$\hat{L}(\tau) = L(A_{\lceil \rho M\rceil}(\tau)). \tag{38}$$

Here, the notation ρ equals to 0.1 according to the initial version of the CE method [22]. Then, quality coefficient $C_{i,k}(\tau+1)$ is computed based on the Kullback-Leibler cross-entropy approach [49]

$$C_{i,k}(\tau+1) = \frac{\sum_{m=1}^{M} I_{\{L(A_m(\tau))\le\hat{L}(\tau)\}} I_{\{a^m_{i,k}(\tau)=1\}}}{\sum_{m=1}^{M} I_{\{L(A_m(\tau))\le\hat{L}(\tau)\}}}. \tag{39}$$

Finally, $\eta_{i,k}(\tau+1)$ is updated in the next round of iteration

$$\eta_{i,k}(\tau+1) = (1-\alpha(\tau))\eta_{i,k}(\tau) + \alpha(\tau)C_{i,k}(\tau+1), \tag{40}$$

where $\alpha(\tau)\in[0,1]$ is the learning rate.

The above iterative process continues until the variation of each $\eta_{i,k}$ (i.e., the difference between $\eta_{i,k}(\tau+1)$ and $\eta_{i,k}(\tau)$) is small enough. When the iterative process stops, we update the optimal Bernoulli-distribution $\{\eta^*_{i,k}\}_{i\in\mathcal{I},k\in\mathcal{K}}$ as the current $\{\eta_{i,k}\}_{i\in\mathcal{I},k\in\mathcal{K}}$. Thus, we can obtain the optimal solution A^* of Problem (Group) according to $\{\eta^*_{i,k}\}_{i\in\mathcal{I},k\in\mathcal{K}}$.

According to the analysis of the CE algorithm in [50], as $\tau\to\infty$, the probability of obtaining the optimal solution A^* tends to 1 at the iteration τ. This property is based on two assumptions, i.e., the targeted problem is a combinatorial optimization problem, and its optimal solution is unique. In addition, the learning rate $\alpha(\tau)$ is set according to the following rule

$$\alpha(\tau) \le 1 - \frac{\log(\tau+1)}{\log(\tau+2)}, \tau\ge\Gamma. \tag{41}$$

Our Problem (Group) meets both two assumptions above. Thus, the CE-based learning algorithm can converge to the optimal solution $(\boldsymbol{\eta}^*, A^*)$, where A^* is the optimal solution, and η^* is defined as

$$\eta^*_{i,k} = \begin{cases} 1, & \text{if ai,k=1 in the A*,} \\ 0, & \text{otherwise.} \end{cases}$$

The complexity analysis of the CE-based algorithm can be given as follows. The CE-based algorithm executes an iterative procedure for finding the optimal NOMA-group. Each iteration involves generating M samples according to a group of probability distributions of variables $\{a_{i,k}\}_{\forall i\in\mathcal{I},k\in\mathcal{K}}$. Assuming that the number of iterations is fixed as χ, the complexity of the CE-based algorithm can be expressed as $\mathcal{O}(\chi M\frac{(t_0^{\text{ub}}-t_0^{\text{lb}})(\beta_k^{\text{ub}}-\beta_k^{\text{lb}})}{\Delta t_0\Delta\beta_k}\log_2(\frac{d_\Omega^{\text{ub}}-d_\Omega^{\text{lb}}}{\varepsilon}))$, where $\frac{(t_0^{\text{ub}}-t_0^{\text{lb}})(\beta_k^{\text{ub}}-\beta_k^{\text{lb}})}{\Delta t_0\Delta\beta_k}\log_2(\frac{d_\Omega^{\text{ub}}-d_\Omega^{\text{lb}}}{\varepsilon})$ is the complexity of solving Problem (OLM) provided in Section V-C before.

C. Results About WDs' NOMA-Groupings

To evaluate the performance of the CE-based learning algorithm, we set the parameters according to [46], such as $\rho = 0.1$, and test the CE-based learning algorithm on the scenario of $K = 4$ and $I = 8$.

Fig. 6 shows the convergence of all WDs' probabilities (i.e., $\{\eta_{i,k}\}_{i\in\mathcal{I},k\in\mathcal{K}}$) of choosing each individual SBS. Specifically, Subplot 6(a) shows the convergence of all WDs' Bernoulli-distribution parameters $\{\eta_{i,1}\}_{i\in\mathcal{I}}$ with respect to SBS 1. As shown in Subplot 6(a), after convergence of CE-algorithm,

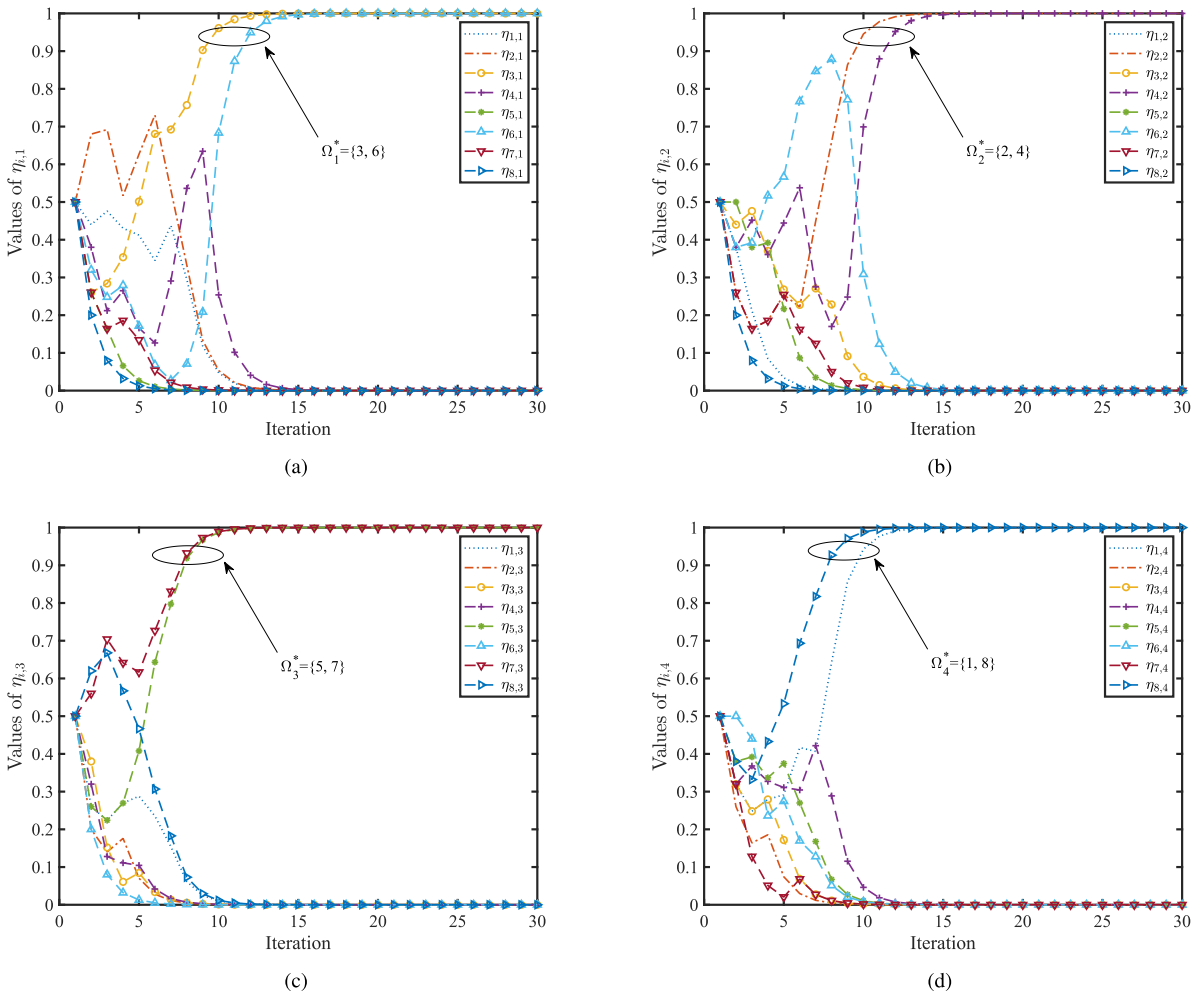


Fig. 6. The convergences of WDs' probabilities (i.e., $\{\eta_{i,k}\}_{i\in\mathcal{I},k\in\mathcal{K}}$) of choosing each SBS in the CE based learning algorithm. (a)The values of $\{\eta_{i,1}\}_{i\in\mathcal{I}}$. (b) The values of $\{\eta_{i,2}\}_{i\in\mathcal{I}}$. (c) The values of $\{\eta_{i,3}\}_{i\in\mathcal{I}}$. (d) The values of $\{\eta_{i,4}\}_{i\in\mathcal{I}}$.

$\eta_{3,1}$ and $\eta_{6,1}$ converge to 1, while all the other $\{\eta_{i,1}\}_{i\neq 3,6}$ converge to zero. These results mean that WD 3 and WD 6 form a NOMA-group for offloading to SBS 1. Subplot 6(b) demonstrates the similar results, in which we show the convergence of all WDs' $\{\eta_{i,2}\}_{i\in\mathcal{I}}$ with respect to SBS 2. In particular, after convergence, $\eta_{2,2}$ and $\eta_{4,2}$ converge to 1, while all other $\{\eta_{i,2}\}_{i\neq 2,4}$ all converge to zero, which means that WD 2 and WD 4 form a NOMA-group for offloading to SBS 2. Subplot 6(c) demonstrates the convergence of $\{\eta_{i,3}\}_{i\in\mathcal{I}}$, and Subplot 6 (d) demonstrates the convergence of $\{\eta_{i,4}\}_{i\in\mathcal{I}}$. As shown in the example in Fig. 6, after convergence, the optimal NOMA-grouping is Ω_1^*={3, 6}, Ω_2^*={2, 4}, Ω_3^*={5, 7}, and Ω_4^*={1, 8}, which are consistent with constraints (33), (34), and (35).

Fig. 7 shows the effectiveness of our CE-based learning algorithm in comparison with the globally optimal NOMA-groupings which are obtained by the enumeration method. Due to the prohibitive complexity in the enumeration method, we test two cases : i) $K = 4, U_0 = 30$ and ii) $K = 4, U_0 = 40$. As shown in Fig. 7, the overall latency of the CE algorithm after convergence can achieve the minimum latency which is achieved by the globally optimal NOMA-groupings by the enumeration, which thus validates the effectiveness of the CE algorithm.

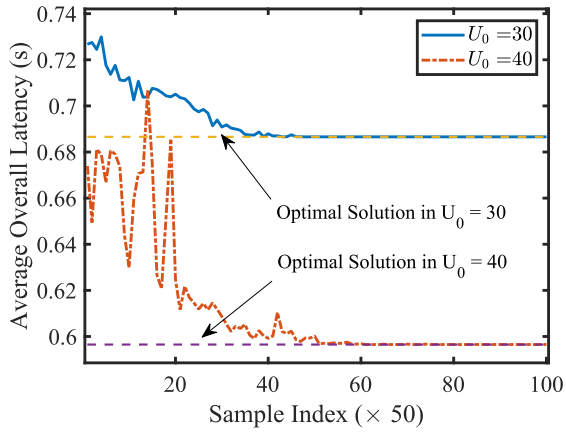


Fig. 7. The effectiveness of CE-based learning Algorithm.

Fig. 8 shows the comparisons between our CE-based algorithm and a randomized grouping scheme, in which all the WDs are randomly selected to form the NOMA-groups for offloading different SBSs. Fig. 8(a) shows the minimum latency under different values of U_0 obtained by the two algorithms. Fig. 8(b) shows the minimum latency under different

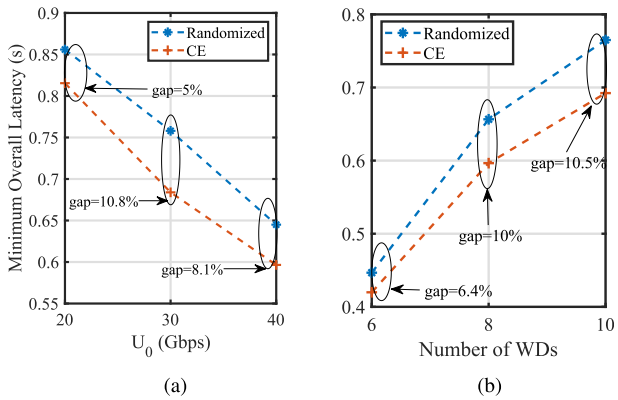


Fig. 8. Comparison between the CE algorithm and the randomized search algorithm: (a) Comparison under different values of U_0, and (b) Comparison under different numbers of the WDs. (a) $K = 4, I = 8$. (b) $U_0 = 40$.

numbers of the WDs. The results in both subplots demonstrate that the CE-based algorithm can effectively reduce the overall latency, compared to the randomized grouping. Moreover, as shown in Fig. 8(b), the benefit of the CE-based algorithm is more significant when the number of the WDs increases.

VIII. Conclusion

In this paper, we have investigated the multi-user dual computation offloading via the hybrid NOMA-FDMA transmission, in which each WD can simultaneously offload its workloads to a cloudlet at the MBS and to the ES at the SBS. We have formulated a joint optimization of all WDs' partial workloads offloading, their FDMA transmission to the MBS, different NOMA-groups' transmission to the SBSs, as well as the computing-rate allocation of the ESs and the cloudlet, with the objective of minimizing the overall latency for completing all WDs' workloads. Despite the strict non-convexity of the joint optimization problem, we have proposed a layered yet cell-based distributed algorithm for finding the optimal dual offloading solution. Based on this optimal dual offloading solution, we have further optimized the WDs' NOMA-groups for their offloading to the SBSs by exploiting the CE based learning algorithm. Numerical results have been provided to validate the effectiveness and efficiency of our proposed algorithms. In our future work, we will investigate the resource sharing between the MBS and the SBSs for improving the overall utilization efficiency of the computation-resources at the MBS and different SBSs. We will also further investigate the hybrid NOMA-FDMA assisted dual computation offloading problem under an imperfect channel state information.

References

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

[2] M. Tang, L. Gao, and J. Huang, "Communication, computation, and caching resource sharing for the Internet of Things," *IEEE Commun. Mag.*, vol. 58, no. 4, pp. 75–80, Apr. 2020.

[3] 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.

[4] J. Chen, H. Xing, Z. Xiao, L. Xu, and T. Tao, "A DRL agent for jointly optimizing computation offloading and resource allocation in MEC," *IEEE Internet Things J.*, vol. 8, no. 24, pp. 17508–17524, Dec. 2021.

[5] J. Zhang *et al.*, "Online optimization of energy-efficient user association and workload offloading for mobile edge computing," *IEEE Trans. Veh. Technol.*, vol. 71, no. 2, pp. 1974–1988, Feb. 2022.

[6] Z. Liang, Y. Liu, T. -M. Lok, and K. Huang, "Multi-cell mobile edge computing: Joint service migration and resource allocation," *IEEE Trans. Wireless Commun.*, vol. 20, no. 9, pp. 5898–5912, Sep. 2021.

[7] 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.

[8] J. Ren, G. Yu, Y. He, and G. Y. Li, "Collaborative cloud and edge computing for latency minimization," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 5031–5044, May 2019.

[9] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On multi-access edge computing: A survey of the emerging 5G network edge cloud architecture and orchestration," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1657–1681, Jul.–Sep. 2017.

[10] C. Rosa *et al.*, "Dual connectivity for LTE small cell evolution: Functionality and performance aspects," *IEEE Commun. Mag.*, vol. 54, no. 6, pp. 137–143, Jun. 2016.

[11] O. N. C. Yilmaz, O. Teyeb, and A. Orsino, "Overview of LTE-NR dual connectivity," *IEEE Commun. Mag.*, vol. 57, no. 6, pp. 138–144, Jun. 2019.

[12] M. Centenaro, D. Laselva, J. Steiner, K. Pedersen, and P. Mogensen, "Resource-efficient dual connectivity for ultra-reliable low-latency communication," in *Proc. IEEE 91st Veh. Technol. Conf.*, 2020, pp. 1–5.

[13] Y. Wu, Y. He, L. Qian, J. Huang, and X. Shen, "Optimal resource allocation for mobile data offloading via dual-connectivity," *IEEE Trans. Mobile Comput.*, vol. 17, no. 10, pp. 2349–2365, Oct. 2018.

[14] H. Cui and F. You, "User-centric resource scheduling for dual-connectivity communications," *IEEE Commun. Lett.*, vol. 25, no. 11, pp. 3659–3663, Nov. 2021.

[15] C. Li, H. Wang, and R. Song, "Intelligent offloading for NOMA-assisted MEC via dual connectivity," *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2802–2813, Feb. 2021.

[16] Y. Liu, W. Yi, Z. Ding, X. Liu, O. Dobre, and N. Al-Dhahir, "Application of NOMA in 6G networks: Future vision and research opportunities for next generation multiple access," 2021, *arXiv:2103.02334*.

[17] Z. Wei, L. Yang, D. W. K. Ng, J. Yuan, and L. Hanzo, "On the performance gain of NOMA over OMA in uplink communication systems," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 536–568, Jan. 2020.

[18] X. Liu, X. B. Zhai, W. Lu, and C. Wu, "QoS-Guarantee resource allocation for multibeam satellite industrial Internet of Things with NOMA," *IEEE Trans. Ind. Informat.*, vol. 17, no. 3, pp. 2052–2061, Mar. 2021.

[19] Y. Wu, Y. Song, T. Wang, L. Qian, and T. Q. S. Quek, "Non-orthogonal multiple access assisted federated learning via wireless power transfer: A cost-efficient approach," *IEEE Trans. Commun.*, vol. 70, no. 4, pp. 2853–2869, Apr. 2022.

[20] 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.

[21] J. Wang, J. Pan, F. Esposito, P. Calyam, Z. C. Yang, and P. Mohapatra, "Edge cloud offloading algorithms: Issues, methods, and perspectives," *ACM Comput. Surv.*, vol. 52, no. 1, pp. 1–23, Feb. 2019.

[22] P. T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Ann. oper. Res.*, vol. 134, no. 1, pp. 19–67, Feb. 2005.

[23] L. E. Schrage, *Optimization Modeling With LINGO*. Chicago, IL, USA: Lindo System, 2006.

[24] X. Chen, C. Wu, Z. Liu, N. Zhang, and Y. Ji, "Computation offloading in beyond 5G networks: A distributed learning framework and applications," *IEEE Wireless Commun.*, vol. 28, no. 2, pp. 56–62, Apr. 2021.

[25] T. Wang *et al.*, "An intelligent dynamic offloading from cloud to edge for smart IoT systems with Big Data," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 4, pp. 2598–2607, Oct–Dec. 2020.

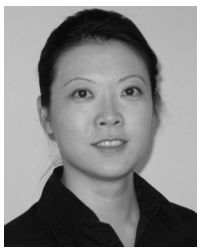
[26] K. Zhang, J. Cao, and Y. Zhang, "Adaptive digital twin and multiagent deep reinforcement learning for vehicular edge computing and networks," *IEEE Trans. Ind. Informat.*, vol. 18, no. 2, pp. 1405–1413, Feb. 2022.

[27] W. Y. B. Lim *et al.*, "Decentralized edge intelligence: A dynamic resource allocation framework for hierarchical federated learning," *IEEE Trans. Parallel Distrib. Syst.*, vol. 33, no. 3, pp. 536–550, Mar. 2022.
[28] X. Huang, S. Leng, S. Maharjan, and Y. Zhang, "Multi-agent deep reinforcement learning for computation offloading and interference coordination in small cell networks," *IEEE Trans. Veh. Technol.*, vol. 70, no. 9, pp. 9282–9293, Sep. 2021.
[29] M. Chen, B. Liang, and M. Dong, "Multi-user multi-task offloading and resource allocation in mobile cloud systems," *IEEE Trans. Wireless Commun.*, vol. 17, no. 10, pp. 6790–6805, Oct. 2018.
[30] Z. Kuang, L. Li, J. Gao, L. Zhao, and A. Liu, "Partial offloading scheduling and power allocation for mobile edge computing systems," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6774–6785, Aug. 2019.
[31] L. Qian, Y. Wu, F. Jiang, N. Yu, W. Lu, and B. Lin, "NOMA assisted multi-task multi-access mobile edge computing via deep reinforcement learning for industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5688–5698, Aug. 2021.
[32] W. Y. B. Lim, J. S. Ng, Z. Xiong, D. Niyato, C. Miao, and D. I. Kim, "Dynamic edge association and resource allocation in self-organizing hierarchical federated learning networks," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3640–3653, Dec. 2021.
[33] Y. Zhang, X. Lan, J. Ren, and L. Cai, "Efficient computing resource sharing for mobile edge-cloud computing networks," *IEEE/ACM Trans. Netw.*, vol. 28, no. 3, pp. 1227–1240, Jun. 2020.
[34] E. El Haber, T. M. Nguyen, and C. Assi, "Joint optimization of computational cost and devices energy for task offloading in multi-tier edge-clouds," *IEEE Trans. Commun.*, vol. 67, no. 5, pp. 3407–3421, May 2019.
[35] Y. Wang, X. Tao, X. Zhang, P. Zhang, and Y. T. Hou, "Cooperative task offloading in three-tier mobile computing networks: An ADMM framework," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2763–2776, Mar. 2019.
[36] J. Du, L. Zhao, J. Feng, and X. Chu, "Computation offloading and resource allocation in mixed fog/cloud computing systems with min-max fairness guarantee," *IEEE Trans. Commun.*, vol. 66, no. 4, pp. 1594–1608, Apr. 2018.
[37] J. Zhao, Q. Li, Y. Gong, and K. Zhang, "Computation offloading and resource allocation for cloud assisted mobile edge computing in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7944–7956, Aug. 2019.
[38] Z. Yang, C. Pan, J. Hou, and M. Shikh-Bahaei, "Efficient resource allocation for mobile-edge computing networks with NOMA: Completion time and energy minimization," *IEEE Trans. Commun.*, vol. 67, no. 11, pp. 7771–7784, Nov. 2019.
[39] Z. Ding, J. Xu, O. A. Dobre, and H. V. Poor, "Joint power and time allocation for NOMA–MEC offloading," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 6207–6211, Jun. 2019.
[40] Y. Wu, K. Ni, C. Zhang, L. P. Qian, and D. H. K. Tsang, "NOMA-Assisted multi-access mobile edge computing: A joint optimization of computation offloading and time allocation," *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 12244–12258, Dec. 2018.
[41] W. Wu, F. Zhou, R. Q. Hu, and B. Wang, "Energy-efficient resource allocation for secure NOMA-enabled mobile edge computing networks," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 493–505, Jan. 2020.
[42] Z. Ding, D. W. K. Ng, R. Schober, and H. V. Poor, "Delay minimization for NOMA-MEC offloading," *IEEE Signal Process. Lett.*, vol. 25, no. 12, pp. 1875–1879, Dec. 2018.
[43] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge, U.K.: Cambridge University Press, 2004.
[44] S. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale ℓ_1-regularized least squares," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 4, pp. 606–617, Dec. 2007.
[45] Y. Yang, M. Pesavento, Z. Q. Luo, and B. Ottersten, "Inexact block coordinate descent algorithms for nonsmooth nonconvex optimization," *IEEE Trans. Signal Process.*, vol. 68, pp. 947–961, 2020.
[46] M. Hu, W. Wang, W. Cheng, and H. Zhang, "Initial probability adaptation enhanced cross-entropy-based tone injection scheme for PAPR reduction in OFDM systems," *IEEE Trans. Veh. Technol.*, vol. 70, no. 7, pp. 6674–6683, Jul. 2021.
[47] M. Kovaleva, D. Bulger, B. A. Zeb, and K. P. Esselle, "Cross entropy method for electromagnetic optimization with constraints and mixed variables," *IEEE Trans. Antennas Propag.*, vol. 65, no. 10, pp. 5532–5540, Oct. 2017.
[48] S. Zhu, W. Xu, L. Fan, K. Wang, and G. K. Karagiannidis, "A novel cross entropy approach for offloading learning in mobile edge computing," *IEEE Wireless Commun. Lett.*, vol. 9, no. 3, pp. 402–405, Mar. 2020.
[49] A. E. Abbas, A. Cadenbach, and E. Salimi, "A Kullback–Leibler view of maximum entropy and maximum log-probability methods," *Entropy*, vol. 19, no. 5, pp. 1–14, May 2017.
[50] L. Margolin, "On the convergence of the cross-entropy method," *Ann. Oper. Res.*, vol. 134, no. 1, pp. 201–214, Feb. 2005.

Yang Li received the B.Sc. degree in computer science and technology from China Three Gorges University, Yichang, China, in 2015, and the M.S. degree in computer technology from Huaqiao University, Quanzhou, China, in 2018. She is currently working toward the Ph.D. degree with the Faculty of Science and Technology, University of Macau, Zhuhai, China. Her research focuses on edge-cloud collaborative computing.

Yuan Wu (Senior Member, IEEE) received the Ph.D. degree in electronic and computer engineering from the Hong Kong University of Science and Technology, Hong Kong, in 2010. He is currently an Associate Professor with the State Key Laboratory of Internet of Things for Smart City, University of Macau, Zhuhai, China, and also with the Department of Computer and Information Science, University of Macau. During 2016–2017, he was a Visiting Scholar with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. His research interests include resource management for wireless networks, green communications and computing, mobile edge computing, and edge intelligence. He was the recipient of the Best Paper Award from the IEEE International Conference on Communications in 2016, and the Best Paper Award from IEEE Technical Committee on Green Communications and Computing in 2017. Dr. Wu is currently on the Editorial Boards of the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING, IEEE INTERNET OF THINGS JOURNAL, and IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY.

Minghui Dai received the Ph.D. degree from Shanghai University, Shanghai, China, in 2021. He is currently a Postdoctoral Fellow with the State Key Laboratory of Internet of Things for Smart City, University of Macau, Zhuhai, China. His research interests include wireless network architecture and vehicular networks.



Bin Lin (Senior Member, IEEE) received the B.S. and M.S. degrees from Dalian Maritime University, Dalian, China, in 1999 and 2003 respectively, and the Ph.D. degree from the Broadband Communications Research Group, Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada, in 2009. She is currently a Full Professor with the Department of Information Science and Technology, Dalian Maritime University, Dalian, China. From 2015 to 2016, she was a Visiting Scholar with George Washington University, Washington, DC, USA. Her research interests include wireless communications, network dimensioning and optimization, resource allocation, artificial intelligence, maritime communication networks, edge/cloud computing, wireless sensor networks, and Internet of Things. Dr. Lin is an Associate Editor for *IET Communications*.

Weijia Jia (Fellow, IEEE) received the B.Sc. and M.Sc. degrees in computer science from Center South University, Changsha, China, in 1982 and 1984, respectively, and the Master of Applied Science and Ph.D. degrees in computer science from the Polytechnic Faculty of Mons, Mons, Belgium, in 1992 and 1993, respectively. He is currently the Chair Professor and Deputy Director of the State Kay Laboratory of Internet of Things for Smart City, University of Macau, Zhuhai, China. He has been the Zhiyuan Chair Professor with Shanghai Jiaotong University, Shanghai, China. From 1993 to 1995, he joined the German National Research Center for Information Science (GMD) in Bonn (St. Augustine) as a Research Fellow. From 1995 to 2013, he worked with the City University of Hong Kong, Hong Kong, as a Professor. His contributions have been recognized as optimal network routing and deployment, vertex cover, anycast and QoS routing, and sensors networking, knowledge relation extractions, NLP and edge computing. He has more than 500 publications in the prestige international journals/conferences and research books and book chapters. He was the recipient of the Best Product awards from the International Science & Tech. Expo (Shenzhen) in 2011/2012 and the 1st Prize of Scientific Research awards from the Ministry of Education of China in 2017 (list 2). He was the Area Editor of various prestige international journals, Chair and PC Member/ Keynote Speaker for many top international conferences. He is the Distinguished Member of CCF.

Xuemin Shen (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990. He is currently a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. His research interests include network resource management, wireless network security, Internet of Things, 5G and beyond, and vehicular networks. Dr. Shen is a registered Professional Engineer of Ontario, Canada, an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, Royal Society of Canada Fellow, Chinese Academy of Engineering Foreign Member, and Distinguished Lecturer of the IEEE Vehicular Technology Society and Communications Society. Dr. Shen was the recipient of the Canadian Award for Telecommunications Research from the Canadian Society of Information Theory in 2021, the R.A. Fessenden Award in 2019 from IEEE, Canada, Award of Merit from the Federation of Chinese Canadian Professionals (Ontario) in 2019, James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society, Joseph LoCicero Award in 2015 and Education Award in 2017 from the IEEE Communications Society (ComSoc), and Technical Recognition Award from Wireless Communications Technical Committee (2019) and AHSN Technical Committee (2013). He was also the recipient of the Excellent Graduate Supervision Award in 2006 from the University of Waterloo and the Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario. He was the Technical Program Committee Chair/Co-Chair for IEEE GLOBECOM'16, IEEE INFOCOM'14, IEEE VTC'10 Fall, IEEE GLOBECOM'07, and the Chair for the IEEE ComSoc Technical Committee on Wireless Communications. Dr. Shen is the President Elect of the IEEE ComSoc. He was the Vice President for Technical & Educational Activities, Vice President for Publications, Member-at-Large on the Board of Governors, the Chair of the Distinguished Lecturer Selection Committee, and a Member of the IEEE Fellow Selection Committee of the ComSoc. He is also the President of the IEEE ComSoc. He was the Editor-in-Chief of the IEEE INTERNET OF THINGS JOURNAL, IEEE NETWORK, and *IET Communications*.