Power-Optimal Scheduling for Delay Constrained Mobile Computation Offloading

Di Han*, Wei Chen*, *Senior Member, IEEE*, and Yuguang Fang†, *Fellow, IEEE*
* Tsinghua National Laboratory for Information Science and Technology (TNList)
Department of Electronic Engineering, Tsinghua University, Beijing, CHINA, 100084
† Department of Electrical and Computer Engineering, University of Florida, Gainesville, USA, FL 32611
Email: hd15@mails.tsinghua.edu.cn, wchen@tsinghua.edu.cn, fang@ece.ufl.edu

***Abstract*—In this paper, we aim to obtain the optimal tradeoff among average delay, and average transmission and computation power consumptions in a mobile computation offloading system. A probabilistic approach is developed to jointly determine the transmission and computation rate in each time-slot. We model the queue lengths in the mobile device and computation resource with a two-dimensional Markov chain. Based on this model, we obtain the average delay and power consumption. Then, we formulate a joint queues aware optimization problem to minimize the average power consumption of the mobile device given constraints on average delay of tasks and average power consumption of the computation resource. By converting the problem into a linear programming, we obtain the optimal power-delay tradeoff and power-optimal Joint Transmission and Computing Scheduling (JTCS) strategy. Finally, the optimization results are validated by extensive simulations.**

I. Introduction

In recent years, the demand for mobile devices to execute high computation tasks is increasing rapidly. However, the resources of mobile devices, e.g., battery life and computation capability, are still poor due to the limited physical form factor. To resolve the design issues between resource-hungry applications and resource-poor mobile devices, the computation offloading is proposed to migrate computation to more powerful computation resources (i.e., servers or more powerful computing devices or facilities)[1][2].

Low latency and high energy efficiency are considered as the critical performance metrics for the support of low latency tolerance applications under the limited energy resource of mobile devices in mobile computation offloading systems [3]. Since both of the accumulated delay and power consumption are composed of transmission and computation partitions, the traditional packet-switched transmission may not provide the timeliness requirements with the best-effort service. In contrast, the joint transmission and computing scheduling holds the promise of meeting the above requirements by using the state information of both transmissions and computations.

The joint optimization of transmission and computing has been studied extensively in the past. Barbarossa et al.[5] proposed a centralized scheduling algorithm to jointly optimize the transmission and computation resource allocation with latency requirements for the single user case. The joint optimization was then extended to the multiuser scenario in [6]; see also [7] for a recent survey on joint optimization for computation offloading in a 5G perspective. Besides, in [8], a joint allocation of transmission and computational resources was proposed for femto-cloud computing systems , where each computation task should be completed within required latency tolerance. The power-delay tradeoff for multi-user mobile-edge computing systems was investigated in [9] via joint management of transmission and computational resources. Furthermore, the probabilistic scheduling approach based on queueing theory could hold the promise of improving the system performance by jointly considering the transmission and computation in the scheduling strategy.

In our previous work [10] and [11], we focused on finding the optimal scheduling strategy and power-delay tradeoff in a single queue system and multiple queues in parrel connection based on a probabilistic scheduling strategy. In this paper, we generalize this method to a computation offloading system with one finite-buffer mobile device and one finite-buffer computation resource. The considered system can be modeled as two queues in series connection and then be formulated into a two-dimensional Markov chain whose state is determined by the state of queues. Our objective is to find the optimal JTCS strategy to minimize the average power consumption of the mobile device, i.e., transmission power consumption under the constraints on average accumulated delay and average power consumption in the computation resource, i.e., computation power consumption, leading to the optimal power-delay tradeoff. This optimization problem can be converted into a linear programming problem so that optimal JTCS and power-delay tradeoff can be obtained efficiently. Then, a JTCS strategy is obtained to jointly determine the probabilities of the transmission and computation rate according to the queue states in the mobile device and computation resource. Moreover, it will be shown that there exists a fundamental tradeoff between average delay and power consumptions. It's also worth mentioning that our probabilistic approach can be used in the scheduling in other queue systems with multiple queues in series connection.

This research was supported in part by the National Science Foundation of China under Grant Nos. 61671269 and 61621091. The work of Y. Fang was partially supported by National Science Foundation under CNS-1717736. The work of W. Chen was partially supported by the National 10000-Talent Program of China.

978-1-5386-3180-5/18/$31.00 ©2018 IEEE

II. System Model

As shown in Fig. 1, we consider a computation offloading system which is composed of one mobile device and one computation resource. The mobile device is connected with the computation resource over one wireless link and the computation resource can run on a virtual machine that can execute computing tasks on behalf of the mobile device through computation offloading. Besides, a scheduler is implemented to control the computation offloading.

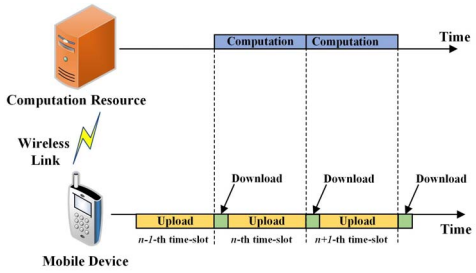

Fig. 1. A scenario of computation offloading system.

Computing tasks are generated in the mobile device randomly and can be offloaded to the computation resource via the wireless link. The total delay of each task incorporates the time to upload the task to the computation resource, the time necessary for the computation resources to execute the tasks, and the time to return the results of computing tasks back to the mobile device. For the sake of simplicity, we assume that the time necessary for return the result of each task is a fixed small value, which can be negligible. Hence, the delay of each task can be considered as the sum of the time necessary for transmission in the upload step and computation in the resource. Then the considered system can be modeled by two queues in series connection, which denote the mobile device and computation resource, as shown in Fig. 2.

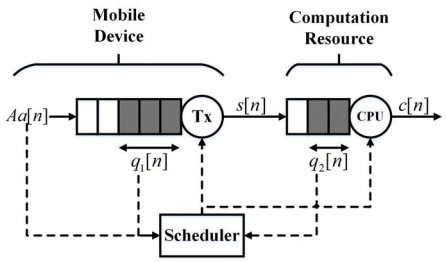

Fig. 2. System model.

The time horizon is divided into time-slots, whose length is T_{s}. Assume that at the beginning of each time-slot, the task packet arrives as Bernoulli Process with an arrival rate α, i.e.,

$$\begin{cases} \Pr\{a[n]=1\}=\alpha, \\ \Pr\{a[n]=0\}=1-\alpha. \end{cases} \tag{1}$$

Each packet arrival contains $A(A \geq 1)$ tasks and each task requires L CPU cycles to execute.

Due to the constraints of the mobile device and computation resource, we assume that at most S and C tasks can be transmitted and computed in each time-slot. In order to guarantee the stability of the system, we set $S \geq A$ and $C \geq A$. The numbers of packets transmitted and computed in the n-th time-slot, i.e., the transmission and computation rate, are denoted by $s[n] \in \{0,1,\ldots,S\}$ and $c[n] \in \{0,1,\ldots,C\}$, whose corresponding power consumption functions are denoted by $p_{\mathrm{s}}[n]=f\left(s[n]\right)$ and $p_{\mathrm{c}}[n]=g\left(c[n]\right)$, respectively.

Assume the length of time-slot T_{s} is long enough, hence the ergodic channel is considered. From perspective of the physical layer, to transmit more bits without increasing the error rate, we should use a larger constellation diagram, therefore more power be consumed for every bit in average. Moreover, the power consumption per CPU cycle in each computation resource can be expressed as [12]

$$P_{\mathrm{c}}(f_{\mathrm{c}})=\kappa {f_{\mathrm{c}}}^2, \tag{2}$$

where f_{c} is the clock frequency and κ is the effective switched capacitance depending on the chip architecture. The total CPU cycles in the n-th time-slot should satisfy

$$f_{\mathrm{c}}[n]T_{\mathrm{s}}=Lc[n]. \tag{3}$$

Then $g\left(c[n]\right)$ can be rewritten as

$$g\left(c[n]\right)=P_{\mathrm{c}}(f_{\mathrm{c}})c[n]L=\frac{\kappa L^3}{{T_{\mathrm{s}}}^2}(c[n])^3. \tag{4}$$

Based on the above analysis, being able to capture the convex relationship between $p_{\mathrm{s}}[n]$ and $s[n]$, and $p_{\mathrm{c}}[n]$ and $c[n]$, i.e., the power consumption functions $f(x)$ and $g(x)$ are strictly increasing and convex.

The arrival tasks can be stored in the finite-buffers of mobile device or computation resource, whose buffer size is M and N tasks, respectively. Let $q_1[n]$ and $q_2[n]$ denote the queue lengths, i.e., queue states, in the mobile device and computation resource at the end of the n-th time-slot, respectively. Consider the typical mobile computation system, it is reasonable to assume that the buffer size of the mobile device should be larger than the number of tasks in one data arrival, i.e., $M > A$. Then we have the basic relationship for $q_1[n]$ and $q_2[n]$ that

$$\begin{cases} q_1[n+1]=\min\left\{M, q_1[n]-s[n]+Aa[n]\right\}, \\ q_2[n+1]=\min\left\{N, q_2[n]-c[n]+s[n]\right\}. \end{cases} \tag{5}$$

The computation offloading system works in the following procedures. The incoming task packet information $Aa[n]$ can be obtained at the beginning of each time-slot, hence it can be taken into consideration along with the states of two buffers, i.e.,$q_1[n]$ and $q_2[n]$ by the scheduler to make a probabilistic strategy for the current time-slot.

III. Delay and Power Analysis based on JTCS

In this section, the JTCS strategy is introduced in a rigorous way firstly. The computation offloading system can be formulated as a two-dimensional Markov chain, based on which, the

average delay and power consumptions of the transmission and the computation can be obtained by the steady-state distribution of the Makov chain.

A. Joint Transmission and Computing Scheduling

In the sense of the average delay, the power-optimal strategy is only aware of how many packets waiting for transmissions, irrespective of when the packet arrives at the queue. Hence, we denote the queue state of the mobile device by

$$\zeta[n] = q_1[n-1] + Aa[n], \tag{6}$$

where $\zeta \in [0, M+1]$. From Eq. (5), we have

$$\zeta[n] = \max\{\zeta[n-1] - s[n-1], 0\} + Aa[n]. \tag{7}$$

On the other hand, the probabilistic scheduling strategy can be determined by the probability $f^{s,c}_{k_1,k_2}$ of $s[n] = s$ and $c[n] = c$ given that $\zeta[n] = k_1$ and $q_2[n] = k_2$, which can be written by

$$f^{s,c}_{k_1,k_2} = \Pr\{s[n] = s, c[n] = c | \zeta[n] = k_1, q_2[n] = k_2\}. \tag{8}$$

Furthermore, the normalization condition always holds for all $k_1 = 1, 2, \ldots, M+1$ and $k_2 = 1, 2, \ldots, N$.

$$\sum_{c=1}^{C}\sum_{s=1}^{S} f^{s,c}_{k_1,k_2} = 1. \tag{9}$$

There are some constraints on the JTCS strategy. To avoid the overflow and underflow, the probabilistic scheduling strategy should satisfy

$$f^{s,c}_{k_1,k_2} = 0 \qquad \text{if } k_1 < s \tag{10.a}$$
$$f^{s,c}_{k_1,k_2} = 0 \qquad \text{if } k_2 < c \tag{10.b}$$
$$f^{s,c}_{k_1,k_2} = 0 \qquad \text{if } k_1 - s + A \geq M+1 \tag{10.c}$$
$$f^{s,c}_{k_1,k_2} = 0 \qquad \text{if } k_2 + c - i \geq N+1, \tag{10.d}$$

where Eqs. (10.a-b) mean that the transmission and computation rate cannot exceed the number of tasks in buffers and Eqs. (10.c-d) are proposed to avoid packet dropping.

The system model with computation offloading procedure have been formulated so far. According to the description of the JTCS strategy, the problem can be formulated as a two-dimensional Markov chain with $\zeta[n]$ and $q_2[n]$ as its states. The state space of the Markov chain is given by

$$\mathrm{M} = \{(\zeta, q_2) | 0 \leq \zeta \leq M+1, 0 \leq q_2 \leq N\}. \tag{11}$$

For ease of understanding, a instance with transition diagram of one state that $\zeta[n] = 3$ and $q_2[n] = 3$ is given in Fig. 3. To keep the figure legible, we denote $(a[n], s[n], c[n]) = (a, s, c)$ as (a, s, c) for each link. Moreover, the state $\zeta[n] = 3$ and $q_2[n] = 3$ cannot transfer to the states that do not have a link with it, e.g., $\zeta[n] = 2$ and $q_2[n] = 1$.

Denote $\lambda^{q_1,q_2}_{k_1,k_2}$ as the transition probability from state (k_1, k_2) to state (q_1, q_2) in the two-dimensional Markov chain, which is summarized in the following theorem.

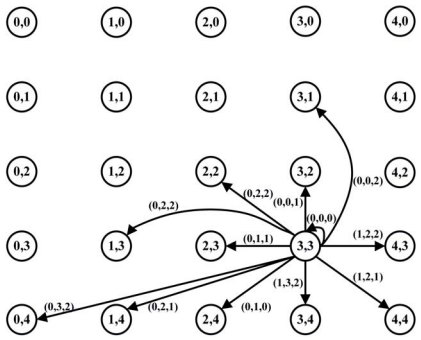

Fig. 3. Transition Diagram of $\zeta[n] = 3$ and $q_2[n] = 3$ ($M = 3$, $N = 4$, $S = 3$, $C = 2$, $A = 3$).

Theorem 1 *The transition probability $\lambda^{q_1,q_2}_{k_1,k_2}$ satisfies*

$$\lambda^{q_1,q_2}_{k_1,k_2} = \alpha f^{s_1,c_1}_{k_1,k_2}{}_{\{s_1\in\mathcal{S},c_1\in\mathcal{C}\}} + (1-\alpha) f^{s_2,c_2}_{k_1,k_2}{}_{\{s_2\in\mathcal{S},c_2\in\mathcal{C}\}}, \tag{12}$$

where

$$\begin{cases} s_1 = A + k_1 - q_1, \\ c_1 = A + k_1 + k_2 - q_2, \\ s_2 = k_1 - 1_1, \\ c_2 = k_1 + k_2 - q_2. \end{cases} \tag{13}$$

Proof: The core idea of this proof is that the transition of the sate only relies on the transmission and computation rate, and the packet arrival in the next time-slot, i.e.,

$$\begin{aligned} \zeta[n+1] - \zeta[n] &= Aa[n] - s[n], \\ q_2[n+1] - q_2[n] &= s[n] - c[n]. \end{aligned} \tag{14}$$

Consider $a[n] = 1$ whose probability is α, we have

$$\begin{aligned} s[n] &= Aa[n] + \zeta[n] - \zeta[n+1] = A + k_1 - q_1, \\ c[n] &= s[n] + q_2[n] - q_2[n+1] = A + k1 + k2 - q_2. \end{aligned} \tag{15}$$

Similarly, we can obtain $s[n]$ and $c[n]$ in the case that $a[n] = 0$. ■

B. Delay and Power Analysis

When the conditions (9) and (10) are satisfied, one can always find a single close positive recurrent aperiodic class in this Markov chain [11]. In this way, the steady-state distribution always exists and is independent of the initial state.

Denote π_{k_1,k_2} as the steady-state distribution probability of state (k_1, k_2) in the Markov chain. The steady-state distribution of this Markov chain is denoted as

$$\boldsymbol{\pi}_{M+2\times N+1} = \begin{bmatrix} \pi_{0,0} & \cdots & \pi_{0,n} & \cdots & \pi_{0,N} \\ \cdots & & \cdots & & \cdots \\ \pi_{1,0} & \cdots & \pi_{m,n} & \cdots & \pi_{m,N} \\ \cdots & & \cdots & & \cdots \\ \pi_{M+1,0} & \cdots & \pi_{M+1,n} & \cdots & \pi_{M+1,N} \end{bmatrix}. \tag{16}$$

For convenience, we convert the matrix $\boldsymbol{\pi}_{M+2\times N+1}$ into a column vector, i.e., the vectorization of a matrix. Then we have the column vector

$$\boldsymbol{\pi}_{(M+2)(N+1)\times 1} = \mathbf{vec}(\boldsymbol{\pi}_{M+2\times N+1}). \tag{17}$$

For simplicity, we refer to $\boldsymbol{\pi}_{(M+2)(N+1)\times 1}$ as $\boldsymbol{\pi}$, where

$$\boldsymbol{\pi} = [\pi_{0,0}, \pi_{0,1}, \ldots, \pi_{0,N}, \ldots, \pi_{M+1,0}, \ldots, \pi_{M+1,N}]^T. \quad (18)$$

Moreover, $\boldsymbol{\pi}$ satisfies

$$\boldsymbol{H}\boldsymbol{\pi} = \boldsymbol{\pi}, \quad (19.a)$$

$$\mathbf{1}^T\boldsymbol{\pi} = 1, \quad (19.b)$$

where $\boldsymbol{H}$ is the transition matrix of the Markov chain and the elements of which are defined by Eq. (12), e.g., the first row of $\boldsymbol{H}$ is given by

$$\boldsymbol{h}_1 = \{\lambda^{0,0}_{0,0}, \lambda^{0,0}_{0,1}, \ldots, \lambda^{0,0}_{0,N}, \ldots, \lambda^{0,0}_{M+1,N}\}. \quad (20)$$

Denote $\mathbf{I}$ as the identity matrix, $\mathbf{1} = [1, \ldots, 1]^T$, and $\mathbf{0} = [0, \ldots, 0]^T$. Moreover, we won't specify their size if there is no ambiguity. Hence,

$$\boldsymbol{\pi} = \mathbf{G}^{-1}\mathbf{c}, \quad (21)$$

where $\mathbf{G} = \begin{bmatrix} \mathbf{1}^T \\ \mathbf{H} - \mathbf{I} \end{bmatrix}$ and $\mathbf{c} = [1, \mathbf{0}]^T$. In other words, $\boldsymbol{H}$ can be used to represent a certain strategy, i.e., $f^{s,c}_{k_1,k_2}$. It will determine the value of $\boldsymbol{\pi}$.

In the n-th time-slot, when $\zeta[n] = k_1$ and $q_2 = k_2$, the power consumption for the mobile device and computation resource is given by $f_{\mathrm{s}}(s)$ and $g_{\mathrm{c}}(c)$ with probability $f^{s,c}_{k_1,k_2}$, $\forall s \in \mathcal{S}$ and $\forall c \in \mathcal{C}$. Then, the average power consumptions and average delay are summarized in the following Proposition and Theorem.

Proposition 1 *The average transmission and computation power consumptions are given by*

$$P^{\mathrm{ava}}_{\mathrm{s}} = \sum_{k_1=0}^{M+1}\sum_{k_2=0}^{N}\sum_{s=0}^{S}\sum_{c=0}^{C} \pi_{k_1,k_2} f^{s,c}_{k_1,k_2} f_{\mathrm{s}}(s), \quad (22.a)$$

$$P^{\mathrm{ava}}_{\mathrm{c}} = \sum_{k_1=0}^{M+1}\sum_{k_2=0}^{N}\sum_{s=0}^{S}\sum_{c=0}^{C} \pi_{k_1,k_2} f^{s,c}_{k_1,k_2} g_{\mathrm{c}}(c). \quad (22.b)$$

Theorem 2 *The average delay is given by*

$$D = \frac{1}{\alpha A}\sum_{k_1=0}^{M+1}\sum_{k_2=0}^{N}(k_1+k_2)\pi_{k_1,k_2}. \quad (23)$$

Proof: The queue length of the computation offloading system is the sum of queue lengths in the mobile device and computation resource. Therefore, the average queue length L is given by

$$\begin{aligned} L &= \lim_{n\to\infty} \mathrm{E}\{q_1[n] + q_2[n]\} \\ &= \lim_{n\to\infty} \mathrm{E}\{\zeta[n+1] - Aa[n+1] + q_2[n]\} \\ &= \sum_{k_1=0}^{M+1}\sum_{k_2=0}^{N}(k_1+k_2)\pi_{k_1,k_2} - A\alpha. \end{aligned} \quad (24)$$

According to Little's Law, the average length L is equal to the arrival rate λA multiplied by the average time T that a task spends in the queue that a task spends in the offloading system. Therefore, the average delay in steady state can be directly expressed as

$$T = \frac{L}{\lambda A} = \frac{1}{\alpha A}\sum_{k_1=0}^{M+1}\sum_{k_2=0}^{N}(k_1+k_2)\pi_{k_1,k_2} - 1. \quad (25)$$

Moreover, the total delay is the sum of waiting time in the queue and serving time of each task, i.e., one time-slot. Finally, we have $D = L + 1$ and then this theorem has been proved. ■

Based on the above analysis of the average delay, and power consumptions in transmission and computation, an optimal JTSC strategy can be obtained in the next section.

IV. Optimal Power-Delay Tradeoff

In typical system, the computing task has the delay tolerance D^{th} before the computation offloading should be completed, which is a fixed design value. In other word, we only need to guarantee that the delay does not exceed the delay tolerance instead of minimizing the delay. Moreover, as we mentioned in the Section II, the battery life of the mobile device is poor, hence it is necessary to conserve the power consumption of the mobile device as soon as possible, i.e., to minimize the average transmission power consumption. Furthermore, there also should be a constraint of average power consumption of the computation resource, i.e., the average computation power consumption constraint $P^{\mathrm{th}}_{\mathrm{c}}$.

Therefore, we aim to the minimize the average transmission power consumption under the constraints on average delay tolerance and the average computation power consumption-Generally speaking, to achieve lower delay, the mobile device and the computation resource should conduct transmission and computation at higher rate, resulting in consuming more energy. Hence there is a fundamental tradeoff between the average delay and average power consumptions. Then, we have the following optimization problem:

$$\min_{\boldsymbol{\pi}, f^{s,c}_{k_1,k_2}} \quad P^{\mathrm{ava}}_{\mathrm{s}} = \sum_{k_1=0}^{M+1}\sum_{k_2=0}^{N}\sum_{s=0}^{S}\sum_{c=0}^{C} \pi_{k_1,k_2} f^{s,c}_{k_1,k_2} f_{\mathrm{s}}(s) \quad (26.a)$$

$$\text{s.t} \quad \frac{1}{\alpha A}\sum_{k_1=0}^{M+1}\sum_{k_2=0}^{N}(k_1+k_2)\pi_{k_1,k_2} \le D^{\mathrm{th}} \quad (26.b)$$

$$\sum_{k_1=0}^{M+1}\sum_{k_2=0}^{N}\sum_{s=0}^{S}\sum_{c=0}^{C} \pi_{k_1,k_2} f^{s,c}_{k_1,k_2} g_{\mathrm{c}}(c) \le P^{\mathrm{th}}_{\mathrm{c}} \quad (26.c)$$

$$\mathbf{1}^T\boldsymbol{\pi} = 1 \quad (26.d)$$

$$\boldsymbol{H}\boldsymbol{\pi} = \pi \quad (26.e)$$

$$\sum_{c=1}^{C}\sum_{s=1}^{S} f^{s,c}_{k_1,k_2} = 1 \quad \forall k_1, k_2 \quad (26.f)$$

$$f^{s,c}_{k_1,k_2} \ge 0 \quad \forall k_1, k_2, s, c \quad (26.g)$$

$$\pi_{k_1,k_2} \ge 0 \quad \forall k_1, k_2, \quad (26.h)$$

where constraints (26.b) and (26.c) denote the constraints

of delay torlerance and computation power consumption, respectively. Clearly, the objective function and constraints in optimization (26) are linear combinations of $\{\pi_{k_1,k_2} f_{k_1,k_2}^{s,c}\}$ $\{f_{k_1,k_2}^{s,c}\}$, or $\{\pi_{k_1,k_2}\}$. By recalling the normalization condition of $\{f_{k_1,k_2}^{s,c}\}$ in Eq. (9), π_{k_1,k_2} can also be expressed as

$$\pi_{k_1,k_2} = \sum_{c=1}^{C}\sum_{s=1}^{S} \pi_{k_1,k_2} f_{k_1,k_2}^{s,c} = \sum_{c=1}^{C}\sum_{s=1}^{S} y_{k_1,k_2}^{s,c}. \tag{27}$$

By substituting Eq. (27) into Eq. (26), the equation constraints (26.d) and (26.e) can be expressed as a matrix equation $\mathbf{Qy} = 0$ with $y_{k_1,k_2}^{s,c} = \{\pi_{k_1,k_2} f_{k_1,k_2}^{s,c}\}$ as variables, where the constant matrix is denoted by $\mathbf{Q}$ and can be derived from $\mathbf{H}$. In this way, the optimization (26) is converted into a linear programming which is summarized in the following theorem.

Theorem 3 The optimization problem (26) is equivalent to the following linear programming problem

$$\min_{y_{k_1,k_2}^{s,c}} P_{\mathrm{s}}^{\mathrm{ava}} = \sum_{k_1=0}^{M+1}\sum_{k_2=0}^{N}\sum_{s=0}^{S}\sum_{c=0}^{C} y_{k_1,k_2}^{s,c} f_{\mathrm{s}}(s) \tag{28.a}$$

$$\text{s.t} \quad \frac{1}{\alpha A}\sum_{k_1=0}^{M+1}\sum_{k_2=0}^{M}\sum_{c=1}^{C}\sum_{s=1}^{S}(k_1+k_2) y_{k_1,k_2}^{s,c} \le D^{\mathrm{th}} \tag{28.b}$$

$$\sum_{k_1=0}^{M+1}\sum_{k_2=0}^{N}\sum_{s=0}^{S}\sum_{c=0}^{C} y_{k_1,k_2}^{s,c} g_{\mathrm{c}}(c) \le P_{\mathrm{c}}^{\mathrm{th}} \tag{28.c}$$

$$\sum_{k_1=0}^{N+1}\sum_{k_2=0}^{M}\sum_{c=1}^{C}\sum_{s=1}^{S} y_{k_1,k_2}^{s,c} = 1 \tag{28.d}$$

$$\boldsymbol{Q}\mathbf{y} = 0 \tag{28.e}$$

$$y_{k_1,k_2}^{s,c} \ge 0 \quad \forall k_1,k_2,s,c, \tag{28.f}$$

where $\mathbf{y}$ is a column vector with $y_{k_1,k_2}^{s,c}$ as components.

Proof: Firstly, we proceed to prove each component in those two problems can be converted equivalently. Define $y_{k_1,k_2}^{s,c} = \{\pi_{k_1,k_2} f_{k_1,k_2}^{s,c}\}$ and π_{k_1,k_2} can be expressed as $\pi_{k_1,k_2} = \sum_{c=1}^{C}\sum_{s=1}^{S} y_{k_1,k_2}^{s,c}$. Therefore, each component in optimization problem (26) can be converted into the corresponding form in the optimization problem (28).

Then we need to prove that all the feasible solution of those two problems are bijective. For each feasible solution π_{k_1,k_2} and $f_{k_1,k_2}^{s,c}$ in problem (26), $y_{k_1,k_2}^{s,c} = \{\pi_{k_1,k_2} f_{k_1,k_2}^{s,c}\}$ is still feasible to problem (28). For each feasible solution $y_{k_1,k_2}^{s,c}$ of problem (28), the corresponding solution of problem (26) can be obtained by $\pi_{k_1,k_2} = \sum_{c=1}^{C}\sum_{s=1}^{S} y_{k_1,k_2}^{s,c}$. Moreover, another part of feasible solution of problem (26), i.e., $f_{k_1,k_2}^{s,c}$, is considered in the following Eqs. (30) and (31).

Therefore, all the feasible solutions of those two problems can be converted equivalently, i.e., bijective. ∎

After the optimal solution ${y_{k_1,k_2}^{s,c}}^*$ of the linear programming (28) is obtained, the corresponding steady-state distribution can be represented as

$$\pi_{k_1,k_2}^* = \sum_{c=1}^{C}\sum_{s=1}^{S} {y_{k_1,k_2}^{s,c}}^*. \tag{29}$$

To obtain the power-optimal strategy, we can derive ${f_{k_1,k_2}^{s,c}}^*$ from ${y_{k_1,k_2}^{s,c}}^*$, which is presented as follows.

Case 1 When $\pi_{k_1,k_2}^* \neq 0$, the optimal strategy is given by

$${f_{k_1,k_2}^{s,c}}^* = \frac{{y_{k_1,k_2}^{s,c}}^*}{\pi_{k_1,k_2}^*}. \tag{30}$$

Case 2 When $\pi_{k_1,k_2}^* = 0$, which means that the state (k_1,k_2) is a transient state. Then, a simple strategy can be used, i.e.,

$${f_{k_1,k_2}^{s,c}}^* = \frac{1}{(1+k_1)(1+k_2)} \quad 0 \le s \le k_1, 0 \le c \le k_2. \tag{31}$$

In conclusion, the optimal JTCS strategy is obtained and the optimal power-delay tradeoff can be achieved by this strategy.

Remark 1 *Our proposed model and approach can be extended and applied to a more generalized scenario whose problem has considered the download step into the analysis. The preocess of downloading results can also be modeled by a new queue whose input is connected with the output of the queue of computation resource in Fig. 2. Then the optimal power-delay tradeoff and JTCS strategy can be obtained using a similar approach in this section.*

V. Numerical Results

In this section, we validate our theoretical results by the simulation studies, and explain the outcomes in a more comprehensive way. Throughout this section, we set $\alpha = 0.5$, $A = 3$, $M = 6$, $N = 5$, $S = 4$, $C = 3$, $f(x) = 2(2^x - 1)$, and $g(x) = x^3$. Based on the optimization problem (28), the optimal JTCS strategy can be obtained.

The optimal tradeoff between average delay and average computation power consumption constraint with different average transmission power consumption $\overline{P}_{\mathrm{s}}^{\mathrm{ava}} = 5$, 6 and 8 are shown in Fig. 4. The optimization results obtained by the optimal JTCS strategy based on linear programming and simulation results are given by Monte-Carlo simulation. The optimization results match perfectly well with the simulation results, which confirms the optimal JTCS strategy. With the increase of the $\overline{P}_{\mathrm{c}}^{\mathrm{th}}$, the average delay decreases and then approach to the different asymptotic lines and the line with a higher $\overline{P}_{\mathrm{s}}^{\mathrm{ava}}$ has the lower value. In particular, the values of the average delays in asymptotic lines for $\overline{P}_{\mathrm{s}}^{\mathrm{ava}} = 5$ and 6 are approximately 125% and 111% of that for $\overline{P}_{\mathrm{s}}^{\mathrm{ava}} = 8$.

The optimal tradeoff among average delay, average transmission and computation power consumptions is shown in Fig. 5. As it is expected, the average transmission power consumption decreases when both average delay constraint and computation power consumption constraint increase, which is also matched with the results in Fig. 4. When the $\overline{P}_{\mathrm{c}}^{\mathrm{th}}$ and D^{th} are small enough, the average transmission power consumption approaches to 7, i.e., $s[n] = 3$ when $a[n] = 3$. Moreover, when $\overline{P}_{\mathrm{s}}^{\mathrm{ava}}$ and $\overline{P}_{\mathrm{c}}^{\mathrm{th}}$ are large enough, the tasks can be transmitted as soon as each packet arrives, i.e., transmission rate is always $\min\{\zeta[n], S\}$, and then be computed at the maximal computation rate $\min\{q_2[n], C\}$ in each time-slot. Hence, the average delay is only two time-slot.

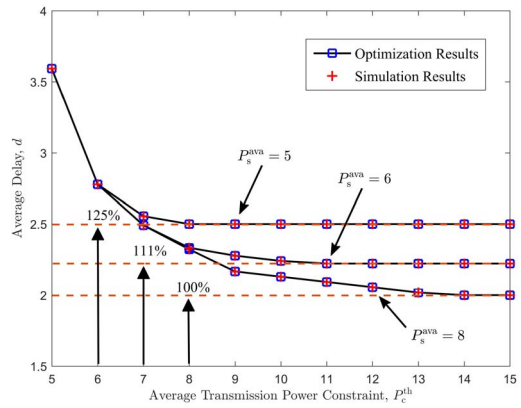

Fig. 4. Comparison of Optimization and Simulation Results.

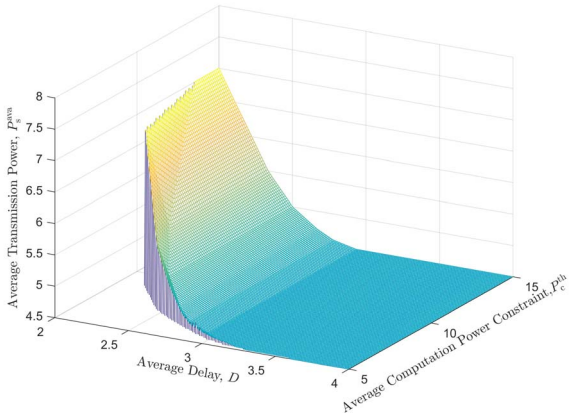

Fig. 5. Optimal Power-Delay Tradeoff.

The relationship between the transmission and computation power consumptions with varying average delay is shown in Fig. 6, which is the cross-sections of Fig. 5. For a given average delay, there are different available requirements on the required average power consumptions. Moreover, for a given average delay constraint, the required average transmission power consumption decreases with the increase of average communication or transmission power consumption and vice versa. Therefore, there is also a tradeoff between the transmission and computation power consumptions with a given average delay. Besides, both minimal available average power consumptions in transmission and computation are decreasing with the increase of the average delay constraint.

VI. Conclusion

In this paper, we have investigated a power-optimal joint transmission and computing scheduling strategy in a mobile computation offloading system. The transmission and computation rates are scheduled according to the buffer states of the mobile device and computation resource. This system can be formulated as a two-dimensional Markov chain. Based on this formulation, the average delay and power consumptions have

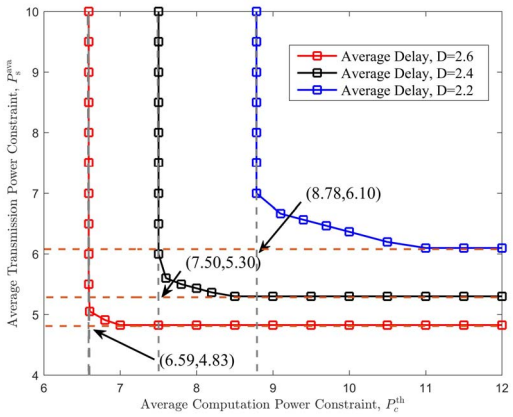

Fig. 6. Optimal Transmission-Computation Power Tradeoff with Different Average Delays.

been analyzed. Then, the optimization problem is solved to minimize the average power consumption of the mobile device under the constraints on the average delay and computation power consumptions of the computation resource. In this way, a JTCS strategy is obtained to obtain the optimal power-delay tradeoff in the computation offloading system, which is validated by simulations. Our future work will focus on a more generalized model combining adaptive transmission and computation rates with fading channels.

References

[1] R. Kemp, N. Palmer, T. Kielmann, and H. Bal, "Cuckoo: a computation offloading framework for smartphones," *Proc. MobiCASE*, Oct. 2010.

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

[3] K. Kumar, J. Liu, Y. Lu, and B. Bhargava, "A survey of computation offloading for mobile systems," *Mobile Netw. Appl.,* vol. 18, no. 1, pp. 129C140, 2013.

[4] Y. Cui, X. Ma, H. Wang, I. Stojmenovic, and J. Liu, "A survey of energy efficient wireless transmission and modeling in mobile cloud computing," *Mobile Netw. Appl.,* vol. 18, no. 1, pp. 148-155, Feb. 2013.

[5] S. Barbarossa, S. Sardellitti, and P. Di Lorenzo, "Computation Offloading for mobile cloud computing based on wide cross-layer optimization," *Proc. Future Netw. Mobile Summit.*, Jul. 2013.

[6] S. Barbarossa, S. Sardellitti, and P. Di Lorenzo, "Joint allocation of computation and communication resources in multiuser mobile cloud computing," *Proc. IEEE SPAWC.*, Jun. 2013.

[7] S. Barbarossa, S. Sardellitti, and P. Di Lorenzo, "Communicating while computing: Distributed mobile cloud computing over 5G heterogeneous networks," *IEEE signal Process. Mag.,* vol. 31, n0. 6, pp. 45-55, 2014.

[8] O. Munoz, A. Iserte, and J. Vidal, "Optimization of radio and computational resources for energy efficiency in latency-constrained application offloading," *IEEE Trans. Veh. Technol.,* vol. 64, no. 10, pp. 4738-4755, Oct. 2015.

[9] Y. Mao, J. Zhang, S. H. Song, and K. B. Letaief, "Power-delay tradeoff in multi-user mobile-edge computing systems," *Proc. IEEE GLOBECOM.,* Dec. 2016,

[10] W. Chen, Z. Cao, and K. B. Letaif, "Optimal delay-power tradeoff in wireless transmission with fixed modulation," *Proc. IEEE IWCLD*, 2007, pp. 60-64.

[11] X. Zhao and W. Chen, "Delay optimal no-orthogonal multuple access with joint scheduling and superposition coding," accepted by *Proc. IEEE GLOBECOM*, 2017.

[12] T. Burd and R. Broderson, "Processor design for portable systems," *J. VLSI Singapore Process*, vol. 13, pp. 203-222, 1996.