Joint Optimization of Offloading and Resources Allocation in Secure Mobile Edge Computing Systems

Jun-Bo Wang, *Member, IEEE*, Hui Yang, Ming Cheng, *Student Member, IEEE*, Jin-Yuan Wang, *Member, IEEE*, Min Lin, *Member, IEEE*, and Jiangzhou Wang, *Fellow, IEEE*

***Abstract*—Mobile edge computing (MEC) has become a promising technology for real-time communications. Mobile devices can reduce the energy consumption and prolong the lifetime significantly via offloading the computing tasks to the MEC server. Moreover, physical layer security techniques can ensure the secure transmission of the offloading data. This paper investigates a MEC system that consists of an access point, multiple mobile devices and a malicious eavesdropper. The tasks allocation, local central processor's frequency, offloading power, and offloading timeslots are optimized jointly to minimize the total energy consumption of the system. A difference of convex algorithm based scheme is proposed to solve the joint optimization problem. Moreover, a Karush Kuhn Tucker conditions based algorithm is also proposed to reduce the computational complexity. Numerical results show that the proposed algorithms are very effective. Moreover, the power consumption for secure offloading decreases with the increase of the distance between the mobile devices and the eavesdropper.**

***Index Terms*—Computation offloading, energy consumption, mobile edge computing, physical layer security, resource allocation.**

I. Introduction

Mobile communication has been developing very fast from traditional low data rate voice communications to ultra high speed data communications [1]-[3]. Mobile devices (MDs) are playing an increasingly important role in our daily lives. The new applications on MDs, such as immersive game, depend heavily on real time communications [4]. Although the processing capability of MDs has been greatly improved, it is difficult to provide highly satisfying quality-of-experience (QoE) to execute computing tasks simply on MDs. Mobile computation offloading technology, which takes advantage of abundant resources hosted by clouds, is a promising method to solve a number of concerns affecting mobile computing [5], such as system architecture [6], virtual machine migration [7], and power management [8]. The computation tasks are migrated from MDs to remote cloud servers so that the MDs are released from intensive processing. In general, the mobile computation offloading techniques are applied to networks having a central cloud. Due to finite backhaul capacity and exponentially growing mobile traffic, the central cloud based architecture has disadvantages of high overhead and long backhaul latency [9]. To cope with the stringent requirements of applications on latency, MEC has been proposed. By harvesting the vast amount of the idle computation power and storage space distributed at the network edges, MEC servers have sufficient capacities to perform computation-intensive and latency-critical tasks for MDs [10]. Leveraging the MEC technology, MDs offload part of the computing tasks to the MEC servers. Then, the MEC servers compute these tasks remotely. Since the MEC technology can reduce the time consumption and extend the lifetime of MDs by providing computation service at the edge of mobile network, it has attracted academic and industrial interest [11]-[16].

MEC technology has been extensively studied. For single user, many works have investigated the MEC system for energy consumption [17], [18], latency [19], and QoE [20]. However, there are multiple users in real MEC scenarios. For multiple users, the problems of energy consumption minimization [21], latency minimization [22], and QoE maximization [23] have also been studied. Recently, MEC technology has been widely applied in different network scenarios to minimize the energy consumption and latency of the network, including vehicle networks with cloud-assisted MEC server [24], mobile networks with multiple MEC servers [25], and smart cities based on the Internet of Things [26]. In these works, Lagrange dual method based algorithms have been widely used in MEC systems to minimize the energy consumption and latency [27], [28]. Despite the advantages of MEC technology in reducing energy consumption and time consumption, information security issues emerge in the task

Jun-Bo Wang, Hui Yang ang Ming Cheng are with National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China (email: {jbwang, yang_hui, mingcheng}@seu.edu.cn).

Jin-Yuan Wang and Min Lin are with Key Lab of Broadband Wireless Communication and Sensor Network Technology, Ministry of Education, Nanjing University of Posts and Telecommunications, Nanjing 210003, China (email: {jywang, linmin}@njupt.edu.cn).

Jiangzhou Wang is with the School of Engineering and Digital Arts, University of Kent, Canterbury, Kent, CT2 7NT, United Kingdom (email: j.z.wang@kent.ac.uk).

offloading procedure. Since the broadcast nature of wireless communications, the offloading information in MEC systems are likely to be overheard by nearby eavesdroppers. Therefore, malicious eavesdroppers should be considered in MEC systems. Some techniques should be applied to ensure secure information offloading.

To complete computing tasks successfully, it is critical to ensure secure wireless communications against eavesdropping attacks. A deep learning based model was proposed to learn the features of attacks and to detect actively the eavesdropping attacks in MEC systems [29]. However, the deep learning based method requires large amount of data and can only be implemented offline. Physical layer security technology has been proved to ensure secure transmission in conventional wireless communication networks [30], [31]. In physical layer security, the key design objective is to maximize the secure communication rate. When the transmission rate is less than the secure rate, the eavesdroppers cannot overhear any information [32]. Recently, the physical layer security technique has been exploited for a MEC system where all nodes have a single antenna [33]. However, the researches of physical layer security techniques in MEC systems are deficient. For example, the resource allocation in secure MEC systems has not been thoroughly studied.

To ensure information security in MEC systems, we apply physical layer security technology in MEC systems. We formulate an energy minimization problem in a secure MEC system and propose joint algorithms to optimize offloading and resources allocation. The following highlights our main contributions in this paper.

- We propose a secure MEC system framework in which a multi-antenna access point (AP) has a MEC server and provides computation offloading services for multiple MDs and a multi-antenna malicious eavesdropper is located near the MDs. Each MD adopts partial offloading protocol and completes the computing tasks in a block. To avoid offloading information leakage to the eavesdropper, we propose secure offloading constraints that the offloading rate at each MD cannot exceed its achievable secrecy rate to the AP.
- We develop an optimization problem to minimize the overall energy consumption of the system, subject to the secure offloading constraints and the computing delay constraints. In the proposed problem, the allocations of computing tasks, local central processing unit (CPU) frequency, offloading power and timeslots are optimized jointly.
- Due to the non-convex nature of the secure offloading constraints, the original optimization problem is non-convex. We employ the difference of convex (DCA) method to obtain the iterative convex approximation of the problem. Then, we adopt the Lagrange dual method to derive the solution in a semiclosed form. To reduce the computational complexity further, we also propose a low complexity algorithm.
- Numerical results are given in the performance of the proposed schemes. It is shown that the proposed schemes reduce the energy consumption of the MEC systems significantly. Specifically, when the distance between the MDs and the eavesdropper exceeds a certain distance, the existence of the eavesdropper has no impact on task offloading, and the energy consumption of the computing tasks is stable. Therefore, a long distance between the mobile devices and the eavesdropper can reduce the power consumption for secure offloading.

The remainder of this paper is organized as follows. Section II introduces the system model. In Section III, we formulate the energy consumption minimization problem, which is a joint optimization problem of offloading and resources allocation. Then, we propose the corresponding resource allocation schemes in Section IV. Moreover, the simulation results and discussion are given in Section V, followed by conclusions in Section VI.

II. System Model

As shown in Fig. 1, we consider a MEC system that consists of an AP having a MEC server and N antennas, K single antenna MDs, and a malicious eavesdropper having M antennas. With the partial offloading protocol, MD k ($k \in \mathcal{K} \triangleq \{1, \ldots, K\}$) can divide its computing tasks D_k into two parts: the $l_k \geq 0$ bits that are computed locally and the rest $D_k - l_k \geq 0$ bits that are securely offloaded to the AP and executed by mobile edge computation server.

The system time can be divided into several blocks. The duration of each block T is chosen to be no larger than the latency of the MEC application and also no larger than the channel coherence time. Therefore, the wireless channels can be treated static during each block. To avoid interference among MDs during tasks offloading, each time block is further divided into $2K$ timeslots and the length of the k-th time slot is denoted as t_k ($\forall k \in \{1, 2, \ldots, 2K\}$), as shown in Fig. 2 [34]. In the former K timeslots, the K MDs offload computing tasks to the AP sequentially. Then, in the latter K timeslots, the AP returns the computing results to all MDs successively. Since the MEC server has sufficient computing capability, the computing time of the MEC server is relatively small and negligible [35]. Therefore, we assume that all MDs can download the computing results immediately after the K timeslots offloading. Furthermore, the AP usually has high transmit power and the computed results usually have small size so that the time spent on result feedback is sufficiently short and can be neglected, i.e. $t_k \approx 0, \forall k \in \{K+1, \ldots, 2K\}$ [36], [37].

For MD $k \in \mathcal{K}$, l_k bits of computing tasks should be computed locally within a certain period of time. The computing

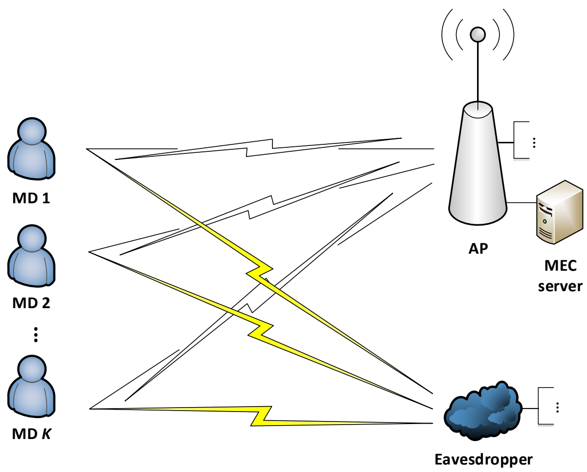

Fig. 1. MEC system model for multi-MDs secure offloading

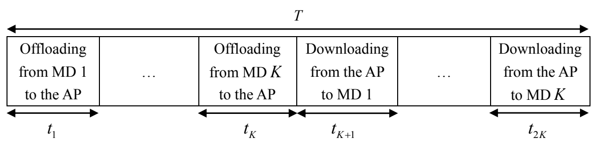

Fig. 2. The timeslots for computation offloading

time cannot exceed the delay constraint τ_k, i.e.,

$$\frac{C_k l_k}{f_k} \leq \tau_k, \forall k \in \mathcal{K}, \tag{1}$$

where C_k is the number of CPU cycles needed for MD k to compute one bit of computing tasks locally, f_k is the CPU frequency of MD k and the value of τ_k cannot be larger than T. Although the dynamic voltage and frequency scaling (DVFS) technology is adopted to control the energy consumption of local computing task by adjusting the CPU frequency [15], the computation capability of MD is not infinite so that the CPU frequency should satisfy

$$0 < f_k \leq f_k^{\max}, \forall k \in \mathcal{K}, \tag{2}$$

where $f_k^{\max}$ denotes the maximum CPU frequency that MD k can achieve.

Due to the limitation of CPU frequency in local computation and its computing tasks, the size of l_k for MD k cannot exceed a certain value, i.e,

$$0 \leq l_k \leq \min\left(D_k, \frac{f_k^{\max}\tau_k}{C_k}\right), \forall k \in \mathcal{K}. \tag{3}$$

The required energy for executing local computing tasks can be expressed as [38]

$$E_k^{\text{loc}} = \xi_k C_k l_k f_k^2, \forall k \in \mathcal{K}, \tag{4}$$

where ξ_k is the effective capacitance coefficient which depends on the chip architecture at MD k.

MD k has $(D_k - l_k)$ bits of computing tasks to be offloaded to the AP. To avoid information leakage to the eavesdropper, the physical layer security technique is adopted in the offloading process. Assuming that the AP employs the maximum ratio combining receiver to decode the information, the achievable secrecy rate is given by [39]

$$R_k = B\left[\log_2\left(1+\frac{\|\mathbf{g}_k\|^2 p_k}{\Gamma\sigma_1^2}\right) - \log_2\left(1+\frac{\|\mathbf{e}_k\|^2 p_k}{\Gamma\sigma_2^2}\right)\right]^+, \forall k \in \mathcal{K}, \tag{5}$$

where $[x]^+ \triangleq \max(x,0)$; B represents the spectrum bandwidth of the offloading channel; $\mathbf{g}_k \in \mathbb{C}^{N\times 1}$ denotes the uplink channel vector from MD k to the AP; $\mathbf{e}_k \in \mathbb{C}^{N\times 1}$ is the channel vector from MD k to the eavesdropper; p_k is the transmit power of MD k for data offloading. σ_1^2 and σ_2^2 are the noise power received by the AP and the eavesdropper, respectively. The coefficient $\Gamma \geq 1$ is an effectiveness factor to adjust the difference with the ideal channel capacity caused by the coding and modulation scheme.

To ensure no offloading information leakage to the eavesdropper, a new secure offloading constraint is required for the offloading rate, that is, the offloading rate of each MD cannot exceed its achievable secrecy rate to the AP. As a result, the offloading rate of MD k to the AP should meet the following inequality

$$\frac{D_k - l_k}{t_k} \leq R_k, \forall k \in \mathcal{K}. \tag{6}$$

Meanwhile, the energy consumption for secure offloading of MD k can be obtained as

$$E_k^{\text{off}} = p_k t_k, \forall k \in \mathcal{K}. \tag{7}$$

The energy consumption of the MEC server is usually proportional to the number of bits offloaded from the MDs, so a simplified linear energy consumption model is adopted for the computing tasks at the MEC server as [16]

$$E_k^{\text{MEC}} = \alpha\left(D_k - l_k\right), \forall k \in \mathcal{K}, \tag{8}$$

where α represents the energy consumption of one bit of computing tasks at the AP. In practice, α depends on the transceiver structure of the AP, the chip structure of the MEC server, and the operated CPU frequency. Since the size of the computing result is usually small, the energy consumption of transferring the computing results to MDs is ignored [36].

III. Problem Formulation

In this paper, the computing capability of the mobile edge cloud is assumed to be infinite. Moreover, the size of computing tasks for all MDs, the CPU cycles needed per bit for local computation, and the energy consumption per bit at the MEC server are all known and can be obtained through feedback. The AP determines the number of offloading bits of each MD and the allocation of computing resources to minimize the weighted sum of the energy consumption of computing tasks, i.e.,

$$E^{\text{sum}} = \sum_{k=1}^{K} \omega_k \left(E_k^{\text{loc}} + E_k^{\text{off}} + E_k^{\text{MEC}}\right), \tag{9}$$

where ω_k represents the weight of MD k in energy consumption, and a large ω_k indicates that MD k has a higher priority. Let $\mathbf{l} \triangleq [l_1, ..., l_K]$, $\mathbf{t} \triangleq [t_1, ..., t_K]$, $\mathbf{P} \triangleq [p_1, ..., p_K]$, and $\mathbf{f} \triangleq [f_1, ..., f_K]$ be the tasks allocation vector for local computing, the timeslots allocation vector for secure offloading, the power allocation vector for secure offloading, and the local CPU frequency allocation vector of all MDs, respectively. Under the constraints of local computing capacity, computational delay, and secure offloading, the joint optimization problem is formulated as

$$\begin{aligned}
\min_{\mathbf{l},\mathbf{t},\mathbf{P},\mathbf{f}} \quad & \sum_{k=1}^{K} \omega_k \left(E_k^{\text{loc}} + E_k^{\text{off}} + E_k^{\text{MEC}}\right) = \\
& \sum_{k=1}^{K} \omega_k \left(\xi_k C_k l_k f_k^2 + p_k t_k + \alpha\left(D_k - l_k\right)\right) && \text{(10a)} \\
\text{s.t.} \quad & \frac{C_k l_k}{f_k} \le \tau_k, \forall k \in \mathcal{K}, && \text{(10b)} \\
& \sum_{k=1}^{K} t_k \le T, \forall k \in \mathcal{K}, && \text{(10c)} \\
& 0 < f_k \le f_k^{\max}, \forall k \in \mathcal{K}, && \text{(10d)} \\
& 0 \le l_k \le \min\left(D_k, \frac{f_k^{\max}\tau_k}{C_k}\right), \forall k \in \mathcal{K}, && \text{(10e)} \\
& p_k \ge 0, \forall k \in \mathcal{K}, && \text{(10f)} \\
& t_k \ge 0, \forall k \in \mathcal{K}, && \text{(10g)} \\
& \frac{D_k - l_k}{t_k} \le R_k, \forall k \in \mathcal{K}. && \text{(10h)}
\end{aligned}$$

In the optimization problem (10), (10b) represents the delay constraints of all MDs for local computing and the value of τ_k cannot be larger than T. (10c) denotes the offloading delay constraints. (10d), (10e), (10f) and (10g) represents the upper and lower bounds of the local CPU frequencies, local computing bits, offloading power, and offloading time of each MD, respectively. (10h) imposes the secure offloading constraints.

Lemma 1: To minimize the energy consumption for local computing, when the number of bits for local computing l_k is given, the optimal local CPU frequency f_k of MD $k \in \mathcal{K}$ in the optimization problem (10) is

$$f_k = \frac{C_k l_k}{\tau_k}, \forall k \in \mathcal{K}. \tag{11}$$

Proof: From (4), the energy consumption for local computing E_k^{loc} is proportional to the size of f_k and l_k. Therefore, it can be derived from (10b) that E_k^{loc} satisfies

$$E_k^{\text{loc}} \ge \frac{\xi_k C_k^3 l_k^3}{\tau_k^2}, \tag{12}$$

which indicates E_k^{loc} takes the minimum value when $f_k = C_k l_k / \tau_k$. ■

Using Lemma 1, we can obtain the optimal local CPU frequency of all MDs in the form of (11). Since the optimal solution of f_k is a function of l_k, we focus on the solution of l_k in the following. The optimization problem (10) can be simplified as

$$\begin{aligned}
\min_{\mathbf{l},\mathbf{t},\mathbf{P}} \quad & \sum_{k=1}^{K} \omega_k \left(E_k^{\text{loc}} + E_k^{\text{off}} + E_k^{\text{MEC}}\right) = \\
& \sum_{k=1}^{K} \omega_k \left(\xi_k \frac{C_k^3 l_k^3}{\tau_k^2} + p_k t_k + \alpha\left(D_k - l_k\right)\right) && \text{(13a)} \\
\text{s.t.} \quad & \sum_{k=1}^{K} t_k \le T, \forall k \in \mathcal{K}, && \text{(13b)} \\
& 0 \le l_k \le \min\left(D_k, \frac{f_k^{\max}\tau_k}{C_k}\right), \forall k \in \mathcal{K}, && \text{(13c)} \\
& p_k \ge 0, \forall k \in \mathcal{K}, && \text{(13d)} \\
& t_k \ge 0, \forall k \in \mathcal{K}, && \text{(13e)} \\
& \frac{D_k - l_k}{t_k} \le R_k, \forall k \in \mathcal{K}. && \text{(13f)}
\end{aligned}$$

Due to the non-convex nature of constraint (13f), problem (13) is non-convex in the current form. However, it can be transformed into a convex form by the appropriate convex approximation method, which is shown in the next section.

IV. Proposed Joint Optimization Schemes

In this section, two schemes are proposed to slove the problem (13). The first scheme is a joint application of DCA and Lagrange dual method, which gives a sub-optimal solution of the problem (13). The second scheme is a direct application of the Karush Kuhn Tucker (KKT) conditions and has a lower complexity compared to the first scheme.

A. Joint Application of DCA and Lagrange Dual Method

Although the objective function in the problem (13) is affine, its constraint (13f) is non-convex, which results in that the problem (13) is a nonconvex optimization problem. To facilitate the following description, let $\mathbf{x}_k \triangleq (l_k, p_k, t_k)$. Then, by using (5), we can derive (13f) as

$$f\left(\mathbf{x}_k\right) - g\left(\mathbf{x}_k\right) \le 0, \forall k \in \mathcal{K}, \tag{14}$$

where $f\left(\mathbf{x}_k\right)$ and $g\left(\mathbf{x}_k\right)$ are

$$f\left(\mathbf{x}_k\right) = D_k - l_k - t_k B \log_2\left(1 + \left\|\mathbf{g}_k\right\|^2 p_k\right) \tag{15}$$

and

$$g\left(\mathbf{x}_k\right) = -t_k B \log_2\left(1 + \left\|\mathbf{e}_k\right\|^2 p_k\right), \tag{16}$$

respectively. Obviously, $f\left(\mathbf{x}_k\right)$ and $g\left(\mathbf{x}_k\right)$ are convex functions and the left expression of (14) is a form of subtraction of two convex functions. The optimization problems involving difference of convex functions (DC) are called DC programming problems [40], which can be solved efficiently by the DCA method.

Using the DCA method, we replace the (16) with its first order Taylor expansion around a feasible point $\mathbf{x}_k^n = (l_k^n, p_k^n, t_k^n)$ at the $(n+1)$-th iteration, i.e.,

$$\hat{g}\left(\mathbf{x}_k; \mathbf{x}_k^n\right) \triangleq g\left(\mathbf{x}_k^n\right) + \nabla g(\mathbf{x}_k^n)^T(\mathbf{x}_k - \mathbf{x}_k^n), k=1,2,....,K. \tag{17}$$

Then, (14) can be approximated as

$$f\left(\mathbf{x}_k\right) - \hat{g}\left(\mathbf{x}_k; \mathbf{x}_k^n\right) \le 0, \forall k \in \mathcal{K}, \tag{18}$$

which is further rewritten in detail as

$$D_k - l_k - t_k B \log_2\left(1+\|\mathbf{g}_k\|^2 p_k\right) + t_k B \log_2\left(1+\|\mathbf{e}_k\|^2 p_k^n\right) + \frac{t_k^n B \|\mathbf{e}_k\|^2}{\left(1+\|\mathbf{e}_k\|^2 p_k^n\right)\ln 2}\left(p_k - p_k^n\right) \le 0, \forall k \in \mathcal{K}. \tag{19}$$

Accordingly, the original non-convex optimization problem (13) can be transformed into a new optimization problem at the $(n+1)$-th iteration as

$$\min_{\mathbf{l},\mathbf{t},\mathbf{P}} \sum_{k=1}^{K} \omega_k \left(E_k^{\text{loc}} + E_k^{\text{off}} + E_k^{\text{MEC}}\right) = \sum_{k=1}^{K} \omega_k \left(\xi_k \frac{C_k^3 l_k^3}{\tau_k^2} + p_k t_k + \alpha\left(D_k - l_k\right)\right), \tag{20}$$

subject to (13b), (13c), (13d), (13e) and (19).

Using the DCA method, Algorithm 1 is proposed to solve the optimization problem (13). Its convergence has been discussed in previous works [41], [42].

Algorithm 1 Solving the problem (13) by the DCA method (DCA scheme)

1: Choose $\mathbf{x}_k^0 = \left(l_k^0, t_k^0, p_k^0\right) (\forall k \in \mathcal{K})$ as an initial feasible point and set $n = 0$.
2: **repeat**
3: Use the feasible point $\mathbf{x}_k^n$ to update the constraints (19) for all $k \in \mathcal{K}$;
4: Solve the optimization problem (20) and obtain the solution as $\mathbf{x}_k^{n+1} (\forall k \in \mathcal{K})$;
5: Update the number of iterations: $n = n + 1$.
6: **until**
7: The solution converges and output the solutions as l_k^{opt}, p_k^{opt} and t_k^{opt}.

Lemma 2: For any optimization problem with a convex feasible set, if the constraint function is differentiable and can be written as the difference of two convex functions, the KKT point can be obtained by the DCA method.

Proof: See Appendix B. ■

Lemma 2 indicates that Algorithm 1 will converge to at least one KKT point. In the following, we will discuss how to solve the problem (20) by using the Lagrange dual technique [43]. The Lagrangian function of (20) is given as

$$L\left(\mathbf{l},\mathbf{t},\mathbf{P},\lambda,\mu\right) = \sum_{k=1}^{K} \omega_k \left(\xi_k \frac{C_k^3 l_k^3}{\tau_k^2} + p_k t_k + \alpha\left(D_k - l_k\right)\right) + \mu\left(\sum_{k=1}^{K} t_k - T\right) + \sum_{k=1}^{K} \lambda_k \left(f\left(\mathbf{x}_k\right) - \hat{g}\left(\mathbf{x}_k; \mathbf{x}_k^n\right)\right), \tag{21}$$

where $\mu \ge 0$ and $\lambda_k \ge 0 (\forall k \in \mathcal{K})$ denotes the Lagrange multipliers associated with the delay constraint for offloading in (13b) and the k-th secure offloading constraint in (19), respectively. Let $\lambda \triangleq [\lambda_1, ..., \lambda_K]^\dagger$. Then, the corresponding dual function is given as

$$\Phi\left(\lambda,\mu\right) = \min_{\mathbf{l},\mathbf{t},\mathbf{P}} L\left(\mathbf{l},\mathbf{t},\mathbf{P},\lambda,\mu\right) \tag{22a}$$

$$\text{s.t. } 0 \le l_k \le \min\left(D_k, f_k^{\max} T / C_k\right), p_k \ge 0, t_k \ge 0, \forall k \in \mathcal{K}. \tag{22b}$$

Therefore, the dual problem of (20) is

$$\max_{\lambda,\mu} \Phi\left(\lambda,\mu\right) \tag{23a}$$

$$\text{s.t. } \mu \ge 0, \lambda_k \ge 0, \forall k \in \mathcal{K}. \tag{23b}$$

Using the dual decomposition theory [44], the Lagrange dual problem (23) can be decomposed into two levels of optimization. At the lower level, the problem (22) can be divided into $2K$ subproblems as follow. The first K subproblems are used to optimize $l_k \, (k \in \mathcal{K})$ as

$$\min_{l_k} \omega_k \xi_k \frac{C_k^3 l_k^3}{\tau_k^2} - \omega_k \alpha l_k - \lambda_k l_k \tag{24a}$$

$$\text{s.t. } 0 \le l_k \le \min\left(D_k, f_k^{\max} T / C_k\right). \tag{24b}$$

Meanwhile, the remaining K subproblems are used to jointly optimize p_k and $t_k \, (k \in \mathcal{K})$ as

$$\min_{p_k, t_k} \omega_k p_k t_k + \mu t_k - \lambda_k B \times\left(t_k \log_2\left(\frac{1+\|\mathbf{g}_k\|^2 p_k}{1+\|\mathbf{e}_k\|^2 p_k^n}\right) - \frac{t_k^n \|\mathbf{e}_k\|^2 p_k}{\left(1+\|\mathbf{e}_k\|^2 p_k^n\right)\ln 2}\right) \tag{25a}$$

$$\text{s.t. } p_k \ge 0, t_k \ge 0. \tag{25b}$$

Since the subproblems in (24) and (25) are convex and satisfy the Slater condition, the optimal solution (l_k^*, t_k^*, p_k^*) $(\forall k \in \mathcal{K})$ for any given $(\lambda, \mu) \in S$ can be obtained by Theorem 1 and Theorem 2.

Theorem 1: For any given $(\lambda, \mu) \in S$ in (24) , the optimal solution of task allocation for local computation is given as

$$l_k^* = \min\left(\sqrt{\frac{\left(\omega_k \alpha + \lambda_k\right)\tau_k^2}{3\omega_k \xi_k C_k^3}}, D_k, \frac{f_k^{\max}\tau_k}{C_k}\right), \forall k \in \mathcal{K}. \tag{26}$$

Proof: By deriving the first-order derivative of the k-th subproblem in (24) with respect to l_k, the following equation can be obtained as

$$\frac{\partial L}{\partial l_k^*} = \frac{3\omega_k \xi_k C_k^3 {l_k^*}^2}{\tau_k^2} - \omega_k \alpha - \lambda_k = 0. \tag{27}$$

Then, the optimal solution of bit allocation for local computation can be obtained in (26). ■

Theorem 2: For any given $(\lambda,\mu)\in S$ in (25), the optimal solution of power allocation is given as

$$p_k^* = \begin{cases} 0, & \text{if } \|\mathbf{g}_k\|^2 \le \|\mathbf{e}_k\|^2 \\ \left[\frac{\lambda_k B}{\omega_k \ln 2} - \frac{1}{\|\mathbf{g}_k\|^2}\right]^+, & \text{if } \|\mathbf{g}_k\|^2 > \|\mathbf{e}_k\|^2 \end{cases}, \forall k\in\mathcal{K}, \quad (28)$$

and the optimal solution of timeslots allocation is derived as

$$t_k^* = \frac{D_k - l_k^*}{R_k^*}, \forall k\in\mathcal{K}, \quad (29)$$

where

$$R_k^* = B\left[\log_2 \frac{1+\|\mathbf{g}_k\|^2 p_k^*}{1+\|\mathbf{e}_k\|^2 p_k^*}\right]^+, \forall k\in\mathcal{K}. \quad (30)$$

Proof: Deriving the first-order derivatives of the k-th subproblem in (25) with respect to p_k and t_k, respectively, we obtain

$$\frac{\partial L}{\partial p_k^*} = \omega_k t_k^* - \frac{\lambda_k t_k^* B\|\mathbf{g}_k\|^2}{\left(1+\|\mathbf{g}_k\|^2 p_k^*\right)\ln 2} + \frac{\lambda_k t_k^n B\|\mathbf{e}_k\|^2}{\left(1+\|\mathbf{e}_k\|^2 p_k^n\right)\ln 2} = 0, \quad (31)$$

$$\frac{\partial L}{\partial t_k^*} = \omega_k p_k^* + \mu - \lambda_k B \log_2\left(\frac{1+\|\mathbf{g}_k\|^2 p_k^*}{1+\|\mathbf{e}_k\|^2 p_k^n}\right) = 0. \quad (32)$$

Then, the optimal solution of power allocation can be derived in (28).

Due to the secure offloading constraint, the offloading rate of MD cannot exceed its achievable secrecy rate to the AP, i.e., $r_k \le R_k\,(\forall k\in\mathcal{K})$, therefore, the offloading time of MD satisfies $t_k = (D_k - l_k)/r_k \ge (D_k - l_k)/R_k(\forall k\in\mathcal{K})$. To minimize the energy consumption for offloading at MD k, we derive the optimal solution of timeslots allocation as (29). ■

At the upper level, the master problem in charge of updating the Lagrange multipliers is shown as (33), where l_k^*, p_k^* and t_k^* $(\forall k\in\mathcal{K})$ can be obtained in (26), (28) and (29), respectively. The dual function $\Phi(\lambda,\mu)$ is usually concave but not differential. Therefore, a method based on subgradient such as ellipsoid method can be used to solve the problem (33) to update (λ,μ). The ellipsoid method is an iterative method for solving convex problems, which is guaranteed to converge in polynomial time [45]. With the ellipsoid method, a dual-maximization-based joint resource-allocation algorithm is proposed in Algorithm 2.

Algorithm 2 Solving the optimization problem (20) at the $(n+1)$-th iteration in Algorithm 1

Choose the initial Lagrange multipliers $\left(\lambda^0,\mu^0\right)$ and set $m=0$.
repeat
  Let $\lambda_k=\lambda_k^m$ and $\mu=\mu^m$, obtain the optimal values l_k^*, p_k^* and t_k^* using (26), (28) and (29), respectively.
  Substitute l_k^*, p_k^* and t_k^* into the optimization problem (33) and solve it to obtain $\left(\lambda^{m+1},\mu^{m+1}\right)$ using the ellipsoid method;
  Update the number of iterations: $m=m+1$.
until
(λ,μ) converges and output l_k^*, p_k^* and t_k^*.

B. Low Complexity Algorithm

Since the joint application of DCA and Lagrange dual method requires two levels of iterations, it needs a large amount of computations to solve the optimization problem (13). To reduce the computation complexity, the KKT conditions are considered to directly solve the problem (13) in a semiclosed form. Let $\lambda_k \ge 0\,(\forall k\in\mathcal{K})$ and $\mu\ge 0$ represent the Lagrangian multiplier associated with the time allocation constraint in (10c) and the secure offloading constraint in (10h), respectively. Then, the Lagrangian function of (13) can be written as

$$U(\mathbf{l},\mathbf{t},\mathbf{P},\lambda,\mu) = \sum_{k=1}^{K}\omega_k\left(\xi_k\frac{C_k^3 l_k^3}{\tau_k^2} + p_k t_k + \alpha(D_k - l_k)\right) + \sum_{k=1}^{K}\lambda_k(D_k - l_k - R_k t_k) + \mu\left(\sum_{k=1}^{K} t_k - T\right). \quad (34)$$

With (34), the KKT conditions can be derived to obtain

$$\frac{\partial U}{\partial l_k^+} = \frac{3\omega_k\xi_k C_k^3 {l_k^+}^2}{\tau_k^2} - \omega_k\alpha - \lambda_k^+ \begin{cases} >0, & l_k^+ = 0 \\ =0, & l_k^+\in(0,D_k) \\ <0, & l_k^+ = D_k \end{cases}, \quad (35)$$

$$\frac{\partial U}{\partial t_k^+} = \omega_k p_k^+ + \mu^+ - \lambda_k^+ B R_k^+ \begin{cases} >0, & t_k^+ = 0 \\ =0, & t_k^+ > 0 \end{cases}, \quad (36)$$

$$\frac{\partial U}{\partial p_k^+} = \omega_k t_k^+ - \lambda_k^+ t_k^+ B\frac{\partial R_k^+}{\partial p_k^+} \begin{cases} >0, & p_k^+ = 0 \\ =0, & p_k^+ > 0 \end{cases}, \quad (37)$$

where $\left(l_k^+, t_k^+, p_k^+\right)(\forall k\in\mathcal{K})$ denotes the optimal solution of KKT conditions, $\left\{\lambda_k^+\right\}(\forall k\in\mathcal{K})$ and μ^+ denote the optimal value of the Lagrangian multipliers. From (35)-(37), Theorem 3 can be derived as follows.

Theorem 3: The optimal solutions of KKT conditions can be derived as

$$l_k^+ = \begin{cases} D_k, & l_k^+ = D_k \\ \sqrt{\frac{\left(\omega_k\alpha + \lambda_k^+\right)\tau_k^2}{3\omega_k\xi_k C_k^3}}, & l_k^+\in(0,D_k) \\ 0, & l_k^+ = 0 \end{cases}, \forall k\in\mathcal{K}, \quad (38)$$

$$p_k^+ = \begin{cases} 0, & l_k^+ = D_k \\ \left(\frac{\Delta - \left(\|\mathbf{g}_k\|^2 + \|\mathbf{e}_k\|^2\right)}{2\|\mathbf{g}_k\|^2\|\mathbf{e}_k\|^2}\right)^+, & l_k^+ \in [0, D_k) \end{cases}, \forall k\in\mathcal{K}, \quad (39)$$

$$t_k^+ = \frac{D_k - l_k^+}{R_k^+}, \forall k\in\mathcal{K}, \quad (40)$$

$$\max_{\lambda,\mu}\ \sum_{k=1}^{K}\lambda_k\left(D_k-\frac{t_k^n B\|\mathbf{e}_k\|^2 p_k^n}{\left(1+\|\mathbf{e}_k\|^2 p_k^n\right)\ln 2}\right)-\mu T+\sum_{k=1}^{K}\left(\omega_k\xi_k\frac{C_k^3 {l_k^*}^3}{\tau_k^2}-\omega_k\alpha l_k^*-\lambda_k l_k^*\right)$$
$$+\sum_{k=1}^{K}\left(\omega_k p_k^* t_k^*+\mu t_k^*-\lambda_k B\left(t_k^*\log_2\left(\frac{1+\|\mathbf{g}_k\|^2 p_k^*}{1+\|\mathbf{e}_k\|^2 p_k^n}\right)-\frac{t_k^n\|\mathbf{e}_k\|^2 p_k^*}{\left(1+\|\mathbf{e}_k\|^2 p_k^n\right)\ln 2}\right)\right) \tag{33}$$

where

$$\Delta=\sqrt{\left(\|\mathbf{g}_k\|^2-\|\mathbf{e}_k\|^2\right)^2+\frac{4\lambda_k^+ B\|\mathbf{g}_k\|^2\|\mathbf{e}_k\|^2\left(\|\mathbf{g}_k\|^2-\|\mathbf{e}_k\|^2\right)}{\omega_k\ln 2}}, \tag{41}$$

$$R_k^+=B\left[\log_2\frac{1+\|\mathbf{g}_k\|^2 p_k^+}{1+\|\mathbf{e}_k\|^2 p_k^+}\right]^+,\forall k\in\mathcal{K}. \tag{42}$$

Proof:

In $l_k^+=D_k$ case, $\partial U/\partial l_k^+<0$ can be obtained by (35), which indicates that all computing tasks are computed locally without offloading. Therefore $p_k^+=0$ and $t_k^+=0$ can be obtained easily.

In $l_k^+\in(0,D_k)$ case, $\partial U/\partial l_k^+=0$ can be obtained from (35), then the solution of bit allocation for local computation at each MD k can be given in (38). Since part of the computing tasks will be offloaded, i.e., $p_k^+>0$ and $t_k^+>0$, $\partial R_k^+/\partial p_k^+=\omega_k/\lambda_k^+ B$ can be obtained from (37), and it is known that R_k^+ is a function of p_k^+ from (42), which indicates that $R_k^+=R\left(P_k^+\right)$. Therefore, p_k^+ can be given in (39). Combined with theorem 2, t_k^+ can be obtained in (40).

In $l_k^+=0$ case, the expressions of p_k^+ and t_k^+ are the same as above case when $l_k^+\in(0,D_k)$. ■

It should be noted that the solutions of (38), (39), and (40) are only functions of λ_k^+. Further, the value range of λ_k^+ can be derived from (35) as following theorem.

Theorem 4: The range of λ_k^+ can be derived as

$$0\le\lambda_k^+\le\frac{3\omega_k\xi_k C_k^3 {l_k^+}^2}{\tau_k^2}-\omega_k\alpha<\frac{3\omega_k\xi_k C_k^3 D_k^2}{\tau_k^2}-\omega_k\alpha,\forall k\in\mathcal{K}, \tag{43}$$

when $l_k^+\in[0,D_k)$. For facilitate the following description, $\lambda_k^{\max}$ can be set as $3\omega_k\xi_k C_k^3 D_k^2/\tau_k^2-\omega_k\alpha$.

Proof: $\partial U/\partial l_k^+=3\omega_k\xi_k C_k^3 {l_k^+}^2\Big/\tau_k^2-\omega_k\alpha-\lambda_k^+\ge 0$ can be obtained from (35) when $l_k^+\in[0,D_k)$. Therefore, the value range of λ_k^+ can be derived by simple mathematical transformation. ■

Using Theorem 4, the approximate solution can be obtained by Algorithm 3.

C. Complexity Analysis

The complexity of Algorithm 1 comes from two aspects. The first aspect is from the DCA method for updating the constraints (19). The second aspect is from the the complexity of Algorithm 2. The complexity of Algorithm 2 comes from the ellipsoid method for computing the Lagrange multipliers and the application of CVX for solving the dual problem. Let L_1 and L_2 denote the number of iterations required for Algorithm 1 and Algorithm 2, respectively. Let ρ denote the tolerance error for the ellipsoid method. Thus, according to the works in [36] and [50], the total complexity of Algorithm 2 is $O\left(1/\rho^2+L_2K^3\right)$. The total complexity of Algorithm 1 is $O\left(L_1K\left(1/\rho^2+L_2K^3\right)\right)$.

Algorithm 3 Low complexity scheme

Set $\lambda_k^l=0$ and $\lambda_k^h=\lambda_k^{\max}$.
while $\lambda_k^h-\lambda_k^l\ge\varepsilon$ **do**
  calculate $\lambda_k^m=\left(\lambda_k^l+\lambda_k^h\right)/2$ and $\sum_{k=1}^{K}t_k$ by (40) for the given λ_k^m;
  if $\sum_{k=1}^{K}t_k>T$ **then**
    let $\lambda_k^l=\lambda_k^m$;
  else
    let $\lambda_k^h=\lambda_k^m$;
  end if
end while
Substitute λ_k^h into (38), (39) and (40) to obtain the solutions l_k^+, p_k^+ and t_k^+, respectively.

The complexity of Algorithm 3 comes from the bisection method for obtaining the optimal solution. This sub-optimal algorithm has low computation complexity. Specifically, let d denote the largest bisection-search interval. Given a solution accuracy $\varepsilon>0$, the bisection method will call for $\log_2(d/\varepsilon)$ times of comparison operations, and thus has the order of complexity $\log(1/\varepsilon)$. For each iteration, the resource-allocation complexity is $O(K)$. Therefore, the total computation complexity of algorithm 3 is $O\left(K\log\left(1/\varepsilon\right)\right)$.

V. Numerical Results

In this section, we evaluate the performance of the proposed algorithms. The number of users accessing the AP simultaneously in the MEC system will not be large due to the limitation of communication and computing resources [35], [46], [47]. In the simulated MEC system, the numbers of antennas at the AP and the eavesdropper are set to 8 and 2, respectively. We assume that the spectral bandwidth for offloading is $B=2$ GHz and the noise power at the receiver is $\sigma^2=10^{-10}$ W. Moreover, the effective capacitance coefficient is $\xi_k=10^{-28}$, the number of CPU cycles required per bit for local computing is $C_k = 10^3$ cycles/bit, and the energy consumption per bit at the MEC server is α=10^{-4} J/bit [16]. The path loss of the channel is modelled as $PL\,(dB) = 10\alpha\log_{10}(d) + \beta + X_\sigma$, where α=1.08, β=70.9, d represents the distance between the nodes, and X_σ represents the shadow fading portion, and σ=$2.5dB$ [48]. Without loss of generality, it is assumed that the distances from all MDs to the AP are identical and the distances from all MDs to the eavesdropper are the same. Moreover, all MDs have the same sizes of computing tasks. To facilitate the following descriptions, the distances from MDs to the AP and the eavesdropper are named as the offloading distance and the leaking distance, respectively.

To evaluate the performance of the proposed algorithms, two benchmark schemes are considered for performance comparison.

1) Local computing scheme: All MDs complete the overall computing tasks locally, i.e. $l_k = D, \forall k \in \mathcal{K}$. Then, the weighted sum of the energy consumption of computing tasks is simplified as

$$\sum_{k=1}^{K} \frac{\omega_k \xi_k C_k^3 D^3}{T^2}. \tag{44}$$

2) Full offloading scheme: All MDs choose to offload the overall computing tasks to the AP. Accordingly, the weighted sum of the energy consumption of computing tasks is equivalent to the solution of problem (10) by setting $l_k = 0, \forall k \in \mathcal{K}$.

Both Fig. 3 and Fig. 4 show the average energy consumption versus the size of computing tasks. The numbers of MDs in Fig. 3 and Fig. 4 are 5 and 20, respectively. Moreover, the offloading distance is 20 m, the leaking distance is 20 m, and the duration of blocks is 10 ms. In Fig. 3, it is observed that the average energy consumption of all schemes increases with the increase of the size of computing tasks. It should be noted that the proposed DCA scheme obtains the lowest average energy consumption and the gap between the proposed DCA scheme and the proposed low complexity scheme is quite small. In particular, when the size of computing tasks is small, the local computing scheme outperforms the full offloading scheme, and approximates to the proposed DCA scheme. This is because the local computing capacity of MD is sufficient to complete most computing tasks while satisfying the delay constraints. However, when the size of computing tasks is large, the local MDs cannot complete computing tasks independently within the delay constraints. Then, offloading more computing tasks to the AP can guarantee the delay constraints while keeping low energy consumption. For this reason, when the size of computing tasks increases, the performance of the full offloading scheme gradually approaches that of the proposed DCA scheme, which means that nearly all computing tasks of MDs have been offloaded to the AP. Moreover, it is observed that Fig. 3 has the similar insight to Fig. 4. Therefore, the proposed schemes can also be applied to MEC systems having more users.

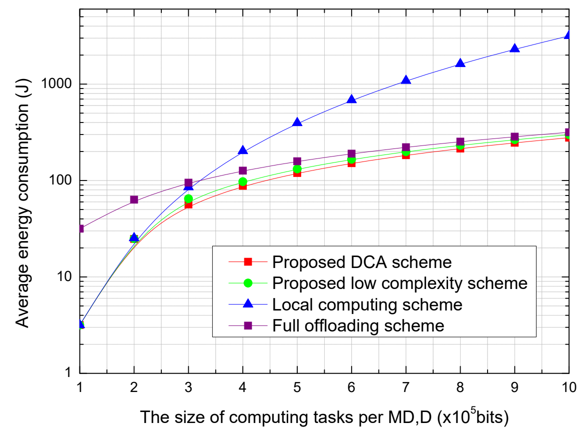

Fig. 3. The average energy consumption versus the size of computing tasks (5 MDs)

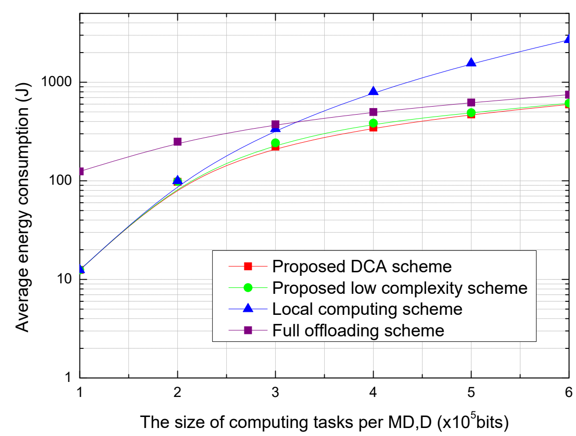

Fig. 4. The average energy consumption versus the size of computing tasks (20 MDs)

Figure 5 shows the average energy consumption versus the leaking distance, where the number of MDs is 5, the size of computing tasks is 3×10^5 bits, the offloading distance is 20 m, and the duration of blocks is 10 ms. It can be observed that the proposed DCA scheme obtains the lowest average energy consumption. The gap between the proposed DCA scheme and the proposed low complexity scheme decreases with the increase of the leaking distance. Particularly, the average energy consumption of the local computing scheme does not change with the leaking distance. Because all computing tasks of MDs are computed locally, and no power consumption is needed for offloading. However, the capacity of leaking channels decreases with the increase of the leaking distance, which results in the decrease of the average power consumption for

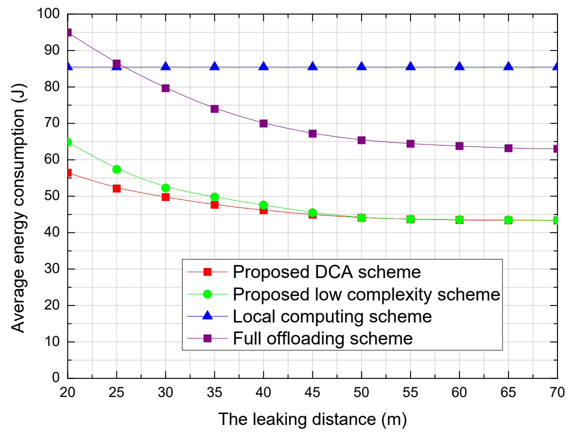

Fig. 5. The average energy consumption versus the leaking distance

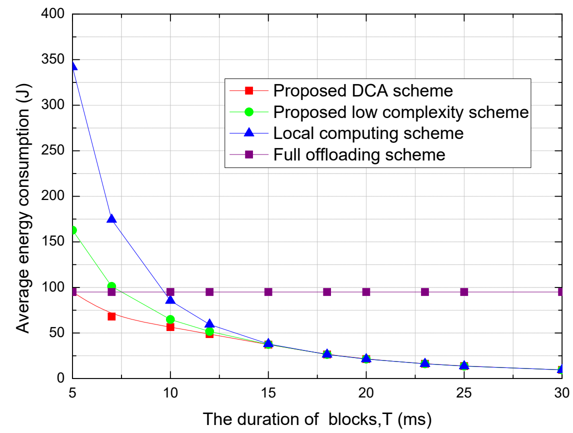

Fig. 6. The average energy consumption versus the duration of blocks

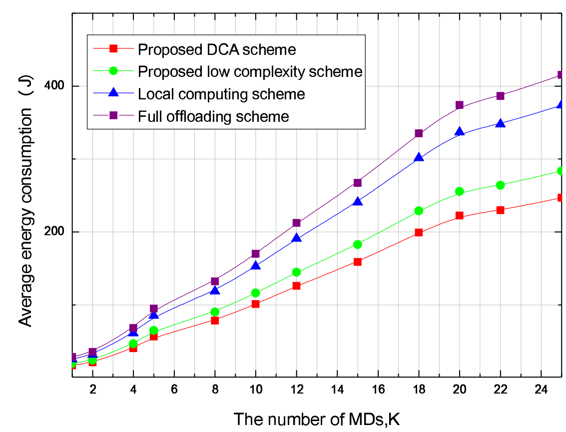

Fig. 7. The average energy consumption versus the number of MDs

the full offloading scheme. the proposed DCA scheme and the proposed low complexity scheme. Moreover, when the leaking distance exceeds a certain value, the average power consumption of the full offloading scheme, the proposed DCA scheme, and the proposed low complexity scheme does not change with the leaking distance. Then, the capacities of leaking channels are too small for the eavesdropper to steal the offloading data.

Figure 6 shows the average energy consumption versus the duration of blocks, where the number of MDs is 5, the size of computing tasks is 3×10^5 bits , the offloading distance is 20 m, and the leaking distance is 20 m. It can be observed that the proposed DCA scheme requires the lowest average energy consumption and the gap between the proposed DCA scheme and the proposed low complexity scheme decreases with the increase of the duration of blocks. Specially, the average energy consumption of the full offloading scheme remains stable with the delay constraint. Because the increase of the delay constraint only reduces the CPU frequency of the local MDs, which indicates that the increase of the duration of blocks reduces the average energy consumption of all schemes except the full offloading scheme. Moreover, when the duration of blocks is sufficiently large, the local computing scheme and the proposed two schemes obtain the approximately same performance and outperform the full offloading scheme significantly. The reason is that when the delay constraints are not strict, nearly all computing tasks can be done locally, the power consumption for wireless transmission of data offloading greatly will be further saved.

Figure 7 shows the average energy consumption versus the number of MDs, where the size of computing tasks is 3×10^5 bits, the offloading distance is 20 m, the leaking distance is 20 m, and the duration of blocks is 10 ms. It is observed that the average energy consumption of all schemes increases with the number of MDs. It can be noted that the proposed DCA scheme requires the lowest average energy consumption. The gap between the proposed DCA scheme and the proposed low complexity scheme is small. Moreover, the energy consumption of the full offloading scheme is slightly higher than that of the local computing scheme and the performance gain obtained by the proposed DCA scheme is significant.

Figure 8 shows the offloading rate and the achievable secrecy rate versus the size of computing tasks, where the number of MDs is 1, the offloading distance is 20 m, the leaking distance is 20 m, and the duration of blocks is 10 ms. It is observed that the offloading rate keeps lower than the achievable secrecy rate with the increase of the size of computing tasks. It is indicated that the computation offloading of each MD strictly adheres to the secure offloading constraint proposed in this paper. In particular, the gap between the offloading rate and the achievable secrecy rate decreases with the increase of the size of computing tasks. The reason is

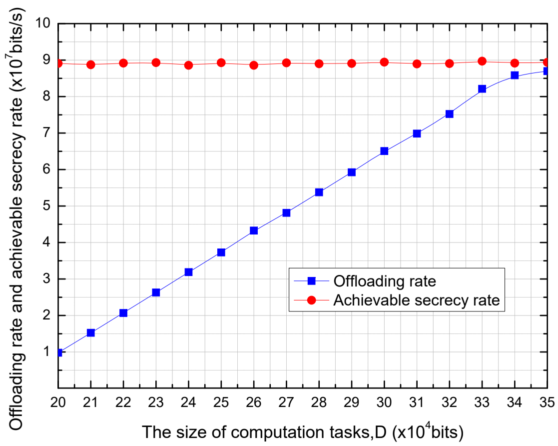

Fig. 8. Offloading rate and achievable secrecy rate versus the size of computing tasks

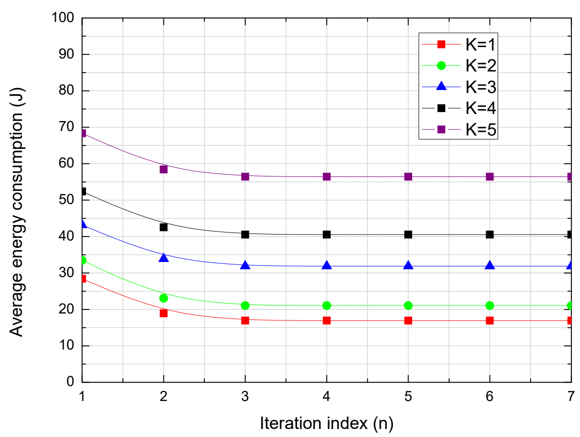

Fig. 9. The convergence of the proposed DCA scheme

that MDs tend to offload more computing tasks to the AP to reduce the energy consumption as the size of computing tasks increases, a higher offloading rate is needed to meet the delay constraints.

Figure 9 shows the convergence performance of the proposed DCA scheme for different numbers of MDs. It can be seen from Fig. 9 that irrespective of the number of MDs, the average energy consumption decreases very quickly and converges within 3 iterations. It is indicated that the proposed DCA scheme will obtain fairly good solutions within a very short computation time in practical systems.

VI. Conclusions

This paper has analyzed the resource allocation problem with physical layer security in MEC systems. Based on the MEC system model for secure offloading, we established the energy consumption model for local computing, offloading, and MEC. To minimize the system energy consumption, two schemes were proposed in this paper to optimize jointly the allocation of local computing tasks, local CPU frequency, offloading power, and offloading timeslots, while ensuring the computing tasks being completed successfully and the secure offloading constraints being met. The simulation results have verified the effectiveness of the proposed schemes. Specifically, when the leaking distance exceeds a certain distance, the existence of the eavesdropper has almost no impact on task offloading, and the energy consumption of the computing tasks is almost stable. Therefore, a safe area and long leaking distance can reduce the power consumption for secure offloading.

Appendix

A. Difference of Convex Algorithm

The general form of DC programming problems is given by

$$\min_{\mathbf{x}} \ f_0\left(\mathbf{x}\right) - g_0\left(\mathbf{x}\right) \tag{45}$$

$$\text{s.t. } f_i\left(\mathbf{x}\right) - g_i\left(\mathbf{x}\right) \leq 0, i = 1, ..., m, \tag{46}$$

where $f_i\left(\mathbf{x}\right)$ and $g_i\left(\mathbf{x}\right)$ $\left(i = 0, ..., m\right)$ are convex functions. The concave convex procedure (CCCP) from the optimization literature can be employed to convexify the problem (45), which approximates the concave function $\left(-g_i\left(\mathbf{x}\right)\right)$ with its first order Taylor expansion around a feasible point $\mathbf{x}^n$ $\left(n \geq 0\right)$, and the optimization problem (45) can be rewritten as

$$\min_{\mathbf{x}} \ f_0\left(\mathbf{x}\right) - g_0\left(\mathbf{x}^n\right) - \nabla g_0(\mathbf{x}^n)^T\left(\mathbf{x} - \mathbf{x}^n\right) \tag{47}$$

$$\text{s.t. } f_i\left(\mathbf{x}\right) - g_i\left(\mathbf{x}^n\right) - \nabla g_i(\mathbf{x}^n)^T\left(\mathbf{x} - \mathbf{x}^n\right) \leq 0, i = 1, ..., m. \tag{48}$$

Since the problem (45) has been transformed into a convex optimization problem (47), it can now be solved by the standard convex optimization technique. Let $\mathbf{x}^{n+1}$ represent the solution of the optimization problem (47), the solution will be further improved by convexifing (45) on the new point $\mathbf{x}^{n+1}$ similar to the procedure performed on $\mathbf{x}^n$. This sequential programming process continues for several iterations.

Lemma 3: All feasible points of the optimization problem (47) are feasible points of the optimization problem (45).

Proof: Let $\hat{g}_i\left(\mathbf{x}; \mathbf{x}^n\right)$ denote the first-order Taylor expansion of $g_i\left(\mathbf{x}\right)$ around $\mathbf{x}^n$. Since the first-order Taylor expansion is a global overestimator, we can obtain

$$f_i\left(\mathbf{x}\right) - \hat{g}_i\left(\mathbf{x}; \mathbf{x}^n\right) \leq f_i\left(\mathbf{x}\right) - g_i\left(\mathbf{x}\right) \leq 0, \left(i = 1, ..., m\right), \tag{49}$$

which indicates that the feasible points, which satisfy all constraints in (48), also satisfy all constraints in (46). ■

B. Proof of Lemma 2

Let $\mathbf{x}^{n+1}$ denote the optimal solution of (47) at the $(n+1)$-th iteration, $g_i(\mathbf{x})$ is replaced with its first-order Taylor expansion $\hat{g}_i(\mathbf{x};\mathbf{x}^n)$ to form the $(n+1)$-th problem. Three properties of DCA method are provided in the follow combined with Lemma 3.

1) The inequality (49) in Lemma 3 holds for arbitrary $\mathbf{x}$ and $\mathbf{x}^n$;
2) The value of $\hat{g}_i(\mathbf{x};\mathbf{x}^n)$ when $\mathbf{x}$ takes $\mathbf{x}^n$ satisfies the following equation, i.e.,

$$\hat{g}_i(\mathbf{x}^n;\mathbf{x}^n) = g_i(\mathbf{x}^n) + \nabla g_i(\mathbf{x}^n)^T(\mathbf{x}^n - \mathbf{x}^n) = g_i(\mathbf{x}^n); \tag{50}$$

3) The gradient of $\hat{g}_i(\mathbf{x};\mathbf{x}^n)$ is given by

$$\begin{aligned}\nabla\hat{g}_i(\mathbf{x};\mathbf{x}^n) &= \nabla\left(g_i(\mathbf{x}^n) + \nabla g_i(\mathbf{x}^n)^T(\mathbf{x}-\mathbf{x}^n)\right)\\ &= \nabla g_i(\mathbf{x}^n) + \nabla\left(\nabla g_i(\mathbf{x}^n)^T(\mathbf{x}-\mathbf{x}^n)\right)\\ &= \nabla g_i(\mathbf{x}^n) + \nabla(\mathbf{x}-\mathbf{x}^n).\end{aligned} \tag{51}$$

Then, the gradient of $\hat{g}_i(\mathbf{x};\mathbf{x}^n)$ when $\mathbf{x}$ takes $\mathbf{x}^n$ can be given as

$$\nabla\hat{g}_i(\mathbf{x}^n;\mathbf{x}^n) = \nabla g_i(\mathbf{x}^n) + \nabla(\mathbf{x}^n - \mathbf{x}^n) = \nabla g_i(\mathbf{x}^n). \tag{52}$$

Since (49), (50) and (52) are sufficient conditions for the convergence to a KKT point [49], the DCA method will converge to a KKT point.

References

[1] J. B. Wang, J. Wang, Y. Wu, J. Wang, H. Zhu, M. Lin, and J. Wang, "A machine learning framework for resource allocation assisted by cloud computing," *IEEE Networks Mag.*, pp. 144-151, March/April 2018.

[2] H. Zhu and J. Wang, "Chunk-based resource allocation in OFDMA systems - Part I: chunk allocation," *IEEE Trans. Commun.*, vol. 57, no. 9, pp. 2734-2744, Sept. 2009.

[3] H. Zhu and J. Wang, "Chunk-based resource allocation in OFDMA systems - Part II: joint chunk, power and bit allocation," *IEEE Trans. Commun.*, vol. 60, no. 2, pp. 499-509, Feb. 2012.

[4] M. Chiang and T. Zhang, "Fog and IoT: An overview of research opportunities," *IEEE Internet Thing J.*, vol. 3, no. 6, pp. 854-864, Dec. 2016.

[5] K. Kumar and Y.-H. Lu, "Cloud computing for mobile users: Can offloading computation save energy?," *Computer*, vol. 43, no. 4, pp. 51-56, Apr. 2010.

[6] E. Cuervo, et al., "MAUI: Making smartphones last longer with code offload," in *Proc. ACM MobiSys*, San Francisco, CA, USA, Mar. 2010, pp. 49-62.

[7] Z. Xiao, W. Song, and Q. Chen, "Dynamic resource allocation using virtual machines for cloud computing environment," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 6, pp. 1107-1117, Jun. 2013.

[8] H. N. Van, F. D. Tran, and J.-M. Menaud, "Performance and power management for cloud infrastructures," in *Proc. IEEE Cloud Comput.*, Jun. 2010, pp. 329-336.

[9] A. Ahmed and E. Ahmed, "A survey on mobile edge computing," in *Proc. IEEE ISCO*, Jan. 2016, pp. 1-8.

[10] "Mobile-edge computingłIntroductory technical white paper," White Paper, ETSI, Sophia Antipolis, France, Sep. 2014.

[11] 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. Survey Tuts*, vol. 19, no. 3, pp. 1657-1681, 4th Quart. 2017.

[12] P. Porambage, J. Okwuibe, M. Liyanage, M. Ylianttila and T. Taleb, "Survey on Multi-Access Edge Computing for Internet of Things Realization," *IEEE Commun. Survey Tuts*, vol. 20, no. 4, pp. 2961-2991, 4th Quart. 2018.

[13] Q. Pham, F. Fang, V. N. Ha, M. Le, Z. Ding, L. B. Le and W. Hwang, "A Survey of Multi-Access Edge Computing in 5G and Beyond: Fundamentals, Technology Integration, and State-of-the-Art," submitted to *IEEE Commun. Survey Tuts*, https://arxiv.org/abs/1906.08452?context=cs.NI.

[14] T. G. Rodrigues, K. Suto, H. Nishiyama, N. Kato and K Temma, "Cloudlets Activation Scheme for Scalable Mobile Edge Computing with Transmission Power Control and Virtual Machine Migration," *IEEE Trans. Comput.*, vol. 67, no. 9, pp. 1287-1300, Sep. 2018.

[15] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Survey Tuts.*, vol. 19, no. 4, pp. 2322-2358, 4th Quart., 2017.

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

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

[18] C. You, K. Huang, and H. Chae, "Energy efficient mobile cloud computing powered by wireless energy transfer," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1757-1771, May 2016.

[19] Y. Mao, J. Zhang, and K. B. Letaief, "Dynamic computation offloading for mobile-edge computing with energy harvesting devices," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3590-3605, Dec. 2016.

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

[21] S. Guo, J. Liu, Y. Yang, B. Xiao and Z. Li, "Energy-Efficient Dynamic Computation Offloading and Cooperative Task Scheduling in Mobile Cloud Computing," *IEEE Trans. Mobile Comput.*, vol. 18, no. 2, pp. 319-333, Feb. 2019.

[22] J. Ren, G. Yu, Y. Cai and Y. He, "Latency Optimization for Resource Allocation in Mobile-Edge Computation Offloading," *IEEE Trans. Wireless Commun.*, vol. 17, no. 8, pp. 5506-5519, Aug. 2018.

[23] M. Nir, A. Matrawy and M. St-Hilaire, "Economic and Energy Considerations for Resource Augmentation in Mobile Cloud Computing," *IEEE Trans. Cloud Comput.*, vol. 6, no. 1, pp.99-113, Jan. 2018.

[24] 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.

[25] T. X. Tran and D. Pompili, "Joint Task Offloading and Resource Allocation for Multi-Server Mobile-Edge Computing Networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 856-868, Jan. 2019.

[26] L. Zhao, J. Wang, J. Liu and N. Kato, "Optimal Edge Resource Allocation in IoT-Based Smart Cities," *IEEE Network*, vol.33, no.2, pp.30-35, Mar. 2019.

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

[28] T. Zhao, S. Zhou, X. Guo, Y. Zhao, and Z. Niu, "A cooperative scheduling scheme of local cloud and Internet cloud for delayaware mobile cloud computing," in *Proc. IEEE Globecom*, pp. 1-6, Dec. 2015.

[29] Y. Chen, Y. Zhang, S. Maharjan, M. Alam and T. Wu, "Deep learning for secure mobile edge computing in cyber-physical transportation systems," *IEEE Network*, vol. 33, no. 4, pp. 36-41, July 2019.

[30] A. D. Wyner, "The wire-tap channel," *Bell Syst. Tech. J.*, vol. 54, no. 8, pp. 1355-1387, Oct. 1975.

[31] Y. Liang, H. V. Poor, and S. Shamai, "Secure communication over fading channels," *IEEE Trans. Inf. Theory*, vol. 54, no. 6, pp. 2470-2492, Jun. 2008.

[32] X. Wang, M. Tao, J. Mo, and Y. Xu, "Power and subcarrier allocation for physical-layer security in OFDMA-based broadband wireless networks," *IEEE Trans. Inf. Forens. Security*, vol. 6, no. 3, pp. 693-702, Sep. 2011.

[33] J. Xu and J. Yao, "Exploiting Physical-Layer Security for Multiuser Multicarrier Computation Offloading," *IEEE Wireless Commun. Lett.*, vol. 8, no. 1, pp. 9-12, Feb. 2019.

[34] X. Wang and G. B. Giannakis, "Power-efficient resource allocation for time-division multiple access over fading channels," *IEEE Trans. Inf. Theory*, vol. 54, no. 3, pp. 1225-1240, Mar. 2008.

[35] S. Bi and Y. J. Zhang, "Computation Rate Maximization for Wireless Powered Mobile-Edge Computing With Binary Computation Offloading," *IEEE Trans. Wireless Commun.*, vol. 17, no. 6, pp. 4177-4190, Jun. 2018.

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

[37] S. Mao, S. Leng, K. Yang, X. Huang, and Q. Zhao, "Fair energy-efficient scheduling in wireless powered full-duplex mobile-egde computing systems," in *Proc. IEEE Global Commun. Conf.*, Singapore, 2017.

[38] T. D. Burd and R. W. Brodersen, "Processor design for portable systems," *J. VLSI Signal Process. Syst.*, vol. 13, nos. 2-3, pp. 203-221, 1996.

[39] M. R. A. Khandaker, C. Masouros, and K. K. Wong, "Constructive interference based secure precoding: A new dimension in physical layer security," *IEEE Trans. Inf. Forens. Security*, vol. 13, no. 9, pp. 2256-2268, Sep. 2018.

[40] T. D. Quoc and M. Diehl, "Sequential convex programming methods for solving nonlinear optimization problems with DC constraints," 2011. Available: arXiv:1107.5841v1.

[41] H. H. Kha, H. D. Tuan, and H. H. Nguyen, "Fast global optimal power allocation in wireless networks by local D.C. programming," *IEEE Trans. Wireless Commun.*, vol. 11, no. 2, pp. 510-515, Feb. 2012.

[42] A. Khabbazibasmenj and S. A. Vorobyov, "Robust adaptive beamforming for general-rank signal model with positive semi-definite constraint via POTDC," *IEEE Trans. Signal Process.*, vol. 61, no. 23, pp. 6103-6117, Dec. 2013.

[43] Y. Liu and W. Chen, "Limited-feedback-based adaptive power allocation and subcarrier pairing for OFDM DF relay networks with diversity," *IEEE Trans. Veh. Technol.*, vol. 61, no. 6, pp. 2559-2571, Jul. 2012.

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

[45] S. Boyd, "Ellipsoid method," Stanford Univ., Stanford, CA, USA, Tech. Rep., May 2014.

[46] C. Wang, C. Liang, F. R. Yu, Q. Chen, L. Tang, "Computation offloading and resource allocation in wireless cellular networks with moible edge computing," *IEEE Trans. Wireless Commun.*, vol. 16, no. 8, pp. 4924-4938, Aug. 2017.

[47] S. Jeong, O. Simeone, and J. Kang, "Mobile edge computing via a UAVmounted cloudlet: Optimization of bit allocation and path planning," *IEEE Trans. Vehicular Technol.*, vol. 67, no. 3, pp. 2049-2063, Mar. 2018.

[48] Y. Zeng, B. Clerckx, and R. Zhang, "Communications and signals design for wireless power transmission," *IEEE Trans. Commun.*, vol. 65, no. 5, pp. 2264-2290, May 2017.

[49] M. Chiang, C. W. Tan, D. P. Palomar, D. O'Neill, and D. Julian, "Power control by geometric programming," *IEEE Trans. Wireless Commun.*, vol. 6, no. 7, pp. 2640-2651, Jul. 2007.

[50] F. Zhou, Y. Wu, R. Q. Hu, and Y. Qian, "Computation rate maximization in uav-enabled wireless powered mobile-edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 1927-1941, Sep. 2018.