Energy-Efficient Robust Computation Offloading for Fog-IoT Systems

Zhikun Wu, Bin Li, Zesong Fei, *Senior Member, IEEE*, Zhong Zheng, Bin Li, and Zhu Han, *Fellow, IEEE*

***Abstract*—As the computing nodes of a fog computing system are located at the network edge, it can provide low-latency and reliable computing services to Internet of Things (IoT) mobile devices (MDs). By wirelessly offloading all/part of the computational tasks from MDs to the infrastructure fog nodes, it addresses the contradiction between the limited battery capacity of MDs and their long-lasting operation requirement. Different from previous works, the uncertainty caused by the channel measurements is taken into account in this paper, which yields a robust offloading strategy against realistic channel estimation errors. For this system, we design an energy-efficient computation offloading strategy, while satisfying the delay constraint. By using the Conditional Value-at-Risk (CVaR) framework, the original offloading problem is transformed into a Mixed Integer Nonlinear Programming (MINLP) problem, which is complicated and very challenging to solve. To overcome this issue, we apply Benders decomposition to find the optimal offloading solution. Numerical results show that proposed offloading strategy efficiently achieves obtain the optimal solution of the MINLP problem, and is robust to channel estimation errors.**

***Index Terms*—Internet of Things, offloading, robust, conditional value-at-risk, benders decomposition.**

I. Introduction

According to the forecast of CISCO, there will be explosive growth in the data traffic and the data transmission rate of mobile devices in future mobile communication networks. From 2017 to 2022, the amount of data transmitted over the wireless network will increase by about 7 times, soaring to about 1 terabyte per year. The average speed of terminal data transmission will increase rapidly from 8.7 Mbps to 28.5Mbps [1]. This trend is mainly driven by emerging communication paradigms, such as Internet of Things (IoT), which impose several significant challenges on the current mobile networks, including a very low individual energy consumption [2]. On the other hand, Mobile Devices (MDs) become computing platforms within the IoT ecosystem, and may potentially consume a considerable amount of energy in computing-intensive applications, such as Virtual Reality (VR) and Augmented Reality (AR). As the battery capacity of MDs cannot be improved significantly [3], there exists contradiction between the limited battery lifetime and the long-lasting operation requirement of MDs [4].

Fog computing is a promising technology to reduce the energy consumption of the MDs. By deploying fog nodes (FNs) at the edge of networks, fog computing provides computation service to IoT MDs served by the corresponding FNs [5], [6]. In order to increase the battery life of MDs and guarantee the delay requirement of certain delay-sensitive IoT applications, IoT MDs may offload the energy-consuming local computation tasks to the nearby FNs, which have more relaxed energy constraints. Therefore, it is essential to design proper task offloading strategy in the fog-enabled IoT networks.

A large body of research has focused on the fog computing [4], [7]–[11]. More explicitly, the authors in [4] proposed a hierarchical fog-cloud computing paradigm for IoT systems, where a computation offloading game was formulated with the aim of maximizing the Quality of Experience (QoE) of each MD. The existence of Nash equilibrium was proven and a sub-optimal resource allocation strategy was proposed. In order to reduce the service delay for IoT-fog-cloud networks, the authors of [7] developed a collaboration and offloading framework for IoT applications. They innovatively considered queue length and different request types with variant processing times in load sharing. In [8], the offloading problem was investigated from the view of multi-objective optimization. The optimal offloading probability and transmit power were calculated for each MD to jointly minimize energy consumption, execution delay, and payment cost. In [9], the authors solved the delay and fairness guaranteed offloading problem in a fog-cloud system by optimizing the computation and wireless resources. A low-complexity suboptimal algorithm relying on semidefinite relaxation, randomization, fractional programming theory, and Lagrangian dual decomposition was proposed to solve the Mixed-Integer Non-Linear Programming (MINLP) problem. In [10], the authors investigated the power-delay trade off in the fog-cloud system. The workload allocation problem was decomposed into three subproblems, and those subproblems

Manuscript received May 5, 2019; revised November 14, 2019; accepted February 10, 2020. Date of publication February 19, 2020; date of current version April 16, 2020. This work was supported in part by Beijing Natural Science Foundation under Grant L182038, in part by the National Natural Science Foundation of China under Grants 61871032, 61701124, and 61901033, and in part by the Sichuan Science and Technology Program under Grant 2019YJ0105. The work of Z. Zheng was supported by the Beijing Institute of Technology Research Fund Program for Young Scholars. The review of this article was coordinated by Prof L. Wang. *(Corresponding author: Zesong Fei.)*

Zhikun Wu, Zesong Fei, and Zhong Zheng are with the School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China (e-mail: wuzhikun@bit.edu.cn; feizesong@bit.edu.cn; zhong.zheng@bit.edu.cn).

Bin Li is with the School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China, and also with the Engineering Research Center of Digital Forensics, Ministry of Education, Beijing 100081, China (e-mail: bin.li@nuist.edu.cn).

Bin Li is with the College of Electrical Engineering, Sichuan University, Chengdu 610065, China (e-mail: bin.li@scu.edu.cn).

Zhu Han is with the Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77004 USA (e-mail: zhan2@uh.edu).

Digital Object Identifier 10.1109/TVT.2020.2975056

were solved by interior-point methods, generalized Benders decomposition (GBD), and Hungarian algorithm, respectively. In [11], the social relationships of energy harvesting MDs were considered in the design of offloading strategy in IoT networks with fog computing. The proposed problem was transformed to a classical Nash equilibrium problem, and then be solved by semi-smooth Newton method with Armijo line. However, to the best of our knowledge, previous works barely take the channel estimation errors into account, which harms the robustness of those offloading strategies.

Against this background, we in this paper consider offloading all or part of the computation tasks of MDs to FNs in the mobile IoT systems, where the offloaded tasks are conveyed via wireless channels. Different from previous works, the channel estimation errors are taken into account when designing the the offloading strategy while satisfying certain delay constraint. Moreover, we aim to guarantee the robustness of the offloading strategy when the probability density functions of channel estimation errors are unknown. Conditional Value-at-Risk (CVaR) [12], [13] is introduced to handle the nondeterminacy of those errors, and Benders decomposition [14] is adopted to handle the transformed problem. The effectiveness of the proposed strategy is proved by the numerical simulation results.

The rest of this paper is organized as follows. The system model and problem formulation is detailed in Section II. In Section III, we detail the proposed algorithm based on CVaR and Benders decomposition. Section IV presents the simulation results. In Section V, we conclude this paper.

Notation: Vectors and matrices are denoted by bold text and bold uppercase text, respectively. ∇ denotes the first order derivation. $\mathbb{R}$ denotes the space of real numbers. $\boldsymbol{\phi}_{[k]}$ is the k-th element of vector $\boldsymbol{\phi}$, and $\boldsymbol{\Phi}_{[ki]}$ is the kth row and ith column element of matrix $\boldsymbol{\Phi}$. $\boldsymbol{A}^T$ means the transposition of A. diag(A) denotes the diagonal matrix of which the elements of main diagonal are composed of the elements of vector A. $[a]^+$ means $\max(a, 0)$. $\mathcal{O}$ is a complexity notation. A list of the notations used throughout the paper is given in Table I.

II. System Model and Problem Formulation

A. Network, Task, and Offloading Model

As shown in Fig. 1, an IoT-fog architecture is considered in this work. There are K FNs randomly deployed in a given area to serve U IoT MDs. FNs are indexed by k, and MDs are indexed by i. The sets of FNs and MDs are denoted by $\mathcal{K}$ and $\mathcal{U}$, respectively. For the ith MD, only one corresponding indivisible computation task i needs to be computed in each time slot, and the maximum tolerable latency introduced by task computing is set to be T. Moreover, we assume the situation that the latency i exceeds T is tolerable if the probability of this situation is strictly limited. The tasks can be either computed locally at the MD, or uploaded to the serving FN and transmitted back via wireless interface, and ith task is described by (L_i, C_i, ϵ_i), where L_i and C_i denote the length of the task and the estimated CPU circles needed to compute the task, respectively. As for ϵ_i, it means the maximum tolerable probability of the violation of the maximum latency constraint.

TABLE I
Parameter Notations

Notation	Implication
k	Index of FNs
i	Index of MDs
$\mathcal{K}_i$	The set of FNs which can serve ith MD
K_i	Number of FNs which can serve ith MD
K	Number of FNs
U	Number of IoT MDs
$\mathcal{U}_k$	Set of MDs which can be served by kth FN
T	Maximal application delay
L_i	Length of the ith task
C_i	Complexity of the ith task
G_{ki}	Channel power gain between kth FN and ith MD
F_k^{FN}	Maximum computational rate of kth FN
F_i^{MD}	Maximum computational rate of ith MD
P_i	Transmit power of ith MD
σ_i^2	Noise power of ith MD
ξ_i	Energy consumption coefficient of ith MD
B	Bandwidth
ϵ_i	QoS demand about delay of ith task
ΔG_{ki}	Error of channel power gain between kth FN and ith MD
f_{ki}^{FN}	Computational rate allocated to to ith task from kth FN
f_i^{MD}	Computational rate allocated to ith task from ith MD
x_{ki}	Indicate whether ith task is computed in kth FN
x_i	Indicate whether ith task is computed locally

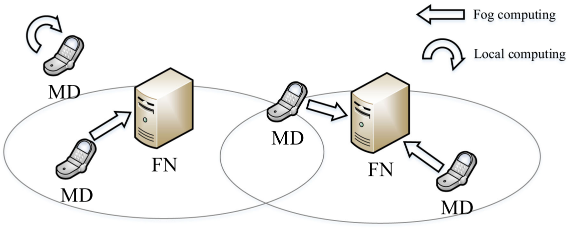


Fig. 1. System model.

B. Communication Model

In this IoT-fog architecture, each MD can upload the computation task to nearby FNs via wireless channel. The set of FNs serving the ith MD are denoted as $\mathcal{K}_i$, and the cardinality of $\mathcal{K}_i$ is denoted as $|\mathcal{K}_i| = K_i$. Without loss of generality, we assume the number of serving FNs $\{K_i\}_{1 \leq i \leq U}$ are equal for all MDs. The set $\mathcal{K}_i$ can be determined as those serving FNs, where the distance between those FNs and ith MD is shorter than a given threshold. Similarly, define $\mathcal{U}_k$ as the set of MDs that can be served by the kth FN. Assume the interferences are avoided by orthogonal wireless resource allocation, the Signal to Noise Ratio (SNR) of the uplink channel between the ith MD and the kth FN is given as

$$\text{SNR}_{ki}(\Delta G_{ki}) = \frac{P_i(G_{ki} + \Delta G_{ki})}{\sigma_i^2}, \tag{1}$$

where P_i is the transmission power of the ith MD, σ_i^2 is the power of the additive white Gaussian noise, G_{ki} is the estimated channel gain between the kth FN and the ith MD, and ΔG_{ki} is caused by the channel estimation error. We assume that the estimation error ΔG_{ki} follows an unknown distribution $\mathbb{P}_{\Delta G_{ki}}$ with mean $\mathbb{E}[\Delta G_{ki}] = \mu_{\Delta G_{ki}}$ and variance $\text{Var}[\Delta G_{ki}] = \Sigma_{\Delta G_{ki}}$. The achievable data rate between the ith MD and the kth FN is calculated as the Shannon rate of the wireless channel as

$$R_{ki}(\Delta G_{ki}) = B\log\left(1+\mathrm{SNR}_{ki}(\Delta G_{ki})\right). \tag{2}$$

where B denotes the transmission bandwidth. The power consumption for uploading the ith task from the ith MD to the kth FN is calculated as

$$p^t_{ki}(\Delta G_{ki}) = \frac{P_i L_i}{R_{ki}(\Delta G_{ki})}. \tag{3}$$

C. Local Computing Model

Let the indicator x_i denote whether the ith task is computed locally at the ith MD, i.e.,

$$x_i = \begin{cases} 1, & \text{if } i\text{th task is computed locally in } i\text{th MD,} \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Let f^{MD}_i denote the local computational rate (CPU cycles/s) of ith MD in the given time slot, which is limited by the maximum computational rate of ith MD, namely F^{MD}_i. The latency due to the local computation at the ith MD can be calculated as

$$t^{\mathrm{MD}}_i(f^{\mathrm{MD}}_i) = \frac{C_i}{f^{\mathrm{MD}}_i}. \tag{5}$$

The corresponding energy consumption is given by [15]

$$p^{\mathrm{MD}}_i = \xi_i C_i, \tag{6}$$

where ξ_i denotes the power consumption of per CPU cycle.

D. Fog Computing Model

Let the indicator x_{ki} denote whether the computation ith task is offloaded to the kth FN, defined as

$$x_{ki} = \begin{cases} 1, & \text{if } i\text{th task is computed in} k\text{th FN,} \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

The total delay caused by offloading the ith task from the ith MD to the kth FN includes the transmission delay and the FN's computing time. That is,

$$t^{\mathrm{FN}}_{ki}(f^{\mathrm{FN}}_{ki}, \Delta G_{ki}) = \frac{L_i}{R_{ki}(\Delta G_{ki})} + \frac{C_i}{f^{\mathrm{FN}}_{ki}}, \tag{8}$$

where f^{FN}_{ki} is the computational rate of the kth FN allocated to the ith task. The first term of (8) is the transmission delay for uploading the computation task from the ith MD to the kth FN, and the second term is the corresponding computation time at the FN. Note that we assume the output of each task is very short compared to the task itself. This assumption holds for many scenarios such as video processing, feature extraction and pattern recognition algorithms, where the program codes and input parameters size are much bigger than the output data. Therefore, the transmission delay for downloading the output from FN to MD is omitted. The computational rate of the kth FN is limited by F^{FN}_k, which is the maximum computational rate. We summarized the notations used in this paper in Table I.

E. Computation Offloading Problem

The total latency of the computation ith task is calculated as

$$t_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) = x_i t^{\mathrm{MD}}_i(f^{\mathrm{MD}}_i) + \sum_{k\in\mathcal{K}_i} x_{ki} t^{\mathrm{FN}}_{ki}(f^{\mathrm{FN}}_{ki}, \Delta G_{ki}), \tag{9}$$

where $\boldsymbol{f}_i = [f^{\mathrm{FN}}_{1i}, f^{\mathrm{FN}}_{2i}, \ldots, f^{\mathrm{FN}}_{Ki}, f^{\mathrm{MD}}_i]^T$, $\boldsymbol{x}_i = [x_{1i}, x_{2i}, \ldots, x_{Ki}, x_i]^T$, and $\Delta\boldsymbol{G}_i = [\Delta G_{1i}, \Delta G_{2i}, \ldots, \Delta G_{Ki}]^T$. It is worth noting that zeros are inserted as elements of the vectors $\boldsymbol{f}_i$, $\boldsymbol{x}_i$, and $\Delta\boldsymbol{G}_i$ for notation convenience. For instance, $x_{li} = 0$, $\forall l \in \mathcal{K}\setminus\mathcal{K}_i$. The latency of the ith task is constrained in a probabilistic manner such that the probability of the latency $t_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i)$ within the maximum tolerable delay T should be larger than a threshold $1-\epsilon_i$, i.e.,

$$\Pr\left[t_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) - T \le 0\right] \ge 1-\epsilon_i, \tag{10}$$

Note that the randomness of the delay t_i is due to the channel estimation error $\Delta\boldsymbol{G}_k$. The power consumed for the ith task in ith MD is formulated as

$$p_i(\boldsymbol{x}_i, \Delta\boldsymbol{G}_i) = x_i p^{\mathrm{MD}}_i + \sum_{k\in\mathcal{K}_i} x_{ki} p^t_{ki}(\Delta G_{ki}), \tag{11}$$

In the IoT systems, the battery capacity of MDs are typically limited. Therefore, we consider the minimizations of the total power consumption of MDs by jointly configuring the task offloading specified by the vectors $\{\boldsymbol{x}_i\}_{1\le i\le U}$ and allocating the computation resource specifying by the vectors $\{\boldsymbol{f}_i\}_{1\le i\le U}$. Therefore, the task offloading problem can be formulated as

$$\min_{\{\boldsymbol{x}_i\in\mathcal{X}_i\},\{\boldsymbol{f}_k\in\mathcal{F}_i\}} \quad \mathbb{E}\left[\sum_{i\in\mathcal{U}} p_i(\boldsymbol{x}_i, \Delta\boldsymbol{G}_i)\right] \tag{12a}$$

$$\text{s.t.} \quad x_i + \sum_{k\in\mathcal{K}_i} x_{ki} = 1, \forall i\in\mathcal{U}, k\in\mathcal{K}_i, \tag{12b}$$

$$\sum_{i\in\mathcal{U}_k} f^{\mathrm{FN}}_{ki} \le F^{\mathrm{FN}}_k, \forall i\in\mathcal{U}, \tag{12c}$$

$$\Pr\left[t_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) - T \le 0\right] \ge 1-\epsilon_i, \tag{12d}$$

where $\mathcal{X}_i = \{\boldsymbol{x}_i : \boldsymbol{x}_{i[k]} \in \{0,1\}, 1\le k\le K+1\}$; $\mathcal{F}_i = \{\boldsymbol{f}_i : 0 \le \boldsymbol{f}_{i[k]} \le F^{\mathrm{FN}}_k\}, 1\le k\le K, 0\le \boldsymbol{f}_{i[K+1]} \le F^{\mathrm{MD}}_i\}$; (12b) means each task can be computed either in local or in fog; (12c) is the constraint of the calculation ability of FN; (12d) is the probability constraint of the violation of the maximum tolerable delay. It is worth noting that we assume FNs are powered by external power system, and the battery lives of MDs are considered preferentially. Therefore, the power consumption of FNs are not considered in (12).

III. Algorithm Design

As the distribution of the channel estimation error $\mathbb{P}_{\Delta\boldsymbol{G}_i}$ is generally unknown, it is hard to solve the task offloading problem (12) directly. Considering the fact that ΔG_{ki} is relatively small compared to G_{ki}, we adopt the first order Taylor expansion to approximate (12a) and (10), which yields explicit formulations.

In particular, the first order Taylor expansion of $p^t_{ki}(\Delta G_{ki})$ is written as

$$\hat{p}^t_{ki}(\Delta G_{ki}) = p^t_{ki}(\Delta G_{ki}=0) + {p^t_{ki}}'(\Delta G_{ki}=0)\Delta G_{ki}, \quad (13)$$

where

$$\begin{aligned} &{p^t_{ki}}'(\Delta G_{ki}=0) \\ &\quad = -\frac{L_i P_i^2}{B\ln 2}\left(\log\left(1+\frac{P_i G_{ki}}{\sigma_i^2}\right)\right)^{-2}(\sigma_i^2 + P_i G_{ki})^{-1}. \end{aligned} \quad (14)$$

Moreover, let

$$\begin{aligned} \tilde{p}^t_{ki} = \mathbb{E}\left[\hat{p}^t_{ki}(\Delta G_{ki})\right] &= p^t_{ki}(\Delta G_{ki}=0) \\ &+ {p^t_{ki}}'(\Delta G_{ki}=0)\mu_{\Delta G_{ki}}. \end{aligned} \quad (15)$$

Then, (12a) can be approximated by

$$\mathbb{E}\left[p_i(\boldsymbol{x}_i, \Delta\boldsymbol{G}_i)\right] \approx \tilde{p}_i(\boldsymbol{x}_i) = x_i p_i^{\mathrm{MD}} + \sum_{k\in\mathcal{K}_i} x_{ki}\tilde{p}^t_{ki}. \quad (16)$$

Similarly, the first order Taylor expansion of the induced delay $t_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i)$ can be written as

$$\begin{aligned} &\hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) \\ &\quad = t_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i = \boldsymbol{0}) + \nabla_{\Delta\boldsymbol{G}_i} t_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i=\boldsymbol{0})^T\Delta\boldsymbol{G}_i, \end{aligned} \quad (17)$$

where

$$\begin{aligned} &\nabla_{\Delta\boldsymbol{G}_i} t_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i = \boldsymbol{0})_{[k]} \\ &\quad = -\frac{x_{ki} L_i P_i}{B\ln 2}\left(\log\left(1+\frac{P_i G_{ki}}{\sigma_i^2}\right)\right)^{-2}\left(\sigma_i^2 + P_i G_{ki}\right)^{-1}. \end{aligned} \quad (18)$$

Therefore, the distribution function (10) can be approximated by

$$\Pr\left[\hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) - T \le 0\right] \ge 1 - \epsilon_i. \quad (19)$$

The set of all possible distributions of $\Delta\boldsymbol{G}_i$ can be defined by

$$\begin{aligned} \mathscr{P}_{\Delta\boldsymbol{G}_i} = \{&\mathbb{P}_{\Delta\boldsymbol{G}_i} | \mathbb{E}_{\mathbb{P}_{\Delta\boldsymbol{G}_i}}[\Delta\boldsymbol{G}_i] = \boldsymbol{\mu}_{\Delta\boldsymbol{G}_i}, \\ &\mathbb{E}_{\mathbb{P}_{\Delta\boldsymbol{G}_i}}\left[(\Delta\boldsymbol{G}_i - \boldsymbol{\mu}_{\Delta\boldsymbol{G}_i})(\Delta\boldsymbol{G}_i - \boldsymbol{\mu}_{\Delta\boldsymbol{G}_i})^T\right] = \boldsymbol{\Sigma}_{\Delta\boldsymbol{G}_i}\}, \end{aligned} \quad (20)$$

where $\boldsymbol{\mu}_{\Delta\boldsymbol{G}_i} = [\mu_{\Delta G_{1i}}, \mu_{\Delta G_{1i}}, \ldots, \mu_{\Delta G_{Ki}}]^T$; $\mathbb{P}_{\Delta\boldsymbol{G}_i}$ is a distribution of $\Delta\boldsymbol{G}_i$, where the mean and the covariance of $\Delta\boldsymbol{G}_i$ are fixed to $\mu_{\Delta\boldsymbol{G}_i}$ and $\Sigma_{\Delta\boldsymbol{G}_i}$, respectively. It is worth mentioning that we assume the gains of different channels are independent. Therfore, $\boldsymbol{\Sigma}_{\Delta\boldsymbol{G}_i}$ is a diagonal matrix, and $\boldsymbol{\Sigma}_{\Delta\boldsymbol{G}_i} = \mathrm{diag}(\Delta\boldsymbol{G}_i)$. In order to guarantee the performance of proposed algorithms in different distributions of $\Delta\boldsymbol{G}_i$, we consider that (10) should hold in the worst case [16], [17]. Therefore, we assume

$$\inf_{\mathbb{P}_{\Delta\boldsymbol{G}_i}\in\mathscr{P}_{\Delta\boldsymbol{G}_i}} \Pr\left[\hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) - T \le 0\right] \ge 1 - \epsilon_i, \quad (21)$$

Then, we solve the following problem instead

$$\begin{aligned} \min_{\{\boldsymbol{x}_i\in\mathcal{X}_i\},\{\boldsymbol{f}_k\in\mathcal{F}_i\}} \quad & \sum_{i\in\mathcal{U}} \tilde{p}_i(\boldsymbol{x}_i) \\ \text{s.t.} \quad & \text{(12b), (12c), (21).} \end{aligned} \quad (22)$$

Evidently, (22) is a non-convex nonlinear optimization problem, and it is hard to solve directly. Conditional Value-at-Risk (CVaR) and Value-at-Risk (VaR) (see e.g., [12] and [13]) are risk measures, which were widely used in economy and have been applied in communications [18], [19]. VaR can be used to describe the "risk" of violating the constraint. It is, in fact, equivalent to (19) [20]. CVaR is more conservative than VaR which is the "average risk" of violating the constraint [20]. However, as shown in [18], [19], [21], CVaR is equivalent to (19) under the distributionally robust setting, which is (21). Alternatively, we adopt the CVaR framework to transform (21) into a tractable formulation. Then, solve the problem (22) with the transformed version of the constraint (21) using the by Benders decomposition. As CVaR have different forms, we adopt the definition detailed in [13]. The CVaR of $\Delta\boldsymbol{G}_i$ with loss function $\hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) - T$ and tolerance ϵ_i can be defined as

$$\begin{aligned} &\mathbb{P}_{\Delta\boldsymbol{G}_i} - \mathrm{CVaR}_{\epsilon_i}\left(\hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) - T\right) \\ &\quad = \inf_{\beta\in\mathbb{R}}\{\beta + \frac{1}{\epsilon_i}\mathbb{E}[\left(\hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) - T - \beta\right)^+]\}, \end{aligned} \quad (23)$$

where β is an auxiliary variable. It is worth noting that the CVaR detailed in (20) can provide a tight convex approximation to (21). According to Sec. 4.3.3 of [22], it can be proved that

$$\begin{aligned} &\inf_{\mathbb{P}_{\Delta\boldsymbol{G}_i}\in\mathscr{P}_{\Delta\boldsymbol{G}_i}} \Pr\left[\hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) - T\right. \\ &\quad \left.\le \mathbb{P}_{\Delta\boldsymbol{G}_i} - \mathrm{CVaR}_{\epsilon_i}\left(\hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) - T\right)\right] \ge 1 - \epsilon_i. \end{aligned} \quad (24)$$

We further restrict $\mathbb{P}_{\Delta\boldsymbol{G}_i} - \mathrm{CVaR}_{\epsilon_i}(\hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) - T) \le 0$, and (24) can be rewritten as

$$\inf_{\mathbb{P}_{\Delta\boldsymbol{G}_i}\in\mathscr{P}_{\Delta\boldsymbol{G}_i}} \Pr\left[\hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) - T \le 0\right] \ge 1 - \epsilon_i. \quad (25)$$

Therefore,

$$\begin{aligned} &\mathbb{P}_{\Delta\boldsymbol{G}_i} - \mathrm{CVaR}_{\epsilon_i}\left(\hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) - T\right) \le 0 \\ &\quad \Rightarrow \inf_{\mathbb{P}_{\Delta\boldsymbol{G}_i}\in\mathscr{P}_{\Delta\boldsymbol{G}_i}} \Pr\left[\hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) - T \le 0\right] \ge 1 - \epsilon_i. \end{aligned} \quad (26)$$

As (26) should hold for different $\mathbb{P}_{\Delta\boldsymbol{G}_i}$, it can be derived as [18], [19], [21]

$$\begin{aligned} &\sup_{\mathbb{P}_{\Delta\boldsymbol{G}_i}\in\mathscr{P}_{\Delta\boldsymbol{G}_i}} \mathbb{P}_{\Delta\boldsymbol{G}_i} - \mathrm{CVaR}_{\epsilon_i}\left(\hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) - T\right) \le 0 \\ &\quad \Rightarrow \inf_{\mathbb{P}_{\Delta\boldsymbol{G}_i}\in\mathscr{P}_{\Delta\boldsymbol{G}_i}} \Pr\left[\hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i) - T \le 0\right] \ge 1 - \epsilon_i. \end{aligned} \quad (27)$$

As $\hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta\boldsymbol{G}_i)$ is linear about $\Delta\boldsymbol{G}_i$, the feasible region of $\{\boldsymbol{x}_i, \boldsymbol{f}_i\}$ defined by the left hand side of (27) is equivalent to that defined by the following inequality system [18], [19], [21]

$$\boldsymbol{\Psi}_i \succeq \boldsymbol{0}, \quad (28a)$$

$$\beta_i + \frac{1}{\epsilon_i}\mathrm{Tr}(\boldsymbol{\Omega}_i\boldsymbol{\Psi}_i) \le 0, \quad (28b)$$

$$\boldsymbol{\Psi}_i - \boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) \succeq \boldsymbol{0}. \quad (28c)$$

where Ψ_i and β_i are introduced as auxiliary variables; Ω_i and $\boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i)$ are defined as

$$\begin{bmatrix} \boldsymbol{\Sigma}_{\Delta \boldsymbol{G}_i} + \boldsymbol{\mu}_{\Delta \boldsymbol{G}_i} \boldsymbol{\mu}_{\Delta \boldsymbol{G}_i}^T & \boldsymbol{\mu}_{\Delta \boldsymbol{G}_i} \\ \boldsymbol{\mu}_{\Delta \boldsymbol{G}_i}^T & 1 \end{bmatrix} \tag{29}$$

and

$$\begin{bmatrix} \boldsymbol{0} & \frac{1}{2}\nabla_{\Delta \boldsymbol{G}_i} \hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta \boldsymbol{G}_i = \boldsymbol{0}) \\ \frac{1}{2}\nabla_{\Delta \boldsymbol{G}_i} \hat{t}_i (\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta \boldsymbol{G}_i = \boldsymbol{0})^T & \hat{t}_i(\boldsymbol{x}_i, \boldsymbol{f}_i, \Delta \boldsymbol{G}_i = \boldsymbol{0}) - T - \beta_i \end{bmatrix}, \tag{30}$$

respectively. Substituting the implicit delay constraint (21) with the explicit expressions (28b) and (28c), the problem (22) becomes

$$\min_{\{\boldsymbol{x}_i \in \mathcal{X}_i\}, \{\boldsymbol{f}_k \in \mathcal{F}_i\}, \{\beta_i\}, \{\Psi_i\}} \quad \sum_{i \in \mathcal{U}} \tilde{p}_i(\boldsymbol{x}_i)$$

$$\text{s.t.} \qquad \text{(12b), (12c), (28a), (28b), (28c).} \tag{31}$$

Since the optimizing variable $\boldsymbol{x}_i$ is a binary integer vector and the constraint (28) is nonlinear, (31) is an MINLP problem, which is difficult to solve directly. Evidently, if the integer vector $\boldsymbol{x}$ is given, (31) becomes a convex optimization problem. Therefore, it is suitable to apply Benders decomposition [14] to decompose (31) into two relatively simple subproblems, namely the primal problem and the master problem. According to [24], the primal problem is given by

Primal problem:

$$\min_{\{\boldsymbol{f}_k \in \mathcal{F}_i\}, \{\beta_i\}, \{\boldsymbol{A}_i\}} \quad \delta_p = \sum_{i \in \mathcal{U}} \tilde{p}_i(\boldsymbol{x}_i) \tag{32a}$$

$$\text{s.t.} \qquad x_{ki} = x_{ki}^n : \lambda_{ki}, \forall i \in \mathcal{U}, k \in \mathcal{K}_i, \tag{32b}$$

$$x_i = x_i^n : \phi_i, \forall i \in \mathcal{U}, \tag{32c}$$

$$\boldsymbol{\Psi}_i - \boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) \succeq \boldsymbol{0} : \boldsymbol{\Phi}_i, \forall i \in \mathcal{U}$$

$$\text{(12c),} \tag{32d}$$

where λ_{ki}, ϕ_i, and $\boldsymbol{\Phi}_i$ are Lagrangian multipliers, and δ_p is the minimum value of the objective function of the object in primal problem, which is a lower bound of (31). By centralized solvers such as CVX, we can solve (32), and obtain the optimal values of $\boldsymbol{f}_i$, β_i, $\boldsymbol{\Psi}_i$ and $\boldsymbol{\Phi}_i$, $\forall i \in \mathcal{U}$. As those multipliers of integers are needed in the master problem, they can be expressed in closed-form expressions.

The Lagrangian of (32) can be expressed as

$$\mathcal{L} = \sum_{i \in \mathcal{U}} \tilde{p}_i(\boldsymbol{x}_i) + \phi_i(x_i - x_i^n)$$

$$+ \operatorname{Tr}\left(\left(\boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) - \boldsymbol{\Psi}_i\right) \boldsymbol{\Phi}_i\right) + \sum_{k \in \mathcal{K}_i} \lambda_{ki}(x_{ki} - x_{ki}^n)$$

$$\text{s.t.} \quad (12c). \tag{33a}$$

According to the the Karush-Kuhn-Tucker (KKT) condition [27], we obtain

$$\nabla_{x_{ki}} \mathcal{L} = \frac{P_i L_i}{R_{ki}(\Delta G_{ki} = 0)} + Q_{ki} + \lambda_{ki}$$

$$+ \nabla_{x_{ki}} \operatorname{Tr}\left(\left(\boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) - \boldsymbol{\Psi}_i\right) \boldsymbol{\Phi}_i\right) = 0, \tag{34a}$$

$$\nabla_{x_i} \mathcal{L} = \xi_i C_i + \phi_i + \nabla_{x_i} \operatorname{Tr}\left(\left(\boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) - \boldsymbol{\Psi}_i\right) \boldsymbol{\Phi}_i\right) = 0. \tag{34b}$$

where

$$Q_{ki} = -\frac{L_i P_i^2}{B \ln 2} \left(\log(1 + \frac{P_i G_{ki}}{\sigma_i^2}) \right)^{-2} (\sigma_i^2 + P_i G_{ki})^{-1} \mu_{G_{ki}}. \tag{35}$$

In order to obtain λ_{ki} and ϕ_i, we need to solve $\nabla_{x_{ki}} \operatorname{Tr}((\boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) - \boldsymbol{\Psi}_i)\boldsymbol{\Phi}_i)$ and $\nabla_{x_i} \operatorname{Tr}((\boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) - \boldsymbol{\Psi}_i)\boldsymbol{\Phi}_i)$. Evidently,

$$\begin{aligned} &\nabla_{x_{ki}} \operatorname{Tr}\left(\left(\boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) - \boldsymbol{\Psi}_i\right) \boldsymbol{\Phi}_i\right) \\ &= \nabla_{x_{ki}} \operatorname{Tr}\left(\boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) \boldsymbol{\Phi}_i\right) \\ &= \nabla_{x_{ki}} \boldsymbol{1}^T \left(\boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) \odot \boldsymbol{\Phi}_i\right) \boldsymbol{1} \\ &= \left(\frac{L_i}{R_{ki}(\Delta G_{ki} = 0)} + \frac{C_i}{f_{ki}^{\mathrm{FN}}} \right) \boldsymbol{\Phi}_{i_{[K+1, K+1]}} - \\ &\quad \times \frac{L_i P_i}{2B \ln 2} \left(\log\left(1 + \frac{P_i G_{ki}}{\sigma_i^2}\right) \right)^{-2} \left(\sigma_i^2 + P_i G_{ki}\right)^{-1} \\ &\quad \times \left(\boldsymbol{\Phi}_{i_{[k, K+1]}} + \boldsymbol{\Phi}_{i_{[K+1, k]}} \right). \end{aligned} \tag{36}$$

Similarly,

$$\begin{aligned} &\nabla_{x_i} \operatorname{Tr}\left(\left(\boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) - \boldsymbol{\Psi}_i\right) \boldsymbol{\Phi}_i\right) \\ &= \nabla_{x_i} \operatorname{Tr}\left(\boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) \boldsymbol{\Phi}_i\right) \\ &= \nabla_{x_i} \boldsymbol{1}^T (\boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) \odot \boldsymbol{\Phi}_i) \boldsymbol{1} \\ &= \frac{C_i}{f_i^{\mathrm{MD}}} \boldsymbol{\Phi}_{i_{[K+1, K+1]}}. \end{aligned} \tag{37a}$$

Hereby, we acquire $\nabla_{x_{ki}} \operatorname{Tr}((\boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) - \boldsymbol{\Psi}_i)\boldsymbol{\Phi}_i)$ and $\nabla_{x_i} \operatorname{Tr}((\boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) - \boldsymbol{\Psi}_i)\boldsymbol{\Phi}_i)$. By substituting (36) and (37) into (34a) and (34b), λ_{ki} and ϕ_i can be calculated.

Note that the primal problem (32) may be infeasible due to the Benders decomposition. To avoid this situation, a penalty term can be introduced and the following problem is solved instead:

$$\min_{\{\boldsymbol{x}_i \in \mathcal{X}_i\}, \{\boldsymbol{f}_i \in \mathcal{F}_i\}, \{\beta_i\}, \{\boldsymbol{\Psi}_i\}} \quad \delta_p = c \tag{38a}$$

$$\text{s.t.} \qquad \boldsymbol{\Psi}_i - \boldsymbol{\Theta}_i(\boldsymbol{x}_i, \boldsymbol{f}_i) + c\boldsymbol{I}_{K+1} \succeq \boldsymbol{0}, \forall i \in \mathcal{U},$$

$$\text{(12c), (23a), (23b).} \tag{38b}$$

The rationality of solving (38) in the infeasible situation is detailed in [24]

The master problem due to the Benders decomposition is given as

Master problem:

$$\min_{\{\boldsymbol{x}_i \in \mathcal{X}_i\}, \delta_m} \quad \delta_m \tag{39a}$$

$$\text{s.t.} \qquad \delta_m \geq \sum_{i \in \mathcal{U}} \tilde{p}_i(\boldsymbol{x}_i^\nu) + \mu_i^\nu (x_i - x_i^\nu) \tag{39b}$$

$$+ \sum_{k \in \mathcal{T}_i} \lambda_{ki}^\nu (x_{ki} - x_{ki}^\nu), \nu \in \mathcal{V}_{fea}^n$$

Algorithm 1: The Proposed Algorithm for Solving (31).

```
Initialize $\epsilon$; $n = 0$; solve (40) to initial $\boldsymbol{x}^0$, $\delta_m^0$
repeat
    if (32) is feasible then
        Solve (32) and acquire $\{\boldsymbol{f}_i^{n+1}\}$ and $\delta_p^{n+1}$
    else if (32) is infeasible then
        Solve (38) and acquire $\{\boldsymbol{f}_i^{n+1}\}$ and $\delta_p^{n+1}$
    end if
    Solve (39) and acquire $\{\boldsymbol{x}_i^{n+1}\}$ and $\delta_m^{n+1}$
    n = n + 1
until $|\delta_m^{n+1} - \delta_p^{n+1}| < \epsilon$
Output $\{\boldsymbol{x}_i\}$ and $\{\boldsymbol{f}_i\}$
```

$$0 \geq c^{\nu} + \sum_{i \in \mathcal{U}} \mu_i^{\nu}(x_i - x_i^{\nu}) + \sum_{k \in \mathcal{T}_i} \lambda_{ki}^{\nu}(x_{ki} - x_{ki}^{\nu}), \nu \in \mathcal{V}_{unf}^n, \quad (12b), \tag{39c}$$

where $\mathcal{V}_{fea}^n$ ($\mathcal{V}_{unf}^n$) denotes the set of feasible (infeasible) iteration numbers. In Benders decomposition, the purpose of solving the master problem is to find suitable $\{\boldsymbol{x}_i \in \mathcal{X}_i\}$, which approaches the optimal solution of (31) iteratively. Therefore, infeasible solutions and nonoptimal solutions should be taken out via solving the master problem. The constraints (39b) and (39c) are Benders cuts and feasibility cuts [23], respectively. For the solutions which do not satisfy (39b), they can be proven to be non-optimal solution of integer variables in (31). Similarly, those solutions which cannot satisfy (39c) can be proven to be unfeasible solutions of those integers. The detailed proof can be found in [24]. When initializing the Benders decomposition algorithm, the following problem is solved [14]

$$\min_{\{\boldsymbol{x}_i \in \mathcal{F}_i\}, \delta_m} \quad \delta_m \tag{40a}$$

$$\text{s.t.} \quad \delta_m \geq \delta_{min}$$

$$(12b), \tag{40b}$$

where δ_{min} is a constant which can be set as a lower bound of the optimization objective of (31). In this paper, δ_{min} can be any negative value. The problem (40) ensure that the initial solution of the integer variables of Benders decomposition locates in the feasible region of (31).

The master problem (39) is a mixed-integer linear programming (MILP) problem, which can be efficiently solved by solvers such as YALMIP [28], the branch-and-bound (BB) algorithm [29] or the simplex method [30]. The proposed Benders decomposition (BD) algorithm for task offloading is summarized in Algorithm 1. Note that the computational complexity of the BD algorithm is $\mathcal{O}(K_i^U)$, which is equal to the computational complexity of solving the master problem.

IV. Simulation Results

In this section, we demonstrate the performance of the proposed BD algorithm. In the numerical Monte-Carlo simulations, 4 FNs are deployed to serve 3 MDs, and each MD can offload its task to any FN. In order to facility the simulation, we assume P_i and $sigma_i^2$ are equal for different MDs, and let $P = 10\log_{10} P_i/\sigma_i^2$. In each simulation, results are averaged over one hundred channel realizations. Main parameters of our simulation are summarized in Table II. The proposed BD algorithm is compared to the existing algorithms without considering the channel estimation errors, which can be summarized as follows:

TABLE II
Main Parameters of Simulation

Notation	Value
K	4
U	3
T	0.05
L_i	0.1 Mb
C_i	1×10^6 cycles
F_k^f	10 GHz
F_i^u	1 GHz
ξ_i	9×10^{-8} W/cycle
σ_i^2	0.05 W
$\mathbb{P}_{\Delta \boldsymbol{G}_i}$	$\mathcal{CN}(0, 0.01\text{diag}(\boldsymbol{G}_i))$
B	1 MHz
ϵ_i	0.05

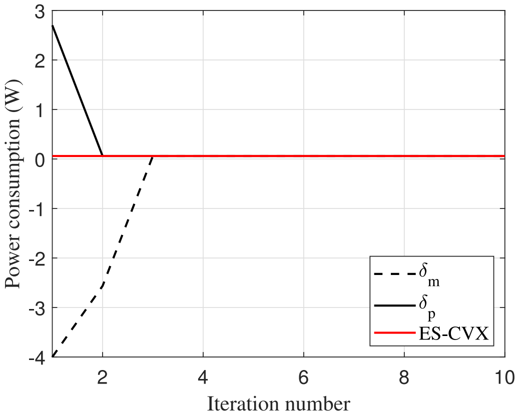


Fig. 2. A convergence example of the BD algorithm when $P = 20$ dB.

- **Exhaustive Search CVX (ES-CVX):** Firstly, we enumerate all $\boldsymbol{x}$ in the feasible region of (22). When $\boldsymbol{x}$ is given, solve (32) by CVX [31] in Matlab. As (32) is a convex optimization problem, CVX can obtain the optimal $\boldsymbol{f}$ and the corresponding value of the objective function in (22). Compare all those values, and we can obtain the global optimal value and the corresponding $(\boldsymbol{x}, \boldsymbol{f})$. The computational complexity of solving the primal problem by CVX is $\mathcal{O}((UK_i)^{3.5})$, and that of enumerating all possible $\boldsymbol{x}$ is $\mathcal{O}(K_i^U)$. Therefore, the computational complexity of the ES-CVX algorithm is $\mathcal{O}(K_i^U(UK_i)^{3.5})$
- **Local Computing (LC):** In this algorithm, all tasks are computed locally.
- **No Robust Benders Decomposition (NRBD):** This algorithm is proposed in [32]. Compared to the proposed BD algorithm, no robust strategy is considered in the NRBD algorithm, which may be subject to the channel estimation errors when offloading the computational tasks.

Fig. 2 shows an example of offloading the computational tasks using the proposed BD algorithm when $P = 20$ dB. The BD

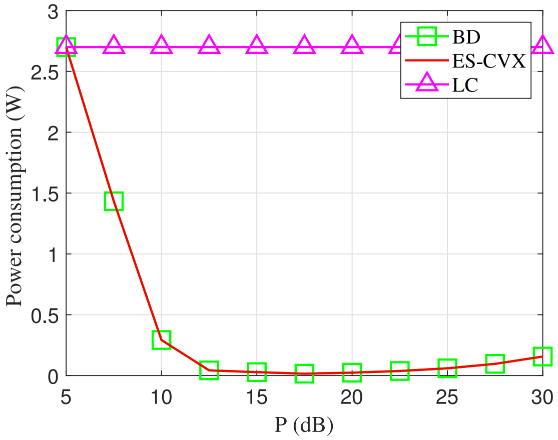


Fig. 3. The power consumption given by different algorithms in terms of different P.

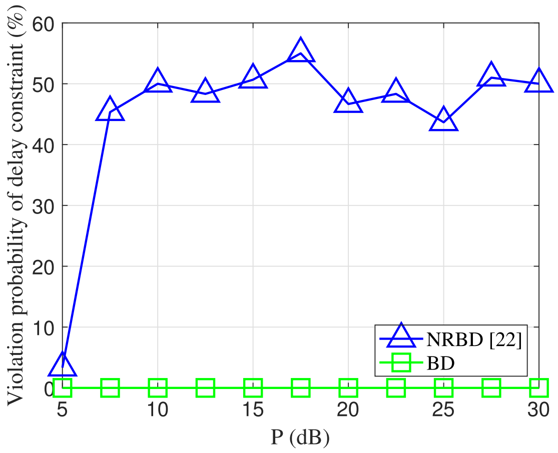


Fig. 5. The violation probability of the delay constraint in terms of different P.

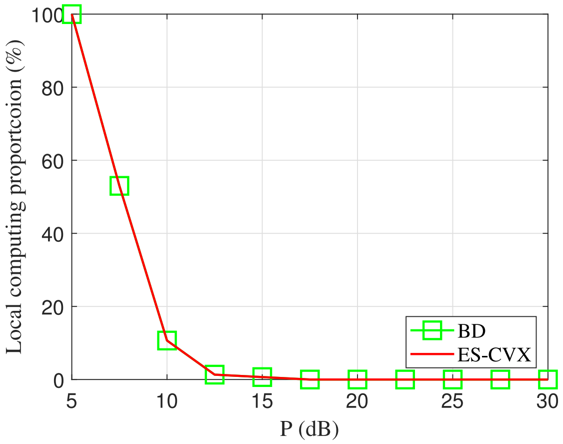


Fig. 4. The proportion of tasks which are computed locally in terms of different P.

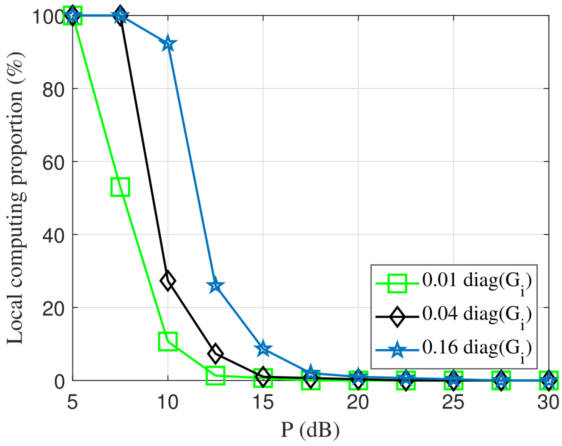


Fig. 6. The proportion of tasks which are computed locally in terms of different $\boldsymbol{\Sigma}_{\Delta G_i}$.

algorithm converges to the optimal power consumption found by the ES-CVX algorithm in only a few iterations. It is also shown that θ_p, namely the value of the objective function of primal problem, is always higher than the optimal value. This phenomenon is reasonable, as (32) can be treated as a special case of (31).

Fig. 3 demonstrates the power consumptions given by different algorithms in terms of different P. It can be observed the the power consumption of LC algorithm is always higher than other algorithms. Moreover, it is also evident that the result given by the BD algorithm can approach the result of the ES-CVX algorithm quite well. When P increases, the power consumptions given by the BD and ES-CVX algorithms increase after an initial decrease. This phenomenon can be explained according to the results demonstrated in Fig. 4.

We detail the proportion of tasks which are computed locally in terms of different P in Fig. 4. Evidently, there is a decreasing tendency when P increases. It is reasonable because the power consumed for task transmission is much less compared to that of local computation. If the channel condition is better (or the transmission power becomes larger), the time consumed for task transmission becomes less, and more tasks tend to be computed in FNs. Therefore, the initial decreasing tendency is reasonable in Fig. 3. When P increases further (e.g. from 20 dB to 30 dB), the power consumption of transmission also increases significantly. Therefore, the power consumption of MDs increases after initial decreasing in Fig. 3. Moreover, it is also evident that the performance of the BD algorithm is equal to that of the ES-CVX algorithm.

In Fig. 5, we demonstrate the violation probability of the delay constraint in terms of different P. As detailed in Table II, we assume the distribution of $\Delta \boldsymbol{G}_i$ is $\mathcal{CN}(0, 0.01\text{diag}(\boldsymbol{G}_i))$. According to Fig. 5, it is clear that when P is low (e.g. 5dB), the violation probabilities of the delay constraint of both algorithms are near-zero. The main reason is that almost all tasks tend to be computed locally when P is low, which is also demonstrated in Fig. 4. When P increases, the violation probability of the delay constraint of the NRBD algorithm is around 50%, as the probability of $\Delta G_{ki} > 0$ or $\Delta G_{ki} < 0$ is 50%. It is also evident that the violation probability of the delay constraint of the BD algorithm is always near-zero, which shows the effectiveness of CVaR.

Fig. 6 demonstrates the proportion of tasks which are computed locally in terms of different $\Sigma_{\Delta G_{ki}}$ if the BD algorithm is adopted. It is evident that when $\Sigma_{\Delta G_{ki}}$ increases, more tasks tend to be computed locally. It is worth mentioning that the delay constraint of each task can be satisfied if it is computed locally. As the uncertainty of the channel estimation may cause the violation of the delay constraint, it is reasonable that more tasks should be computed locally when $\Sigma_{\Delta G_{ki}}$ increases.

V. Conclusion

In this paper, we designed a robust computation offloading strategy where the error of channel estimation was considered. To minimize the power consumption of MDs, the offloading decisions and computing resources are jointly optimized with the latency requirements. With the help of CVaR, we transformed the original problem into an MINLP problem, and solve it by Benders decomposition. The computational complexity of the BD algorithm was much less when compared to the conventional algorithm based on exhaustive searching. Simulation results showed that the proposed BD algorithm can converge quickly, and the results approach the global optimum value quite well with the satisfactory delay constraint.

References

[1] "Cisco visual networking index: Global mobile data traffic forecast update, 2016-2021," Mar. 2017, [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html

[2] Z. Yin *et al.*, "Secrecy rate analysis of satellite communications with frequency domain NOMA," *IEEE Trans. Veh. Technol.*, vol. 68, no. 12, pp. 11847–11858, Dec. 2019.

[3] M. R. Palacin, "Recent advances in rechargeable battery materials: A chemists perspective," *Chem. Soc. Rev.*, vol. 38, no. 9, pp. 2565–2575, 2009.

[4] H. S. Mansouri and V. W. S. Wong, "Hierarchical fog-cloud computing for IoT systems: A computation offloading game," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 3246–3257, Aug. 2018.

[5] C. Mouradian, D. Naboulsi, S. Yangui, R. H. Glitho, M. J. Morrow, and P. A. Polakos, "A comprehensive survey on fog computing: State-of-the-Art and research challenges," *IEEE Commun. Surv. Tuts.*, vol. 20, no. 1, pp. 416–464, Jan.–Mar. 2018.

[6] M. Jia, Z. Yin, D. Li, Q. Guo, and X. Gu, "Toward improved offloading efficiency of data transmission in the IoT-cloud by leveraging secure truncating OFDM," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4252–4261, Jun. 2019.

[7] A. Yousefpour, G. Ishigaki, R. Gour, and J. P. Jue, "On reducing IoT service delay via fog offloading," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 998–1010, Apr. 2018.

[8] 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.

[9] 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.

[10] R. Deng, R. Lu, C. Lai, T. H. Luan, and H. Liang, "Optimal workload allocation in fog-cloud computing toward balanced delay and power consumption," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1171–1181, Dec. 2016.

[11] L. Liu, Z. Chang, and X. Guo, "Socially aware dynamic computation offloading scheme for fog computing system with energy harvesting devices," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1869–1879, Jun. 2018.

[12] A. Ben-Tal and M. Teboulle, "Expected utility, penalty functions and duality in stochastic nonlinear programming," *Manage. Sci.*, vol. 32, pp. 1445–1466, 1986.

[13] R. T. Rockafellar and S. Uryasev, "Optimization of conditional value-at-risk," *J. Risk*, vol. 2, pp. 21–41, Feb. 2000.

[14] A. J. Conejo, E. Castillo, R. Mínguez, and R. G.-Bertrand, *Decomposition Techniques in Mathematical Programming: Engineering and Science Applications*. Berlin, Germany: Springer, 2006.

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

[16] Y. Rong, S. Shahbazpanahi, and A. B. Gershman, "Robust linear receivers for space-time block coded multiaccess MIMO systems with imperfect channel state information," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 3081–3090, Aug. 2005.

[17] Y. Rong, S. Shahbazpanahi, and A. B. Gershman, "Exploiting the structure of OSTBCs to improve the robustness of worst-case opti- mization based linear multi-user MIMO receivers," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Mar. 2005, vol. 4, pp. 781–784.

[18] B. Li, Y. Rong, J. Sun, and K. L. Teo., "A distributionally robust linear receiver design for multi-access space-time block coded MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 16, no. 1, pp. 464–474, Jan. 2017.

[19] B. Li, Y. Rong, J. Sun, and K. L. Teo., "A distributionally robust minimum variance beamformer design," *IEEE Signal Process. Lett.*, vol. 25, no. 1, pp. 105–109, Jan. 2018.

[20] W. Chen, M. Sim, J. Sun, and C. P. Teo, "From CVaR to uncertainty set: Implications in joint chance-constrained optimization," *Oper. Res.*, vol. 58, no. 2, pp. 470–485, Apr. 2010.

[21] S. Zymler, D. Kuhn, and B. Rustem, "Distributionally robust joint chance constraints with second-order moment information," *Math. Program.*, vol. 137, pp. 167–198, Feb. 2013.

[22] A. B.-Tal, L. El Ghaoui, and A. Nemirovski, *Robust Optimization*. Princeton, NJ, USA: Princeton Univ. Press, 2009.

[23] E. Castillo, R. Mínguez, A. J. Conejo, and R. G.-Bertrand, *Decomposition Techniques in Mathematical Programming*. Berlin, Germany: Springer, 2006.

[24] M. J. Bagajewicz and V. Manousiouthakis, "On the generalized Benders decomposition," *Comput. Chem. Eng.*, vol. 15, no. 10, pp. 691–700, Jul. 1991.

[25] S. Guo, B. Xiao, Y. Yang, and Y. Yang, "Energy-efficient dynamic offloading and resource scheduling in mobile cloud computing," in *Proc. IEEE INFOCOM*, Apr. 2016, pp. 1–9.

[26] T. Chen and G. B. Giannakis, "Bandit convex optimization for scalable and dynamic IoT management," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 1276–1286, Feb. 2019.

[27] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[28] J. Lofberg, "YALMIP: A toolbox for modeling and optimization in MATLAB," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2004, pp. 284–289.

[29] E. L. Lawler and D. E. Wood., "Branch-and-bound methods: A survey". *Operations Res.*, vol. 14, no. 4, pp. 699–719, 1966.

[30] J. A. Nelder. "Simplex method for function minimization," *Comput. J.*, vol. 7, 1965.

[31] M. Grant and S. Boyd. "CVX: Matlab software for disciplined convex programming, version 2.0 beta," Apr. 2018. [Online]. Available: http://cvxr.com/cvx

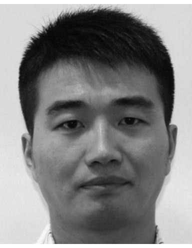
[32] Y. Yu, X. Bu, K. Yang, Z. Wu and Z. Han, "Green large-scale fog computing resource allocation using joint benders decomposition, Dinkelbach algorithm, ADMM, and branch-and-bound," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4106–4117, Jun. 2019.

Zhikun Wu received the B.Sc. degree in information engineering from the Beijing Institute of Technology, Beijing, China, in 2014, where he is currently working toward the Ph.D. degree with the School of Information and Electronics. His research interests include MIMO transceiver design, C-RANs, and ultradense networks.

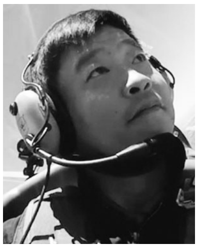
Bin Li received the Ph.D. degree from the School of Information and Electronics, Beijing Institute of Technology, Beijing, China, in 2019. From 2013 to 2014, he was a Research Assistant with the Department of Electronic and Information Engineering, Hong Kong Polytechnic University, Hong Kong. From 2017 to 2018, he was a Visiting Student with the Department of Informatics, University of Oslo, Oslo, Norway. In 2019, he joined the School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, China. His research interests include physical-layer security, wireless cooperative networks, mobile edge computing.

Zesong Fei (Senior Member, IEEE) received the Ph.D. degree in electronic engineering from the Beijing Institute of Technology (BIT), Beijing, China, in 2004. He is the Chief Investigator with the National Natural Science Foundation of China, Beijing, China. He is currently a Professor with the Research Institute of Communication Technology, BIT, where he is involved in the design of the next generation high-speed wireless communication. His research interests include wireless communications and multimedia signal processing. Dr. Fei is the Senior Member of the Chinese Institute of Electronics and the China Institute of Communications.

Zhong Zheng received the B.Eng. degree from the Beijing University of Technology, Beijing, China, in 2007, the M.Sc. degree from the Helsinki University of Technology, Espoo, Finland, in 2010, and the D.Sc. degree from Aalto University, Espoo, Finland, in 2015. From 2015 to 2018, he held visiting positions with The University of Texas at Dallas, Richardson, TX, USA, and National Institute of Standards and Technology, Gaithersburg, MD, USA. In 2019, he joined the School of Information and Electronics, Beijing Institute of Technology, Beijing, China, as an Associate Professor. His research interests include massive MIMO, secure communications, millimeter wave communications, random matrix theory, and free probability theory.



Bin Li (Senior Member, IEEE) received the bachelor’s degree in automation and the master’s degree in control science and engineering from the Harbin Institute of Technology, Harbin, China, in 2005 and 2008, respectively, and the Ph.D. degrees in mathematics and statistics from Curtin University, Bentley, Australia, in 2011. From 2012 to 2014, he was a Research Associate with the School of Electrical, Electronic and Computer Engineering, the University of Western Australia, Crawley, Australia. From 2014 to 2017, he was a Research Fellow with the Department of Mathematics and Statistics, Curtin University. Currently, he is a Research Professor with the College of Electrical Engineering, Sichuan University, Chengdu, China. His research interests include signal processing, wireless communications, optimization, and optimal control.



Zhu Han (Fellow, IEEE) received the B.S. degree in electronic engineering from Tsinghua University, in 1997, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Maryland, College Park, MD, USA, in 1999 and 2003, respectively. From 2000 to 2002, he was an R&D Engineer with JDSU, Germantown, MD, USA. From 2003 to 2006, he was a Research Associate with the University of Maryland. From 2006 to 2008, he was an Assistant Professor with Boise State University, Boise, ID, USA. Currently, he is a Professor with the Electrical and Computer Engineering Department as well as with the Computer Science Department, the University of Houston, Houston, TX, USA. His research interests include wireless resource allocation and management, wireless communications and networking, game theory, big data analysis, security, and smart grid. Dr. Han was the recipient of an NSF Career Award in 2010, the Fred W. Ellersick Prize of the IEEE Communication Society in 2011, the EURASIP Best Paper Award for the Journal on Advances in Signal Processing in 2015, IEEE Leonard G. Abraham Prize in the field of Communications Systems (best paper award in IEEE JSAC) in 2016, and several best paper awards in IEEE conferences. Currently, he is an IEEE Communications Society Distinguished Lecturer.