Tasks Scheduling and Resource Allocation in Heterogeneous Cloud for Delay-bounded Mobile Edge Computing

Tianchu Zhao, Sheng Zhou, Xueying Guo, Zhisheng Niu
Tsinghua National Laboratory for Information Science and Technology
Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
Email: zhaotc13@mails.tsinghua.edu.cn, sheng.zhou@tsinghua.edu.cn
guo-xy11@mails.tsinghua.edu.cn, niuzhs@tsinghua.edu.cn

***Abstract*—Mobile edge computing is a novel technique in which mobile devices offload computation-intensive tasks with stringent delay requirements to the edge cloud. However, the limited computational resource in the edge cloud may result in the Quality of Service degradation. In this paper, we address this issue by coordinating the heterogeneous cloud which includes the edge cloud and the remote cloud. Considering the offloading of delay-bounded tasks, we study into the scheduling of heterogeneous cloud in order to maximize the probability that tasks can have the delay requirements met. The problem formulation is proved to be concave, and an optimal algorithm is proposed accordingly. The optimal policy with heterogeneous cloud is notably different from the policy merely using the edge cloud. With only the edge cloud, the system serves tasks with loose delay bounds and drops tasks with stringent delay bounds when the traffic load is heavy. However, with the heterogeneous cloud, tasks with stringent delay bounds are offloaded to the edge cloud and tasks with loose delay bounds are offloaded to the remote cloud. In numerical results, the probability that the delay bounds of tasks are satisfied can be improved by about 40% with the assistance of the remote cloud.**

I. Introduction

With the development of smart phones and wearable devices, an increasing number of mobile applications are widely used by mobile users. By February 2017, more than 2.7 million applications are available in the Android market [1]. However, the proliferation of applications poses big challenges to the computational resource and battery life of mobile devices. To address the challenges, Mobile Cloud Computing (MCC) has recently been proposed [2]. In MCC systems, mobile devices can offload computation-intensive tasks to the cloud servers so that their performance is highly improved. However, the conventional centralized cloud is generally deployed remotely from users. For delay-sensitive applications, the data transmission delay over the Internet is a major obstacle to satisfy the bounded delay requirements [3]. To serve the delay-sensitive applications, the Mobile Edge Computing (MEC) architecture has been proposed [4] [5]. In MEC, the cloud servers are deployed locally, which decreases the data transmission delay between mobile users and the cloud, and is consequently more efficient in delay-sensitive applications.

The MEC is a distributed cloud system, and each edge cloud only serves the users in a small area. Here, the edge servers in the network is referred to as the edge cloud. In contrast, the conventional centralized cloud is referred to as the remote cloud. Since the scale of the edge cloud is small, the multiplexing gain of the edge cloud is not as large as the remote cloud. A survey of Microsoft [6] shows that the total cost of a cloud increases with the decrease of the scale. Thus, the computational resource in the edge cloud is generally limited, while the resource in the remote cloud is abundant [7]. In fact, the edge cloud and the remote cloud have different features. To better serve users with different Quality of Service (QoS) requirements, the heterogeneous cloud are suggested to be jointly scheduled [8].

There are some recent papers focusing on the optimization of the QoS in the MEC system. In [9], the authors study ways to minimize the overall energy consumption of mobile users in the MEC system by the joint optimization of radio and computational resources. In [10], the authors propose an algorithm for uplink and downlink beamforming and computational resource allocation so that the total energy consumption of users is minimized. In [11], the authors consider the scenario of TDMA networks and the optimal radio and computational resource allocation policy is derived in close-form. In [12], the authors study the energy-delay tradeoff in MEC systems, and they design algorithms to decide if tasks should be offloaded to the edge cloud. These works consider the usage of the edge cloud, which may result in the degradation of the QoS when the traffic load is heavy. Further more, the mean delay bounds are considered to be the constraints in the models. However, the stochastic wireless channel results in the random data transmission delay, and the time-varying traffic load also leads to the uncertainty of the cloud execution delay. The randomness of data transmission delay and cloud execution delay is a challenge to meet the fixed delay bounds of mobile users. In [13] and [14], the workload sharing between the edge and the remote cloud is studied. The papers optimize the tradeoff between the mean offloading delay and the cost of the system. The optimization of the mean delay can hardly guarantee the QoS of mobile users with different requirements. To offload tasks with different delay bounds, the proposed algorithms are hard to be implemented.

In this paper, we study into how the tasks are scheduled to heterogeneous cloud and how the computational resource in the edge cloud is allocated to users. Considering the offloading of tasks with bounded delay requirements, each task is firstly transmitted in the wireless networks and then executed in the cloud. If the total offloading delay is not larger than the given delay bound, the task is successfully processed; otherwise, the task fails. Taking the stochastic wireless channel and the time-varying traffic load into consideration, the offloading delay is modeled after random variable. The optimization problem is to maximize the probability that the corresponding delay bounds of tasks are satisfied. We find that the optimal offloading policies under heterogeneous-cloud scenario and edge-cloud scenario are notably different when the traffic load is heavy. If the tasks are only offloaded to the edge cloud, the computational resource is allocated to the tasks with loose delay bounds, and the tasks with stringent delay bounds are dropped. However, if both the edge and the remote cloud are available, the edge cloud allocates more computational resource to the tasks with stringent delay bounds, and the tasks with loose delay bounds are scheduled to the remote cloud. As the edge cloud is designed to serve delay-sensitive tasks, the optimal policy under edge-cloud scenario obviously contradicts the original purpose. Thus, the heterogeneous cloud are necessary to work together so that the users with different delay requirements can be simultaneously served.

The rest of the paper is organized as follows. Section II introduces the system model and formulates the problem. In Section III, we study the tasks scheduling policy in the single-user case. In Section IV, we study the tasks scheduling and resource allocation policy in the multi-user case. In Section V, the numerical results are shown to validate our analysis. The paper is concluded in Section VI.

II. System Model and Problem Formulation

We consider a MEC system with wireless network, the edge cloud and the remote cloud, which is shown in Fig. 1. When a task arrives, the user firstly transmits the data to the access point by the wireless network. As long as the transmission of the data is finished, the task goes to either the edge cloud or the remote cloud. Virtualized machines (VM) in the edge cloud are assigned to the users, which are used to execute the tasks. Meanwhile, the VMs in the remote cloud are already prepared for the users.

A. User Model

There are N mobile users in the MEC system, which are denoted by the set $\mathcal{N} = [1, ..., N]$. We assume that each user offloads the same kind of tasks to the cloud, while tasks between users might be different. From the statistical data of Google data centers [15], it is shown that the arrival intervals between tasks are exponentially distributed. Thus, we assume that the arrival process of tasks are Poisson process, and the arrival rate of tasks from the ith user is λ_i. Here, λ_i denotes the number of arrived tasks in a period of time. Each task

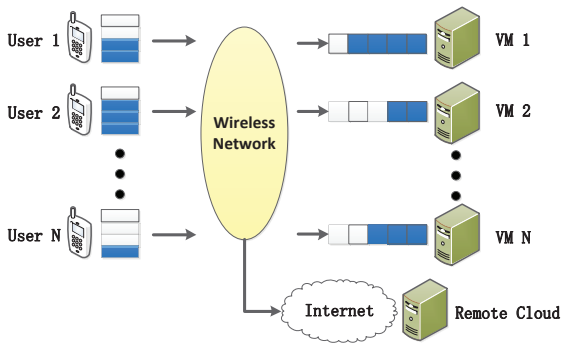


Fig. 1. System model.

should transmit a certain amount of data to the cloud server before it is executed. Due to the relative small size of the downloading data [11], we mainly consider the uploading transmission delay. We assume that the uploading data size of the ith user is L_i bits.

The total offloading delay is composed of two parts: wireless transmission delay and cloud execution delay. For delay-bounded tasks, each task is successfully executed as long as the total offloading delay is smaller than the given delay bound. For the ith user, we assume that the delay bound is denoted by T_i, the wireless transmission delay is $T_{\mathrm{w},i}$, and the cloud execution delay is $T_{\mathrm{c},i}$. We define $p_{\mathrm{succ},i}$ as the probability that the offloading delay of the tasks from the ith user is smaller than the delay bound, which indicates the proportion of tasks satisfying the corresponding delay bound.

$$p_{\mathrm{succ},i} = P(T_{\mathrm{w},i} + T_{\mathrm{c},i} \leq T_i) \tag{1}$$

B. Wireless Transmission Model

Assume that users are allocated with orthogonal resource (e.g., OFDMA systems), and the bandwidth resource allocated to each user is B. For the ith user, the data is transmitted to the access point in packets. Assume that $|h_i|^2$ is the channel gain, γ_i is the SNR and r_i is the transmission rate. At the access point, the packet can be successfully decoded as long as the channel capacity is larger than r_i. Otherwise, the transmission is failed and the packet should be retransmitted. The condition that a packet is successfully decoded by the access point is that:

$$B\log(1 + |h_i|^2\gamma_i) \geq r_i \tag{2}$$

For the ith user, the transmission period of each packet is $T_{\mathrm{p},i} = L_i/r_i$. If a packet is transmitted within the time $T_{\mathrm{w},i}$, at least one packet is successfully decoded by the access point among $\lfloor T_{\mathrm{w},i}/T_{\mathrm{p},i} \rfloor$ packets. Here, $\lfloor x \rfloor$ indicates the largest integer that is not larger than x. Thus, the the cumulative distribution function of the transmission delay $T_{\mathrm{w},i}$ is:

$$P(t \leq T_{\mathrm{w},i}) = 1 - [1 - P(B\log(1 + |h|^2\gamma) \geq r)]^{\lfloor T_{\mathrm{w},i}/T_{\mathrm{p},i} \rfloor} \tag{3}$$

C. Cloud Execution Model

As the arrival of tasks follows Poisson process and the transmission delay is exponentially distributed, the input of

the edge cloud is also Poisson process. According to the data from Google servers [15], the leaving intervals between tasks are exponentially distributed. Thus, the execution delay of the edge cloud is modelled after M/M/1 queue, which is adopted by many related works to evaluate the delay of cloud servers (e.g., [16], [17]). We assume that the total computational resource of the edge cloud is $\mu_{\max}$, which denotes the maximum service rate. There are totally N users, and we assume that the computational resource allocated to the ith user is μ_i. The execution delay of the offloaded tasks from the ith user follows the distribution:

$$P(t \leq T_{\mathrm{c},i}) = 1 - e^{-(\mu_i - \lambda_i)T_{\mathrm{c},i}} \tag{4}$$

If a task of the ith user is offloaded to the remote cloud, the cloud execution delay includes both the Internet transmission delay and the remote cloud execution delay. It is shown that the Internet transmission delay follows an empirical distribution [18], which is assume to be $P(t \leq T_I)$. As the remote cloud has abundant resource, the execution delay distribution $P(t \leq T_{re})$ is weakly influenced by the variation of the workload. Thus, we assume that the remote cloud execution delay of the ith user follows the distribution: $P(t \leq T_I + T_{re})$.

D. Problem Formulation

Considering the stochastic wireless channel and the random cloud execution delay, the delay bound of each task can only be satisfied with a probability. To optimize the QoS of the system, the success probability, which denotes the probability that delay bounds of tasks are satisfied, should be maximized. The success probability of tasks from all users is:

$$\frac{\sum_{i=1}^{N} \lambda_i p_{\mathrm{succ},i}}{\sum_{i=1}^{N} \lambda_i} \tag{5}$$

As $\sum_{i=1}^{N} \lambda_i$ is a fixed value, the optimization problem is formulated as problem P1. In the formulation, $\sum_{i=1}^{N} \lambda_i p_{\mathrm{succ},i}$ denotes the total number of tasks satisfying the corresponding delay bounds in a period of time. As μ_i denotes the computational resource of the edge cloud that is allocated to the ith user, the constraint $\sum_{i=1}^{N} \mu_i \leq \mu_{\max}$ indicates the limitation of the total computational resource.

$$\begin{aligned} \max \quad & \sum_{i=1}^{N} \lambda_i p_{\mathrm{succ},i} \\ \text{s.t.} \quad & \sum_{i=1}^{N} \mu_i \leq \mu_{\max} \end{aligned} \tag{P1}$$

III. Tasks Scheduling in the Single-User Case

For the ith user, the optimization objective is $\lambda_i p_{\mathrm{succ},i}$. In this part, we study into how the tasks of a single user should be scheduled to both the edge cloud and the remote cloud so that the success probability is maximized.

A. Wireless Transmission Delay

In our scenario, we consider Reyleigh fading channel in which $|h_i|^2$ follows exponential distribution with mean value 1. The success decoding probability of each packet is:

$$P(B\log(1+|h_i|^2\gamma_i) \geq r_i) = \mathrm{e}^{-\frac{1}{\gamma_i}(2^{\frac{r_i}{B}}-1)} \tag{6}$$

In Lemma 1, the transmission delay is approximated to be exponential distribution. In fact, exponentially distributed transmission delay is also adopted by many papers to capture the retransmission phenomenon [19] [20]. The optimal transmission rate and packet transmission delay are derived accordingly.

Lemma 1 *For the ith user whose SNR is γ_i, the optimal packet transmission delay distribution is approximated to be:*

$$P(t \leq T_{w,i}) \approx 1 - e^{-\omega_i T_{w,i}} \tag{7}$$

where

$$\begin{aligned} \omega_i &= \frac{B}{L_i}\left[\frac{r_{opt,i}}{B} e^{-\frac{1}{\gamma_i}(2^{\frac{r_{opt,i}}{B}}-1)}\right] \\ r_{opt,i} &= \frac{BW(\gamma_i)}{\ln(2)} \end{aligned} \tag{8}$$

and W indicates the Lambert W function.

Proof: See Appendix A. □

In Lemma 1, ω_i denotes the service rate of the wireless channel. In other words, ω_i indicates the number of successfully transmitted packets in one period of time, which is proportional to B and $1/L_i$. In the mobile device of each user, the packets are queued to be transmitted. Thus, the wireless transmission delay is:

$$P(t \leq T_{\mathrm{w},i}) = 1 - e^{-(\omega_i - \lambda_i)T_{\mathrm{w},i}} \tag{9}$$

B. Offloading Delay

Each task is firstly transmitted in the wireless network, and then goes to either the edge or the remote cloud. We assume that the arrival rate of tasks which are scheduled to the edge cloud is $\lambda_{\mathrm{e},i}$, and the arrival rate of remotely-offloaded tasks is $\lambda_i - \lambda_{\mathrm{e},i}$ accordingly.

For the tasks which are scheduled to the edge cloud, the offloading delay distribution is:

$$\begin{aligned} P_{\mathrm{e},i}(t \leq T_i) &= P(T_{\mathrm{w},i} + T_{\mathrm{c},i} \leq T_i) \\ &= \int_0^{T_i} (1 - \mathrm{e}^{-(\omega_i - \lambda_i)t})(\mu_i - \lambda_{\mathrm{e},i})\mathrm{e}^{-(\mu_i - \lambda_{\mathrm{e},i})(T_i - t)}\, dt \\ &= 1 - \frac{(\omega_i - \lambda_i)\mathrm{e}^{-(\mu_i - \lambda_{\mathrm{e},i})T_i} - (\mu_i - \lambda_{\mathrm{e},i})\mathrm{e}^{-(\omega_i - \lambda_i)T_i}}{(\omega_i - \lambda_i) - (\mu_i - \lambda_{\mathrm{e},i})} \end{aligned}$$

For the tasks which are scheduled to the remote cloud, the offloading delay distribution is $P_{\mathrm{r},i}(t \leq T_i) = P(T_{\mathrm{w},i} + T_{\mathrm{I}} + T_{\mathrm{re}} \leq T_i)$. Let $P_{\mathrm{r},i}(t \leq T_i) = C_i$.

C. Optimal Offloading Policy of a Single Users

The success probability of one single user is maximized by jointly scheduling tasks to the edge and the remote cloud. The optimization problem is formulated as problem P2. $\lambda_{\mathrm{e},i}P_{\mathrm{e},i}(t \leq T_i)$ denotes the number of tasks satisfying the delay bound which are scheduled to the edge cloud, and $(\lambda_i - \lambda_{\mathrm{e},i})P_{\mathrm{r},i}(t \leq T_i)$ indicates the number of tasks satisfying the delay bound which are scheduled to the remote cloud.

$$\max_{\lambda_{\mathrm{e},i}} \ \lambda_{\mathrm{e},i}P_{\mathrm{e},i}(t \leq T_i) + (\lambda_i - \lambda_{\mathrm{e},i})P_{\mathrm{r},i}(t \leq T_i) \tag{P2}$$

Lemma 2 *Problem P2 is a concave optimization problem.*

Proof: See Appendix B. □

In Lemma 2, the problem P2 is proved to be concave. As the only variable is $\lambda_{\mathrm{e},i}$, the problem can be solved by gradient descent algorithm.

D. Optimal Offloading Policy with Sufficient Bandwidth Resource

When the users are allocated with sufficient bandwidth resource, $\omega_i \gg 1$ holds. Thus, $\mathrm{e}^{-(\omega_i - \lambda_i)T_i} \approx 0$, and the previous problem P2 is turned into the following problem P3, which could be solved in close-form.

$$\max_{\lambda_{\mathrm{e},i}} \ \lambda_{\mathrm{e},i}(1 - \mathrm{e}^{-(\mu_i - \lambda_{\mathrm{e},i})T_i}) + C_i(\lambda_i - \lambda_{\mathrm{e},i}) \tag{P3}$$

Lemma 3 *The solution of P3 is*

$$\lambda_{e,i}^{opt} = \min\{\lambda_i, \max[0, \frac{W((1-C_i)e^{\mu_i T_i + 1}) - 1}{T_i}]\} \tag{10}$$

where W indicates the Lambert W function.

Proof: See Appendix C. □

In Lemma 3, the number of tasks that should be offloaded to the edge cloud is shown. In fact, it is determined by the parameters $\mu_i T_i$ and C_i. When $\mu_i T_i$ becomes larger or C_i is smaller, more tasks should be scheduled to the edge cloud.

IV. Tasks Scheduling and Resource Allocation in the Multi-User Case

In the multi-user case, the system, which maximizes the success probability of tasks from all users, should decide how the computational resource in the edge cloud is allocated to the users and how the tasks are scheduled to heterogeneous cloud. In this part, we will jointly study the resource allocation problem and tasks scheduling problem.

A. Optimal Offloading Policy of Multiple Users

Taking the offloading delay distribution of a single-user (as shown in P2) into P1, the optimization problem in multi-user case is formulated in P4. In the formulation, $\lambda_{\mathrm{e},i}P_{\mathrm{e},i}(t \leq T_i)$ denotes the number of successfully executed tasks that are offloaded to the edge cloud, which is related to the allocated computational resource μ_i. $(\lambda_i - \lambda_{\mathrm{e},i})P_{\mathrm{r},i}(t \leq T_i)$ denotes the number of successfully executed tasks that are offloaded to the remote cloud. As the total computational resource in the edge cloud is limited, $\sum_{i=1}^{N} \mu_i \leq \mu_{\max}$ holds. Furthermore, the arrival rate of tasks scheduled to the edge cloud is constrainted by $0 \leq \lambda_{\mathrm{e},i} \leq \lambda_i$.

$$\begin{aligned} \max_{\mu_i, \lambda_{\mathrm{e},i}} \quad & \sum_{i=1}^{N} \lambda_{\mathrm{e},i}P_{\mathrm{e},i}(t \leq T_i) + (\lambda_i - \lambda_{\mathrm{e},i})P_{\mathrm{r},i}(t \leq T_i) \\ \text{s.t.} \quad & \sum_{i=1}^{N} \mu_i \leq \mu_{\max} \\ & 0 \leq \lambda_{\mathrm{e},i} \leq \lambda_i \quad \forall i \end{aligned} \tag{P4}$$

Lemma 4 *Problem P4 is a concave optimization problem.*

Proof: See Appendix D. □

In Lemma 4, the problem P4 is proved to be concave. As the success probability of each user increases with the allocated resource μ_i, the constraint $\sum_{i=1}^{N} \mu_i = \mu_{\max}$ holds. Meanwhile, the problem has constraints of $\lambda_{\mathrm{e},i}$, which are inequalities. To solve the concave problem with inequality constraints, we rely on the Interior Point Method to solve the problem. Let $\Lambda = [\lambda_{\mathrm{e},1}, \lambda_{\mathrm{e},2}, ..., \lambda_{\mathrm{e},N}]$ and $\Pi = [\mu_1, \mu_2, ..., \mu_{N-1}, \mu_{\max} - \sum_{i=1}^{N-1} \mu_i]$. We define the barrier function as:

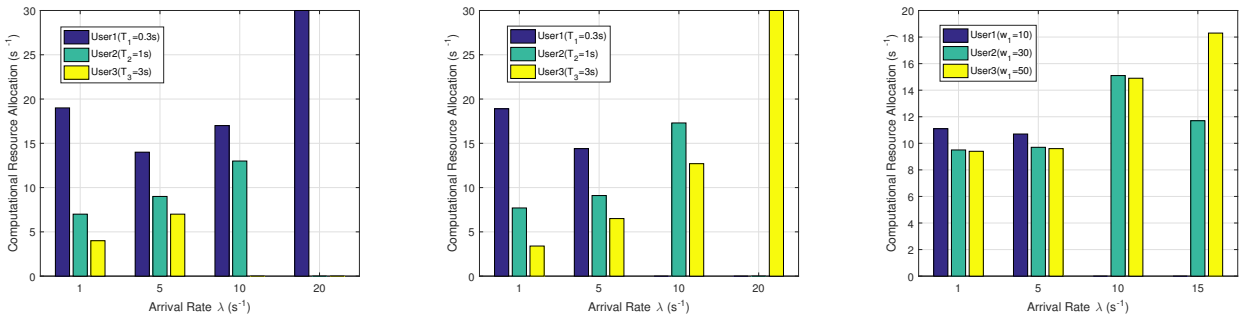
$$\begin{aligned} \phi_k(\Lambda, \Pi) = & \sum_{i=1}^{N} \lambda_{\mathrm{e},i}P_{\mathrm{e},i}(t \leq T_i) + (\lambda_i - \lambda_{\mathrm{e},i})P_{\mathrm{r},i}(t \leq T_i) \\ & + \epsilon_k(\ln(\lambda_{\mathrm{e},i}) + \ln(\lambda_i - \lambda_{\mathrm{e},i})) \end{aligned} \tag{11}$$

Based on the barrier function $\phi(\Lambda, \Pi)$, the problem P4 is turned into a optimization problem which can be solve by the gradient descent algorithm. We define that $GD(\phi_k(\Lambda, \Pi))$ is the gradient descent algorithm which gets the solution of $\max \phi(\Lambda, \Pi)$. The reformulated problem is solved by the following algorithm.

Algorithm 1 Find the optimal tasks scheduling and resource allocation policy

Input: $[\lambda_1, ..., \lambda_N], [T_1, ..., T_N], [\omega_1, ..., \omega_N], [C_1, ..., C_N], \mu_{\max}$
Output: Λ, Π

$\epsilon_k \leftarrow 10^{-k}, \forall k$
$\Delta \leftarrow 10^{-7}$
$k \leftarrow 1$
$\Lambda_1^{\mathrm{opt}}, \Pi_1^{\mathrm{opt}} \leftarrow GD(\phi_1(\Lambda, \Pi))$
$k \leftarrow 2$
$\Lambda_2^{\mathrm{opt}}, \Pi_2^{\mathrm{opt}} \leftarrow GD(\phi_2(\Lambda, \Pi))$
$\xi \leftarrow \|\Lambda_2^{\mathrm{opt}} - \Lambda_1^{\mathrm{opt}}\| + \|\Pi_2^{\mathrm{opt}} - \Pi_1^{\mathrm{opt}}\|$
while $\xi > \Delta$ **do**
    $k \leftarrow k + 1$
    $\Lambda_k^{\mathrm{opt}}, \Pi_k^{\mathrm{opt}} \leftarrow GD(\phi_1(\Lambda, \Pi))$
    $\xi \leftarrow \|\Lambda_k^{\mathrm{opt}} - \Lambda_{k-1}^{\mathrm{opt}}\| + \|\Pi_k^{\mathrm{opt}} - \Pi_{k-1}^{\mathrm{opt}}\|$
end while
$\Lambda \leftarrow \Lambda_k^{\mathrm{opt}}$
$\Pi \leftarrow \Pi_k^{\mathrm{opt}}$



(a) Optimal Policy with Heterogeneous Cloud (b) Optimal Policy with the Edge Cloud (c) Optimal Policy with different wireless transmission delay

Fig. 2. Optimal computational resource allocation policies of the edge cloud: (a) the optimal policy when tasks are scheduled to both the edge and the remote cloud. (b) the optimal policy when tasks are scheduled to the edge cloud. (c) the optimal policy when the different wireless transmission delay of users are considered.

B. Optimal Offloading Policy with Sufficient Bandwidth Resource

With sufficient bandwidth resource, $\mathrm{e}^{-(\omega_i-\lambda_i)T_i} \approx 0$ holds, and the problem is turned into the following optimization problem P5. If the tasks are only offloaded to the edge cloud, the problem is formulated as P6, which can be solved in close-form. By comparing the optimal policy of problem P5 and problem P6, we emphasize the influence of the remote cloud on the system.

1) Heterogeneous Cloud Scenario: The optimization problem in the heterogeneous cloud scenario is:

$$\begin{aligned} \max_{\mu_i,\lambda_{\mathrm{e},i}} \quad & \sum_{i=1}^{N} \lambda_{\mathrm{e},i}(1-\mathrm{e}^{-(\mu_i-\lambda_{\mathrm{e},i})T_i}) + C_i(\lambda_i-\lambda_{\mathrm{e},i}) \\ \text{s.t.} \quad & \sum_{i=1}^{N} \mu_i \le \mu_{\max} \end{aligned} \tag{P5}$$

Taking $\lambda_{\mathrm{e},i}^{\mathrm{opt}}$ into P5, the optimization problem is simplified. As $\lambda_{\mathrm{e},i}^{\mathrm{opt}}$ is the function of the allocated computational resource μ_i, the variables are actually μ_i. With the Lagrange multipliers, the simplified problem can be solved by the gradient descent algorithm directly.

2) Edge Cloud Scenario: If the users can only offload the tasks to the edge cloud, the optimization problem is:

$$\begin{aligned} \max_{\mu_i} \quad & \sum_{i=1}^{N} \lambda_i(1-\mathrm{e}^{-(\mu_i-\lambda_i)T_i}) \\ \text{s.t.} \quad & \sum_{i=1}^{N} \mu_i \le \mu_{\max} \end{aligned} \tag{P6}$$

To solve the problem P6, the Lagrange function is:

$$L = \sum_{i=1}^{N} \lambda_i(1-\mathrm{e}^{-(\mu_i-\lambda_i)T_i}) + \eta(\mu_{\max} - \sum_{i=1}^{N} \mu_i) \tag{12}$$

Let $\frac{\partial L}{\partial \mu_i} = 0$, the computational resource allocated to the ith user is:

$$\mu_i^* = \lambda_i + \frac{\ln(\lambda_i T_i) - \ln(\eta^*)}{T_i} \tag{13}$$

where η^* is the optimal value of the Lagrange multiplier.

From the solution, it is shown that the edge cloud will not serve users that $\lambda_i T_i < \eta^*$ holds. Thus, when the total arrival rate of tasks is large, the edge cloud will drop the users whose delay bounds are stringent. The edge cloud is deployed to serve the users with stringent delay bounds, but it refuses to serve them when the traffic load is heavy. In fact, the optimal policy of the edge cloud and its purpose contradict with each other.

V. Numerical Results

In this part, the numerical results are simulated to show the optimal policies and evaluate their performances. We consider tasks whose delay bounds vary from 0.3 second to 3 seconds. The range of arrival rates satisfy $\lambda_i \in [1, 30]$. For the parameters of wireless transmission delay, the size of transmitted data ranges from 10^3 bits to 10^6 bits, the distribution of SNR is $\gamma_i \in [1, 30]$, and the bandwidth allocated to each user is 1MHz.

In Fig. 2, the optimal policies in different scenarios are compared. In the figure, x-axis denotes the arrival rates of tasks, and y-axis denotes the allocation of computational resource in the edge cloud. To better find out the features of the policies, we only consider 3 users with the same arrival rate and different other parameters. In Fig. 2(a), the optimal policy in the system with both the edge and the remote cloud is shown. When the tasks are scheduled to heterogeneous cloud, more computational resource is allocated to the users with stringent delay bounds. Especially when the arrival rate is large, the edge cloud serves users whose delay bounds are stringent, and the tasks with loose delay bounds are scheduled to the remote cloud. In Fig. 2(b), the optimal policy in the system with only the edge is shown. When the arrival rate of tasks is small, the policy is similar to the one which schedule tasks to heterogeneous cloud. However, when the arrival rate is large, the edge cloud allocates more computational resource to the users whose delay bounds are loose. Meanwhile, the edge cloud drops the tasks with stringent delay bounds so that the total success probability is maximized. In figure. 2(c), the optimal policy is shown when considering different ω_i of

users. Here, the ω_i indicates the service rate of the wireless channel, which is shown in Lemma 1. And $\frac{1}{\omega_i}$ denotes the mean transmission delay of each packet. The result indicates that when the arrival rate is small, more computational resource is allocated to the users with longer data transmission delay. But when the arrival rate is large, the policy is changed and more computational resource is allocated to the users with shorter data transmission delay.

In Fig. 3, the relationship between the probability that a task is successfully processed within the delay bound and the computational resource in the VM is revealed. With the increase of the allocated computational resource in the VM, more tasks can be offloaded to the edge cloud, and the success probability almost increases linearly until it reaches 1. Meanwhile, with larger arrival rate and smaller delay bound, a user needs more computational resource for the same success probability. When the computational resource satisfy $\mu = 0$, all tasks are scheduled to the remote cloud and the success probability is C_i.

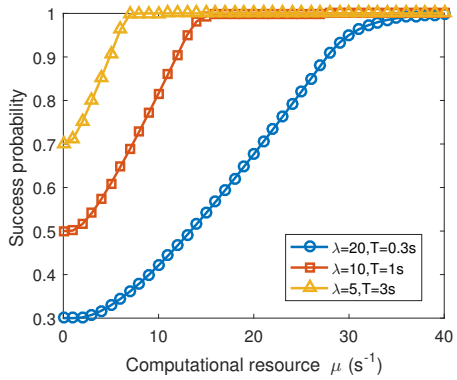


Fig. 3. Success probability vs. computational resource.

In Fig. 4, the performance of three policies are compared. In the figure, x-axis denotes the computational resource in the edge cloud, and y-axis denotes the total success probability of tasks from all users. The first policy is the optimal one which schedules tasks to both the edge and the remote cloud. In the second policy, tasks are scheduled to both the edge and the remote cloud, but the computational resource in the edge cloud is equally allocated to each user. The third policy is the optimal policy which schedules tasks to only the edge cloud. The numerical result shows that the optimal heterogeneous-cloud offloading policy outperforms the other two, especially when the computational resource is limited. By offloading tasks to heterogeneous cloud, the total success probability can be improved by about 40% compared with the edge-cloud offloading policy. The policy that equally allocates the computational resource performs bad, especially when the arrival rates of tasks from the users are different.

VI. Conclusions

In this work, we consider the mobile cloud computing scenario in which both the edge and the remote cloud serve

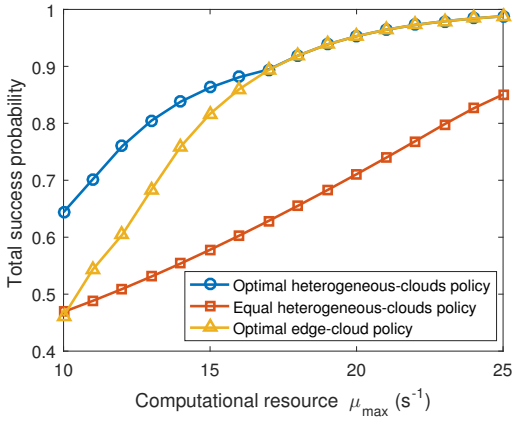


Fig. 4. Total success probability under different policies

mobile users. By jointly scheduling tasks to heterogeneous cloud and allocating computational resource in the edge cloud, we optimize the system so that the probability of tasks satisfying the corresponding delay bounds is maximized. By solving the optimization problem, we find that: 1) In the scenario with heterogeneous cloud, the edge cloud should allocate more computational resource to users with stringent delay bounds. However, in the scenario with only the edge cloud, more computational resource is allocated to users with loose delay bounds when traffic load is heavy. By using heterogeneous cloud, the edge cloud can better serve delay-sensitive tasks. 2) When the traffic load is light, more computational resource is allocated to the users with longer data transmission delay. However, when the traffic load is heavy, more computational resource is allocated to the users with shorter data transmission delay. The heterogeneous-cloud based offloading policy outperforms the edge-cloud based offloading policy, which can improve the probability that the delay bounds of tasks are satisfied by about 40%.

Appendix

A. Proof of Lemma 1

The success decoding probability of a packet is $P(B\log(1+|h_i|^2\gamma_i) \geq r_i)$. The number of transmitted packets in one period of time is denoted by $\omega_i = \frac{P(B\log(1+|h_i|^2\gamma_i)\geq r_i)}{T_{\mathrm{p},i}}$. Considering that the

$$\begin{aligned} P(t \leq T_{\mathrm{w},i}) &= 1-(1-\omega_i T_{\mathrm{p},i})^{\lfloor T_{\mathrm{w},i}/T_{\mathrm{p},i}\rfloor} \\ &\approx \lim_{T_{\mathrm{p},i}\to 0} 1-(1-\omega_i T_{\mathrm{p},i})^{\frac{1}{\omega_i T_{\mathrm{p},i}}\omega_i T_i} \\ &\approx 1-\mathrm{e}^{\omega_i T_i} \end{aligned} \tag{14}$$

In rayleigh fading channel,

$$\omega_i = \frac{r_i}{L_i}\mathrm{e}^{-\frac{1}{\gamma_i}(2^{\frac{r_i}{B}}-1)} \tag{15}$$

The maximum ω_i results in the optimal packet transmission delay. Let $\frac{\partial \omega_i}{\partial r_i} = 0$, and the optimal transmission rate is derived:

$$r_{\text{opt},i} = \frac{B\text{W}(\gamma_i)}{\ln(2)} \tag{16}$$

B. Proof of Lemma 2

Let $a_i = (\omega_i - \lambda_i)T_i$, and define $g_i(x)$ as

$$\begin{aligned} g_i(x) &= 1 - \frac{a_i \text{e}^{-x} - x\text{e}^{-a_i}}{a_i - x} \\ &= 1 - \text{e}^{-a_i} - a_i \text{e}^{-a_i} \frac{\text{e}^{a_i - x} - 1}{a_i - x} \end{aligned} \tag{17}$$

where

$$\begin{aligned} &\frac{\partial g_i(x)}{\partial x} = -a_i \text{e}^{-a_i} \frac{\text{e}^{a_i - x}(1 - (a_i - x)) - 1}{(a_i - x)^2} \geq 0 \\ &\frac{\partial^2 g_i(x)}{\partial x^2} = -a_i \text{e}^{-a_i} (-\frac{\text{e}^{a_i - x}}{a_i - x} + \frac{2\text{e}^{a_i - x}}{(a_i - x^2)} + \frac{2(1 - \text{e}^{a_i - x})}{(a_i - x)^3}) \leq 0 \end{aligned} \tag{18}$$

The problem P2 equals to:

$$h_i(\lambda_{\text{e},i}) = \lambda_{\text{e},i} g_i((u_i - \lambda_{\text{e},i})T_i) - C_i(\lambda_i - \lambda_{\text{e},i}) \tag{19}$$

Because the following inequation holds, problem P2 is concave.

$$\begin{aligned} &\frac{\partial^2 h_i(\lambda_{\text{e},i})}{\partial \lambda_{\text{e},i}^2} \\ &= -2T_i g_i'((u_i - \lambda_{\text{e},i})T_i) + \lambda_{\text{e},i} T_i^2 g_i''((u_i - \lambda_{\text{e},i})T_i) \leq 0 \end{aligned} \tag{20}$$

C. Proof of Lemma 3

Calculate the derivative of problem P3, and let it equals to 0.

$$\lambda_{\text{e},i} - \text{e}^{-(\mu_i - \lambda_{\text{e},i})T_i}(1 + \lambda_{\text{e},i} T_i) - C_i = 0 \tag{21}$$

By solving this problem, the answer to the equation is:

$$\lambda_{\text{e},i}^{\text{opt}} = \min\{\lambda_i, \max[0, \frac{\text{W}((1 - C_i)\text{e}^{\mu_i T_i + 1}) - 1}{T_i}]\} \tag{22}$$

D. Proof of Lemma 4

The same definitions of $h_i(\mu_i, \lambda_{\text{e},i})$ and $g_i(x)$ as in the Proof of Lemma 2 are adopted, but the variables of $h_i(\mu_i, \lambda_{\text{e},i})$ are both μ_i and $\lambda_{\text{e},i}$.

$$\frac{\partial^2 h_i}{\partial \mu_i^2} = \lambda_{\text{e},i} T_i^2 g_i''((u_i - \lambda_{\text{e},i})T_i) \leq 0 \tag{23}$$

$$\frac{\partial}{\partial \lambda_{\text{e},i}}(\frac{\partial h_i}{\partial \mu_i}) = T_i g_i'((u_i - \lambda_{\text{e},i})T_i) - \lambda_{\text{e},i} T_i^2 g_i''((u_i - \lambda_{\text{e},i})T_i) \tag{24}$$

$$\frac{\partial}{\partial \mu_i}(\frac{\partial h_i}{\partial \lambda_{\text{e},i}}) = T_i g_i'((u_i - \lambda_{\text{e},i})T_i) - \lambda_{\text{e},i} T_i^2 g_i''((u_i - \lambda_{\text{e},i})T_i) \tag{25}$$

Calculate the Hessian and the following inequation holds. Thus, problem P4 is concave.

$$\begin{aligned} &\frac{\partial^2 h_i}{\partial \lambda_{\text{e},i}^2} \frac{\partial^2 h_i}{\partial \mu_i^2} - \frac{\partial^2 h_i}{\partial \lambda_{\text{e},i} \partial \mu_i} \frac{\partial^2 h_i}{\partial \mu_i \partial \lambda_{\text{e},i}} \\ &= -(T_i g_i'((u_i - \lambda_{\text{e},i})T_i))^2 \leq 0 \end{aligned} \tag{26}$$

ACKNOWLEDGMENT

This work is sponsored in part by the Nature Science Foundation of China (No. 61461136004, No. 91638204, No. 61571265), and Intel Collaborative Research Institute for Mobile Networking and Computing.

REFERENCES

[1] [Online]. Available: https://www.appbrain.com/stats/number-of-android-apps

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

[3] F. Liu, P. Shu, H. Jin, L. Ding, J. Yu, D. Niu, and B. Li, "Gearing resource-poor mobile devices with powerful clouds: architectures, challenges, and applications," *IEEE Wireless Communications*, vol. 20, no. 3, pp. 14–22, June 2013.

[4] S. Ibrahim, H. Jin, B. Cheng, H. Cao, S. Wu, and L. Qi, "Cloudlet: towards mapreduce implementation on virtual machines," in *Proceedings of the 18th ACM international symposium on High performance distributed computing*. ACM, 2009, pp. 65–66.

[5] J. Liu, T. Zhao, S. Zhou, Y. Cheng, and Z. Niu, "CONCERT: a cloud-based architecture for next-generation cellular systems," *IEEE Wireless Communications*, vol. 21, no. 6, pp. 14–22, 2014.

[6] R. Harms and M. Yamartino, "The economics of the cloud," *Microsoft whitepaper, Microsoft Corporation*, 2010.

[7] T. Zhao, S. Zhou, X. Guo, Y. Zhao, and Z. Niu, "A cooperative scheduling scheme of local cloud and internet cloud for delay-aware mobile cloud computing," in *2015 IEEE Globecom Workshops (GC Wkshps)*, Dec 2015, pp. 1–6.

[8] Z. Sanaei, S. Abolfazli, A. Gani, and R. Buyya, "Heterogeneity in mobile cloud computing: taxonomy and open challenges," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 369–392, 2014.

[9] S. Sardellitti, G. Scutari, and S. Barbarossa, "Joint optimization of radio and computational resources for multicell mobile-edge computing," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 1, no. 2, pp. 89–103, June 2015.

[10] J. Cheng, Y. Shi, B. Bai, and W. Chen, "Computation offloading in cloud-ran based mobile cloud computing system," in *2016 IEEE International Conference on Communications (ICC)*, May 2016, pp. 1–6.

[11] C. You, K. Huang, H. Chae, and B. H. Kim, "Energy-efficient resource allocation for mobile-edge computation offloading," *IEEE Transactions on Wireless Communications*, vol. PP, no. 99, pp. 1–1, 2016.

[12] Y. Mao, J. Zhang, S. H. Song, and K. B. Letaief, "Power-delay tradeoff in multi-user mobile-edge computing systems," in *2016 IEEE Global Communications Conference (GLOBECOM)*, Dec 2016, pp. 1–6.

[13] R. Deng, R. Lu, C. Lai, and T. H. Luan, "Towards power consumption-delay tradeoff by workload allocation in cloud-fog computing," in *2015 IEEE International Conference on Communications (ICC)*, June 2015, pp. 3909–3914.

[14] E. Gelenbe, R. Lent, and M. Douratsos, "Choosing a local or remote cloud," in *Network Cloud Computing and Applications (NCCA), 2012 Second Symposium on*. IEEE, 2012, pp. 25–30.

[15] C. Jiang, Y. Chen, Q. Wang, and K. J. R. Liu, "Data-driven stochastic scheduling and dynamic auction in iaas," in *2015 IEEE Global Communications Conference (GLOBECOM)*, Dec 2015, pp. 1–6.

[16] Y. Feng, B. Li, and B. Li, "Price competition in an oligopoly market with multiple iaas cloud providers," *IEEE Transactions on Computers*, vol. 63, no. 1, pp. 59–73, Jan 2014.

[17] M. Guevara, B. Lubin, and B. C. Lee, "Navigating heterogeneous processors with market mechanisms," in *High Performance Computer Architecture (HPCA2013), 2013 IEEE 19th International Symposium on*, Feb 2013, pp. 95–106.

[18] G. Hooghiemstra and P. Van Mieghem, "Delay distributions on fixed internet paths," Delft University of Technology, Tech. Rep., 2001.

[19] F. P. Kelly, *Reversibility and Stochastic Networks*. Cambridge University Press, 2011.

[20] X. Guo, R. Singh, T. Zhao, and Z. Niu, "An index based task assignment policy for achieving optimal power-delay tradeoff in edge cloud systems," in *2016 IEEE International Conference on Communications (ICC)*, May 2016, pp. 1–7.