Performance Analysis and Optimization for Multimedia Cloud Based on Queuing Model

Xiaoming Nan, Yifeng He, Member, IEEE, and Ling Guan, Fellow, IEEE

Department of Electrical and Computer Engineering
Ryerson University, Toronto, Ontario, Canada
{xnan, yhe, lguan}@ee.ryerson.ca

Abstract

Multimedia cloud, as a specific cloud paradigm, addresses how cloud can effectively process multimedia services and provide Quality of Service (QoS) provisioning for multimedia applications. There are two major challenges in multimedia cloud. The first challenge is the service response time in multimedia cloud, and the second challenge is the cost of the allocated cloud resources. In this paper, we optimize the resource allocation for multimedia cloud based on queuing models. Specifically, we examine the resource allocation problems in single-class service case, multiple-class service case and priority service case, respectively. In each case, we formulate and solve the service response time minimization problem and resource cost minimization problem, respectively. Simulation results demonstrate that the proposed optimal allocation scheme can optimally utilize the cloud resources to achieve a minimum mean service response time or a minimum resource cost.

I. INTRODUCTION

Cloud computing is an emerging computing paradigm that can provide computation, communications, and storage resources as services in a scalable and virtualized manner. According to the provisions at different levels, the cloud computing services can be generally divided into three major categories: Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS) [1], [2]. In the SaaS category, cloud customers send application requests to the data center, which processes the requests using a multi-tenant architecture and then delivers the processing results back to customers.

Among various cloud-based SaaS applications, multimedia applications strongly need assistance from cloud computing. Many multimedia applications, such as image/video retrieval, free viewpoint video and 3D visualization, typically require intensive computation and/or intensive storage, which may become burdens to client devices, especially to the resource-constrained mobile devices. In Multimedia Cloud (MC) [3], cloud service providers deploy cloud resources as utilities to process multimedia requests and then deliver computing results to users. By using multimedia cloud service, users do not need to own costly computing devices. Instead, they can request the powerful cloud servers to process multimedia applications by paying for the occupied resources. However, it's
challenging to satisfy the Quality of Service (QoS) requirements of various multimedia applications in multimedia cloud computing due to the special characteristics of multimedia, such as delay-sensitive requirement. Currently, a lot of research efforts on MC have been focused on QoS provisions for multimedia applications.

This paper focuses on performance analysis and optimization for MC service providers. There are two major concerns for MC service providers: the service response time and the cost. The service response time in the data center is defined as the duration from the time when the service request arrives at the data center to the time when the service result completely departs from the data center. The service response time is a significant Quality of Experience (QoE) factor to measure the performance of a multimedia cloud application. A lower service response time will lead to a higher QoE. Thus, it is important for cloud service providers to meet customers’ requirements on service response time. The second concern is the cost of the allocated cloud resources. The cloud service can generally be divided into three consecutive phases: schedule, computation and transmission. Inappropriate resource allocation among the three phases will result in resource waste and QoE degradation. For example, with much resource allocated on computation phase while little resource on transmission phase, customer’s requests will be processed fast, but the service results cannot be transmitted in time due to the limited transmission resource capacity. Therefore, it is challenging for cloud service providers to optimally allocate resources among the three phases to minimize the cost under the service response time requirement or minimize the mean service response time under the resource cost constraint.

In this paper, we employ the queuing model to investigate the MC resource allocation problems in single-class service case, multiple-class service case and priority service case, respectively. Specifically, we optimize the resource allocation to minimize the mean service response time or minimize the cost in each case. Our contributions in this paper are listed as follows.

1) We model the service process at multimedia cloud data center as three concatenated queuing systems, which are schedule queue, computation queue and transmission queue. We theoretically analyze the relationship between the service response time and the allocated resources in each queuing system.

2) Based on the queuing model, we study cloud resource optimization problems in single-class service case, multiple-class service case and priority service case, respectively. In each case, we formulate and solve the resource optimization problems to minimize the mean service response time and minimize the resource cost, respectively.

The remainder of this paper is organized as follows. Section II discusses the related work. Section III presents the system models. In Section IV, we optimize the cloud resources in single-class service case, multiple-class service case and priority service case, respectively. The simulation results are provided in Section V and the conclusions are drawn in Section VI.

II. RELATED WORK

Cloud computing, as a promising computing paradigm, has broadly attracted significant attentions of researchers from both industry [4], [5], [6], [7] and academia [8]-[19]. Survey work on the general definition, advantages and
challenges of cloud computing can be found from [8], [9], [10].

Multimedia cloud majorly addresses how cloud can process multimedia applications and provide QoS provisioning for multimedia services. Zhu et al. [3] elaborate the multimedia cloud from the perspectives of multimedia-aware cloud and cloud-aware multimedia, and propose the media edge cloud computing architecture aiming to satisfy QoS requirements of multimedia processing. Several cloud-based multimedia applications have also been proposed in the recent years [11]-[14]. Ferretti et al. [11] develop a cross layer architecture to support wireless devices executing multimedia applications through cloud. Wen et al. [12] present a cloud-based video sensing system to perform scalable and adaptive online monitoring. Ye et al. [13] provide a video-based location search service in cloud environment by allowing users upload a short video clip as search sample. Lau et al. [14] propose an architectural framework to employ the on-demand cloud service for IPTV, which enables subscribers to receive television programs and video streams from anywhere. Among these cloud-based multimedia applications, the service response time in data center is taken as an important QoS requirement due to the delay sensitive characteristic of multimedia processing.

The resource allocation strategies in cloud computing are studied in [15]-[19]. Lin et al. [15] develop a self-organizing model to manage cloud resources in the absence of centralized management control. Shi et al. [16] focus on the maximization of the steady-state throughput by deploying resources for the independent equal sized tasks in the cloud. Teng et al. present a resource pricing and equilibrium allocation policy based on the consideration of cloud users’ competition for limited resources [17]. However, the cloud customers’ requirements for service response time are not considered in [15], [16], [17]. Another direction of resource management is to model the relationship between service performance and allocated resources. In [18], a video streaming workload analysis and prediction technique is proposed. With this method, multimedia workload can be inspected from the user access perspective. In [19], the authors classify the application requests into disk- and CPU-bound ones and then construct scaling functions of system capacity accordingly to predict the required resources. Compared to the work in [15]-[19], our work demonstrates the following novelties: 1) we study the relationship between QoS and cloud resource allocation in different service phase based on the proposed queuing model; 2) we analyze the cloud resource allocation in single-class service case, multiple-class service case and priority service case, and provide optimal resource allocation respectively to meet the service response time requirements or to meet the resource cost constraints.

III. SYSTEM MODELS

In this section, we present our system models, including data center architecture model, queuing model and cost model. The data center architecture model is constructed to abstract the practical cloud data center service; the queuing model is built to analyze the relationship between the service performance and the allocated resources; and the cost model is used to evaluate the resource cost.
A. Data Center Architecture

Most of clouds are built in the form of data centers [3], [10]. The data center architecture in this paper consists of a master server, a number of computing servers and a transmission server. All these servers are virtual machine (VM) instances generated in the virtualized way from physical cloud resources. The master server receives all coming requests, and then schedules the requests to the computing servers. The number of the computing servers is denoted by \( N \). The computing servers act as the real processors, which receive tasks from the master server and then process customers’ requests using their own resources and associated media data. The master server and all computing servers are connected with high-speed communication links. We assume that the latency of internal communications between the master server and the computing servers is negligible. Moreover, we assume that each user task is indecomposable and independent with each other, and thus one task can only be processed by one computing server. After processing from the computing servers, all the service results will be transmitted back to customers by the transmission server.

The allocated cloud resources in our system include the resources at the master server, the computing servers and the transmission server. The allocated resource at the master server is represented by the scheduling rate \( S \) in terms of the number of requests scheduled per second; the allocated resource at the computing server \( i \) is represented by the computing rate \( C_i \) in terms of the number of instructions executed per second; and the allocated resource at the transmission server is represented by the transmission rate \( B \) in terms of the number of bits transmitted per second. The resource allocation at the cloud is to determine the scheduling rate \( S \) at the master server, the computing rate \( C_i \) at the computing server \( i (\forall i = 1, 2, \ldots, N) \), and the transmission rate \( B \) at the transmission server, respectively.

Suppose that the cloud data center provides \( M \) classes of services. The class-\( i \) (\( \forall i = 1, 2, \ldots, M \)) service is characterized by four parameters: the average arrival rate \( \lambda_i \) (requests/s) of the requests, the average task size \( F_i \) which is specified by the number of instructions, the average result size \( D_i \) in bits, and the upper bound of service response time \( \tau_i \) in seconds. In the priority service case, each class is assigned a priority parameter.

B. Queuing Model

The queuing model of the data center is shown in Fig. 1. The model consists of three concatenated queuing systems, which are schedule queue, computation queue and transmission queue. The master server maintains the schedule queue to receive all requests from cloud customers. Since the two consecutive arriving requests may be sent from two different customers, the inter-arrival time is a random variable, which can be modeled as an exponential random variable in cloud computing [20]. Therefore, the arrivals of the requests follow a Poisson Process with average arrival rate \( \lambda \). The requests in the schedule queue are sent to the computing servers at the scheduling rate \( S \) by the master server. Each computing server has a corresponding computation queue to store task requests waiting for processing. The results output from the computing servers are first stored in the transmission queue, and then transmitted back to the customers at the transmission rate \( B \) by the transmission server. For each request, a service result will be generated by the computing server. Therefore, the number of the results is equal to the number of the
requests. We also assume that the arrivals of the results at the transmission queue follow a Poisson Process with average arrival rate \( \lambda \).

**C. Cost Model**

The resource cost in the data center is determined by both the utilized resources and the occupied time. The more utilized resources and the longer occupied time will lead to a higher cost. The allocated cloud resources include the resources at the master server, the computing servers, and the transmission server. In this paper, we employ a linear function to model the relationship of the cost and the allocated resources. The total cost is formulated as

\[
C^{tot} = \left( \alpha S + \beta \sum_{i=1}^{N} C_i + \gamma B \right) T, \tag{1}
\]

where \( T \) is the occupied time during which the cloud-based services are provided, \( S \) is the scheduling rate at the master server, \( C_i \) is the computing rate at the computing server \( i \), \( B \) is the transmission rate at the transmission server, \( \alpha \) is the scheduling cost rate at the master server in terms of dollars per request, \( \beta \) is the computing cost rate at the computing servers in terms of dollars per instruction, and \( \gamma \) is the transmission cost rate at the transmission server in terms of dollars per bit. The linear cost model in Equation (1) has been justified by numerical analysis in [4], [5].

**IV. Optimal Cloud Resources Allocation**

In this section, we will use the proposed models described in Section III to study the resource allocation problems in single-class service case, multiple-class service case and priority service case, respectively. In each case, we optimize the allocation of cloud resources to minimize the service response time or minimize the resource cost, respectively.

**A. Single-Class Service Case**

In this subsection, we study the problem of cloud resource allocation in single-class service case, in which there is only one kind of application service provided in the data center.
As described in Section III, the arrivals of requests at the data center follow a Poisson Process with average arrival rate $\lambda$. All requests enter into the schedule queue first. The schedule queue is modeled as an $M/M/1$ queuing system with mean service rate equal to the scheduling rate $S$ of the master server. In order to maintain a stable queue, $\lambda < S$ is required. The mean service response time of the schedule queue is given by $T_{sche}^{sing} = \frac{1}{1-\lambda/S}$.

Since every computing server has the same service procedure in the single-class service case, we employ a weighted scheduling scheme, in which a task request is assigned to the computing server $i$ with probability $p_i (\forall i = 1, 2, \ldots, N)$, and $\sum_{i=1}^{N} p_i = 1$. According to the decomposition property of Poisson Process, the arrivals of the requests at computing server $i$ follow a Poisson Process with average arrival rate $p_i \lambda$. Suppose that the task size of the single-class service is denoted by $F$ in terms of number of instructions. The service at the computing server $i$ is modeled as an $M/M/1$ queuing system with the mean service rate equal to $C_i/F$. To maintain a stable queue, the constraint $p_i \lambda < C_i/F$ should be satisfied. The mean service response time in computation phase is formulated by $T_{comp}^{sing} = \sum_{i=1}^{N} p_i T_{comp}^{(i)} = \sum_{i=1}^{N} \frac{p_i F/C_i}{1-p_i \lambda F/C_i}$.

All service results output from the computing servers are sent to the transmission queue. Since a service result is generated for each request, the arrivals of the results at the transmission queue follow the same Poisson Process with mean arrival rate $\lambda$. The service at the transmission server is also modeled as an $M/M/1$ queuing system with the mean service rate equal to $B/D$ where $D$ is the result size in bits and $B$ is the transmission rate of the transmission server in bits/s. To maintain a stable queue, $\lambda < B/D$ is required. The mean service response time at the transmission server is given by $T_{tran}^{sing} = \frac{D/B}{1-\lambda D/B}$.

The total service response time in the single-class service case is the summation of service response time for the three queues, which can be formulated as

$$T_{tot}^{sing} = T_{sche}^{sing} + T_{comp}^{sing} + T_{tran}^{sing} = \frac{1}{1-\lambda/S} + \sum_{i=1}^{N} \frac{p_i F/C_i}{1-p_i \lambda F/C_i} + \frac{D/B}{1-\lambda D/B}. \tag{2}$$

The total resource cost in the single-class service case can be formulated as

$$C_{tot}^{sing} = \left( \alpha S + \beta \sum_{i=1}^{N} C_i + \gamma B \right) T. \tag{3}$$

1) Service Response Time Minimization Problem: In cloud-based multimedia applications, the end-to-end delay for a request is defined as the sum of the forward transmission delay from the client to the cloud, the backward transmission delay from the cloud to the client, and the service response time at the cloud. For computationally intensive multimedia applications, the service response time at the cloud is a significant component of the end-to-end delay. Thus, reducing the service response time can greatly help reduce the end-to-end delay. However, there is a trade-off between the service response time and the utilized resources at the cloud. Therefore, we formulate a service response time minimization problem, which is stated as: to minimize the total service response time in single-class service case by jointly optimizing the scheduling rate at the master server, the computing rate at each
computing server, and the transmission rate at the transmission server, subject to the queuing stability constraint in each queuing system and the resource cost constraint. Mathematically, the problem can be formulated as follows.

Minimize $T_{\text{sing}}^{\text{tot}} = \frac{1}{1-\lambda/S} + \sum_{i=1}^{N} \frac{p_i F/C_i}{1-p_i F/C_i} + \frac{D/B}{1-\lambda D/B}$

subject to

\begin{align*}
\lambda &< S, \\
p_i \lambda &< C_i / F, \quad \forall i = 1, \ldots, N, \\
\lambda &< B / D, \\
\left( \alpha S + \beta \sum_{i=1}^{N} C_i + \gamma B \right) T &\leq C_{\text{max}},
\end{align*}

(4)

where $C_{\text{max}}$ is the upper bound of the resource cost.

In the optimization problem (4), the objective function is the total service response time. The constraint, $\lambda < S$, represents the queuing stability constraint at the master server. The constraints, $p_i \lambda < C_i / F$, represent the queuing stability constraints at the computing servers. The constraint, $\lambda < B / D$, represents the queuing stability constraint at the transmission server. The constraint, $(\alpha S + \beta \sum_{i=1}^{N} C_i + \gamma B)T \leq C_{\text{max}}$, represents the resource cost constraint.

The Lagrange multiplier method [21] is applied to solve the optimization problem (4). The optimal analytical solution to the service response time minimization problem (4) is given as follows.

\begin{align*}
S^* &= \frac{C_{\text{max}} - (\alpha + \beta F + \gamma D) \lambda T}{\sqrt{\alpha + \beta F \sum_{i=1}^{N} \sqrt{F_i} + \gamma D}} + \lambda, \\
C_i^* &= \frac{(C_{\text{max}} - (\alpha + \beta F + \gamma D) \lambda T) \sqrt{p_i F}}{\sqrt{\alpha + \beta F \sum_{i=1}^{N} \sqrt{F_i} + \gamma D}} + p_i \lambda F, \\
B^* &= \frac{(C_{\text{max}} - (\alpha + \beta F + \gamma D) \lambda T) \sqrt{D}}{\sqrt{\alpha + \beta F \sum_{i=1}^{N} \sqrt{F_i} + \gamma D}} + \lambda D.
\end{align*}

(5)

It should be pointed out that the optimal analytical solution (5) is obtained under the scheduling probability settings $p_i$, $(\forall i = 1, \ldots, N)$. Based on the service response time minimization problem (4), we have the following theorem on the relationship between the service response time and the scheduling probability settings.

**Theorem 1**: Given the service response time minimization problem (4) in single-class service case based on the scheduling probability settings $p_i$, $(\forall i = 1, \ldots, N)$, we have the following two statements: 1) when scheduling probability $p_i$, $(\forall i = 1, \ldots, N)$, is 1 and all other probabilities $p_j$, $(j \neq i, j = 1, \ldots, N)$ are 0, the minimum total service response time is achieved; 2) when scheduling probabilities are all equal, i.e. $p_1 = p_2 = \cdots = p_N = \frac{1}{N}$, the maximum total service response time is achieved.

**Proof**:

Substituting the optimal analytical solution of $S^*$, $C_i^*$, $(\forall i = 1, \ldots, N)$, and $B^*$ in (5) to the objective function
of the service response time minimization problem (4), we can get the total service response time as

\[ T_{\text{tot sing}} = \left( \sqrt{\alpha + \beta F} \sum_{i=1}^{N} \sqrt{p_i} + \sqrt{\gamma D} \right)^2 \]

\[ = \frac{C_{\text{max}}}{T} - (\alpha \lambda + \beta F \lambda + \gamma D \lambda) \]

\[ = (\xi + \sqrt{\beta F} y)^2 \]

(6)

where \( \xi = \sqrt{\alpha + \gamma D} > 0 \), \( y = \sum_{i=1}^{N} \sqrt{p_i} > 0 \), and \( \delta = \frac{C_{\text{max}}}{T} - (\alpha \lambda + \beta F \lambda + \gamma D \lambda) \).

We first prove that the service response time \( T_{\text{tot sing}} \) in Equation (6) is a monotonically increasing function of \( y \). Given \( y_1 \), \( y_2 \), and assuming \( 0 < y_1 < y_2 \), we can get that \((\xi + \sqrt{\beta F} y_1)^2 < (\xi + \sqrt{\beta F} y_2)^2\), since \( \xi > 0 \) and \( \sqrt{\beta F} > 0 \). Moreover, according to the constraints in (4) that \( \lambda < S \), \( p_i \lambda F < C_i \) (\( \forall i = 1, \ldots, N \)), \( \lambda D < B \), and \((\alpha S + \beta \sum_{i=1}^{N} C_i + \gamma B)T \leq C_{\text{max}} \), we can get the following relationship,

\[ \alpha \lambda + \beta F \lambda + \gamma D \lambda \]

\[ = \alpha \lambda + \beta \sum_{i=1}^{N} p_i \lambda F + \gamma \lambda D \]

\[ < \alpha S + \beta \sum_{i=1}^{N} C_i + \gamma B \]

\[ \leq \frac{C_{\text{max}}}{T} \cdot \delta \]

Thus, \( \delta = \frac{C_{\text{max}}}{T} - (\alpha \lambda + \beta F \lambda + \gamma D \lambda) > 0 \). We therefore derive that \((\xi + \sqrt{\beta F} y_1)^2 < (\xi + \sqrt{\beta F} y_2)^2\) for any given \( 0 < y_1 < y_2 \). So, the service response time \( T_{\text{tot sing}} \) in Equation (6) is a monotonically increasing function of \( y \). Thus, the minimum service response time will be achieved at the minimum \( y \), while the maximum service response time achieved at the maximum \( y \).

To achieve the minimal service response time in Equation (6), we need to find the minimum \( y \). We have the following inequation

\[ y^2 = \left( \sum_{i=1}^{N} \sqrt{p_i} \right)^2 \]

\[ = \sum_{i=1}^{N} p_i + 2 \sum_{i=1}^{N} \sum_{j=i+1}^{N} \sqrt{p_i} \sqrt{p_j} \]

\[ \geq \sum_{i=1}^{N} p_i = 1, \]

and the equality holds if and only if \( p_i p_j = 0 \) \( (i \neq j) \). Therefore, solving the following equations

\[
\begin{cases}
\sum_{i=1}^{N} p_i = 1, \\
p_i p_j = 0, \quad i \neq j, \forall i, j = 1, \ldots, N,
\end{cases}
\]

we can get that \( y = \sum_{i=1}^{N} \sqrt{p_i} \) reaches the minimum value 1 when probability \( p_i \) \( (\forall i = 1, \ldots, N) \) is 1 and all other probabilities \( p_j \) \( (j \neq i, j = 1, \ldots, N) \) are 0. Given the service response time minimization problem (4)
based on the scheduling probability settings, when scheduling probability \( p_i \) (\( \forall i = 1, \ldots, N \)), is 1 and all other probabilities \( p_j \) (\( j \neq i, j = 1, \ldots, N \)) are 0, we can achieve the minimum total service response time, which is given by

\[
T_{\text{tot(min)}} = \frac{\left(\sqrt{\alpha} + \sqrt{\beta F} + \sqrt{\gamma D}\right)^2}{C_{\text{max}} - (\alpha \lambda + \beta F \lambda + \gamma D \lambda)}.
\]  

To find the scheduling probability settings which leads to the maximum service response time, we need to get the maximum \( y \). From inequation \((\sqrt{p_i} - \sqrt{p_j})^2 \geq 0\), we can derive that \( p_i + p_j \geq 2\sqrt{p_i} \sqrt{p_j} \), (\( \forall i, j = 1, \ldots, N \)).

Thus, we can derive the following relationship

\[
y^2 = \left(\sum_{i=1}^{N} \sqrt{p_i}\right)^2
= \sum_{i=1}^{N} p_i + \sum_{i=1}^{N} \sum_{j=i+1}^{N} (2\sqrt{p_i} \sqrt{p_j})
\leq \sum_{i=1}^{N} p_i + \sum_{i=1}^{N} \sum_{j=i+1}^{N} (p_i + p_j)
= p_1 + p_2 + p_3 + \cdots + p_{N-1} + p_N
+ (N-1) \quad p_1 + p_2 + p_3 + \cdots + p_{N-1} + p_N
+ \cdots
+ p_N + p_N
= N \sum_{i=1}^{N} p_i
= N,
\]

and the equality holds if and only if \( p_i = p_j \), (\( i \neq j, \forall i, j = 1, \ldots, N \)). Thus, solving the following equations

\[
\begin{align*}
\sum_{i=1}^{N} p_i &= 1, \\
p_i &= p_j, \quad i \neq j, \forall i, j = 1, \ldots, N,
\end{align*}
\]

we can get that when scheduling probability \( p_1 = p_2 = \cdots = p_N = \frac{1}{N} \), the maximum value \( y = \sqrt{N} \) is achieved.

Given the service response time minimization problem (4) based on the scheduling probability settings, when scheduling probabilities are all equal, i.e. \( p_1 = p_2 = \cdots = p_N = \frac{1}{N} \), the maximum total service response time is achieved. The maximum total service response time is given by

\[
T_{\text{tot(max)}} = \frac{\left(\sqrt{\alpha} + \sqrt{N \beta F} + \sqrt{\gamma D}\right)^2}{C_{\text{max}} - (\alpha \lambda + \beta F \lambda + \gamma D \lambda)}.
\]

\( \diamond \)

Theorem 1 is verified in the following example. The number of computing servers is 3 and the scheduling probabilities are denoted as \( p_1, p_2, p_3 \), respectively. The cost rate for scheduling, computing and transmission are set as \( \alpha = 5 \times 10^{-4} \) dollars/request, \( \beta = 6 \times 10^{-6} \) dollars/million instructions (MIs), \( \gamma = 0.08 \) dollars/gigabit, respectively. The upper bound of resource cost \( C_{\text{max}} \) is 5000 dollars, and the service occupation time \( T \) is 1 hour. For the service, the task size is set as \( F = 500 \) MIs, and the result size is set as \( D = 15 \) megabit. The request arrival rate is 150 requests/s. Fig. 2 shows the service response time with different scheduling probability settings. In Fig. 2...
the scheduling probabilities $p_1$, $p_2$, and $p_3$ are all in the range of $[0, 1]$. Given probabilities $p_1$ and $p_2$, $p_3$ can be determined by $1 - p_1 - p_2$. From Fig. 2, we can see that the service response time achieves the minimum value 0.0183 seconds when the scheduling probability settings $(p_1, p_2, p_3)$ are at $(1, 0, 0)$, $(0, 1, 0)$ and $(0, 0, 1)$, while it reaches the maximum value 0.0337 seconds at $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$, which justifies the statements in Theorem 1. Moreover, when $p_2$ is fixed, the maximum service response time is acquired when $p_1 = p_3 = \frac{1}{2} - p_2$. Similar conclusion can be acquired when $p_1$ or $p_3$ is fixed.

2) Resource Cost Minimization Problem: In cloud computing environment, the service providers would like to provide satisfactory services at the minimal resource cost. The resource cost minimization problem can be stated as: to minimize the total resource cost in single-class service case by jointly optimizing the scheduling rate at the master server, the computing rate at each computing server, and the transmission rate at the transmission server, subject to the queuing stability constraint in each queuing system and the constraint on the service response time.

Mathematically, the problem can be formulated as follows.

$$\text{Minimize } C_{\text{sing}}^{\text{tot}} = \left( \alpha S + \beta \sum_{i=1}^{N} C_i + \gamma B \right) T$$

subject to

$$\lambda < S,$$

$$p_i \lambda < C_i / F, \quad \forall i = 1, \ldots, N,$$

$$\lambda < B / D,$$

$$\frac{1}{1 - \lambda S} + \sum_{i=1}^{N} \frac{p_i F / C_i}{1 - \lambda S} + \frac{D / B}{1 - \lambda S} \leq \tau,$$

where $\tau$ is the upper bound of the service response time.

In the optimization problem (9), the objective function is the total resource cost. The constraint, $\lambda < S$, represents
the queuing stability constraint at the master server. The constraints, \( p_i \lambda < C_i / F \) \((\forall i = 1, \ldots, N)\), represent the queuing stability constraints at the computing servers. The constraint, \( \lambda < B / D \), represents the queuing stability constraint at the transmission server. The constraint, \( \frac{1}{1 - \lambda / S} + \sum_{i=1}^{N} \frac{p_i F/C_i}{1 - \lambda / S} + \frac{D/B}{1 - \lambda D / B} \leq \tau \), represents the service response time constraint.

Similarly, we employ the Lagrange multiplier method [21] to solve the optimization problem (9), and get the optimal analytical solution as follows.

\[
S^* = \frac{\sqrt{\alpha} + \sqrt{\beta F} \sum_{i=1}^{N} \sqrt{p_i} + \sqrt{\gamma D}}{\sqrt{\gamma \tau}} + \lambda,
\]

\[
C^*_i = \frac{\left( \sqrt{\alpha} + \sqrt{\beta F} \sum_{i=1}^{N} \sqrt{p_i} + \sqrt{\gamma D} \right) \sqrt{p_i} F}{\sqrt{\beta \tau}} + p_i \lambda F,
\]

\[
\forall i = 1, \ldots, N,
\]

\[
B^* = \frac{\left( \sqrt{\alpha} + \sqrt{\beta F} \sum_{i=1}^{N} \sqrt{p_i} + \sqrt{\gamma D} \right) \sqrt{D}}{\sqrt{\gamma \tau}} + \lambda D.
\]

The optimal analytical solution (10) is obtained under the scheduling probability settings \( p_i \), \((\forall i = 1, 2, \ldots, N)\). Based on the resource cost optimization problem (9), we have the following theorem on the relationship between the resource cost and the scheduling probability settings.

**Theorem 2:** Given the resource cost optimization problem (9) in single-class service case based on the scheduling probability settings \( p_i \), \((\forall i = 1, \ldots, N)\), we have the following two statements: 1) when scheduling probability \( p_i \), \((\forall i = 1, \ldots, N)\), is 1 and all other probabilities \( p_j, (j \neq i, j = 1, \ldots, N) \) are 0, the minimum total resource cost is achieved; 2) when scheduling probabilities are all equal, i.e. \( p_1 = p_2 = \cdots = p_N = \frac{1}{N} \), the maximum total resource cost is achieved.

**Proof:**

Substituting the optimal analytical solution of \( S^*, C^*_i \), \((\forall i = 1, \ldots, N)\), and \( B^* \) in (10) to the objective function of the resource cost minimization problem (9), we can get the total resource cost as

\[
C_{\text{tot}}^{\text{sing}} = \frac{(\sqrt{\alpha} + \sqrt{\beta F} \sum_{i=1}^{N} \sqrt{p_i} + \sqrt{\gamma D})^2 T}{\tau} + (\alpha + \beta F + \gamma D) \lambda T - \frac{\left( \xi + \sqrt{\beta F} y \right)^2 T}{\tau} + \varpi,
\]

where \( \xi = \sqrt{\alpha} + \sqrt{\gamma D} > 0 \), \( y = \sum_{i=1}^{N} \sqrt{p_i} > 0 \), and \( \varpi = (\alpha + \beta F + \gamma D) \lambda T > 0 \).

We first prove that the resource cost \( C_{\text{tot}}^{\text{sing}} \) in Equation (11) is a monotonically increasing function of \( y \). Given \( y_1, y_2 \), and assuming \( 0 < y_1 < y_2 \), we can derive that \( \frac{\left( \xi + \sqrt{\beta F} y_1 \right)^2 T}{\tau} + \varpi < \frac{\left( \xi + \sqrt{\beta F} y_2 \right)^2 T}{\tau} + \varpi \), since \( \xi > 0, T > 0, \tau > 0 \) and \( \varpi > 0 \). So, the resource cost \( C_{\text{tot}}^{\text{sing}} \) in Equation (11) is a monotonically increasing function of \( y \). Thus, the minimum resource cost will be achieved at the minimum \( y \), while the maximum resource cost achieved at the maximum \( y \).

To achieve the minimal resource cost, we need to find the minimum \( y \). From the proof in Theorem 1, we can get that \( y = \sum_{i=1}^{N} \sqrt{p_i} \) reaches the minimum value 1 when probability \( p_i, (\forall i = 1, \ldots, N) \) is 1 and all other
probabilities $p_j, (j \neq i, j = 1, \ldots, N)$ are 0. Given the resource cost minimization problem (9) based on the scheduling probability settings, when scheduling probability $p_i, (\forall i = 1, \ldots, N)$, is 1 and all other probabilities $p_j, (j \neq i, j = 1, \ldots, N)$ are 0, we can achieve the minimum total resource cost, which is given as follows,

$$C_{singly}^{\text{tot}(\min)} = \frac{(\sqrt{\alpha} + \sqrt{\beta F} + \sqrt{\gamma D})^2T}{\tau} + (\alpha + \beta F + \gamma D)\lambda T. \quad (12)$$

To get the scheduling probability settings for the maximum resource cost, we need to find the maximum $y$. According to the proof in Theorem 1, $y = \sum_{i=1}^{N} p_i$ reaches the maximum value $\sqrt{N}$ when scheduling probability $p_1 = p_2 = \cdots = p_N = \frac{1}{N}$. Given the resource cost minimization problem (9) based on the scheduling probability settings, when scheduling probabilities are all equal, i.e., $p_1 = p_2 = \cdots = p_N = \frac{1}{N}$, the maximum total resource cost is achieved. The maximum total resource cost is given by

$$C_{singly}^{\text{tot}(\max)} = \frac{(\sqrt{\alpha} + \sqrt{\beta F} + \sqrt{\gamma D})^2T}{\tau} + (\alpha + \beta F + \gamma D)\lambda T. \quad (13)$$

We verify Theorem 2 in the following example. The number of computing servers is 3 and the scheduling probabilities are denoted as $p_1, p_2, p_3$, respectively. The service response time upper bound $\tau$ is set as 0.1 seconds. The parameters of the cost rates $\alpha, \beta, \gamma$, the occupation time $T$, the requests arrival rate $\lambda$, the task size $F$ and result size $D$ are configured the same as those in the example for Theorem 1 in Section IV-A1. Fig. 3 shows the resource cost with different scheduling probability settings. In Fig. 3, the scheduling probabilities $p_1, p_2$, and $p_3$ are all in the range of $[0, 1]$. Given $p_1$ and $p_2$ settings, $p_3$ can be determined by $1 - p_1 - p_2$. From Fig. 3,
we can find that the minimum resource cost 2987.8 dollars is achieved when the scheduling probability settings \(\langle p_1, p_2, p_3 \rangle\) are at \((1, 0, 0), (0, 1, 0)\) and \((0, 0, 1)\), while the maximum resource cost 3368.3 dollars is acquired at scheduling probability settings \(\langle \frac{1}{3}, \frac{1}{3}, \frac{1}{3} \rangle\), which justifies the statements in Theorem 2. Additionally, when \(p_1\) is fixed, the maximum resource cost is acquired when \(p_2 = p_3 = \frac{1-p_1}{2}\). Similar conclusion can be drawn when \(p_2\) or \(p_3\) is fixed.

### B. Multiple-Class Service Case

In this subsection, we study the resource allocation problem in multiple-class service case, in which multiple classes of application services are provided by the data centre. Each class of service has a different processing procedure at the computing server, a different number of instructions for the service task, a different size of the result, and a different requirement on the service response time.

In multiple-class service case, different classes of requests are first stored at the schedule queue, and then assigned to corresponding computing servers for processing. Suppose that there are \(M\) classes of services and the requests arrivals for each class follow a Poisson Process with mean arrival rates \(\lambda_1, \lambda_2, \ldots, \lambda_M\), respectively. According to Poisson composition property, the total arrivals of the requests follow a Poisson Process with an average arrival rate \(\lambda = \sum_{i=1}^{M} \lambda_i\). Thus, the schedule queue is modeled as an \(M/M/1\) queuing system with mean service rate equal to the the scheduling rate \(S\) of the master server. In order to maintain a stable queue, \(\lambda < S\) is required. The mean service response time of the schedule queue is given by \(T_{sche} = \frac{1}{S} \frac{1}{1-\lambda/S}\).

Since each class of service requires a different processing procedure, we assume \(M\) virtual machines are virtually generated as computing servers to provide the \(M\) classes of services correspondingly, in which computing server \(i\) specially serves the class-\(i\) service requests. Without loss of generality, we can also deploy \(K\) computing servers for one service, but as given by Theorems 1 and 2, the service response time and resource cost of using \(K\) computing servers to process one service are worse than those of using only one computing server which has resource capacity as the sum of all \(K\) computing servers. Thus, we only employ one computing server for each service. The requests arriving at the computing server \(i\) are all for class-\(i\) service, following a Poisson Process with average arrival rate \(\lambda_i\). The service at the computing server \(i\) is therefore modeled as an \(M/M/1\) queuing system with the mean service rate equal to \(C_i/F_i\). To maintain a stable queue, the constraint \(\lambda_i < C_i/F_i\) should be satisfied. The mean service response time at the computing server \(i\) is given by \(T_{comp}^{i} = \frac{F_i/C_i}{1-\lambda_i F_i/C_i}\). Thus, the mean service response time in computation phase is formulated by \(T_{mult}^{comp} = \sum_{i=1}^{M} \frac{\lambda_i}{S} T_{comp}^{i} = \sum_{i=1}^{M} \frac{\lambda_i F_i/C_i}{\lambda (1-\lambda_i F_i/C_i)}\).

After processing, all service results are sent to the transmission queue. Since a service result is generated for each request and the system is a close system, the average arrival rate of results at the transmission queue is also \(\lambda\). Each class of service has a different size of service result. The average size of result for class-\(i\) service is denoted by \(D_i\). The transmission time for the class-\(i\) service result is exponentially distributed with mean service time \(B_i^{-1} = D_i/B\), where \(B\) is the transmission rate at the transmission server. Therefore, the transmission queue can be viewed as a queuing system in which customers are grouped into a single arrival stream and the service distribution is a mixture of \(M\) exponential distributions. In fact, the service time follows hyper-exponential distribution [20]. The transmission
queue is actually an $M/H_M/1$ queuing system, where $H_M$ represents a hyperexponential-$M$ distribution. And the response time of the $M/H_M/1$ queuing system can be derived from $M/G/1$ queuing system [20]. The mean service rate at the transmission server for transmitting all $M$ classes results is given by $B\sum_{i=1}^{M} \frac{\lambda_i D_i}{\lambda B}$, and the mean service response time of transmission queue is formulated as $T_{\text{tran}} = \frac{1}{B^2 - B\sum_{i=1}^{M} \lambda_i D_i} + \frac{\sum_{i=1}^{M} \lambda_i D_i}{\lambda B}$. To ensure a stable queue, $\sum_{i=1}^{M} \lambda_i D_i < B$ is required.

Based on the analysis of service response time in schedule, computation and transmission phases, we can then get the total service response time in the multiple-class service case as follows.

$$T_{\text{tot}} = T_{\text{sche}} + T_{\text{comp}} + T_{\text{tran}}$$

$$= \frac{1}{S} + \sum_{i=1}^{M} \frac{\lambda_i D_i}{B^2 - B\sum_{i=1}^{M} \lambda_i D_i}$$

Moreover, the mean service response time for class-$i$ service in the data center is formulated as

$$T_{\text{tot}}^{(i)} = T_{\text{sche}} + T_{\text{comp}} + T_{\text{tran}}$$

$$= \frac{1}{S} + \frac{\lambda_i D_i}{B^2 - B\sum_{i=1}^{M} \lambda_i D_i}$$

And the total resource cost for the multiple-class service can be formulated as

$$C_{\text{tot}} = \left(\alpha S + \beta \sum_{i=1}^{M} C_i + \gamma B\right) T.$$  

1) Service Response Time Minimization Problem: Since there exist different types of multimedia applications, such as video conference, cloud TV, and 3D rendering, multimedia cloud should supply different types of multimedia services for users simultaneously. But it’s challenging to provide various multimedia services with minimum total service response time under certain cost constraint. Therefore, we formulate the service response time minimization problem, which can be stated as: to minimize the total service response time in multiple-class service case by jointly optimizing the scheduling rate at the master server, the computing rate at each computing server, and the transmission rate at the transmission server, subject to the queuing stability constraint in each queuing system and the resource cost constraint. Mathematically, the problem can be formulated as follows.

Minimize

$$T_{\text{tot}} = \frac{1}{S} + \sum_{i=1}^{M} \frac{\lambda_i D_i}{B^2 - B\sum_{i=1}^{M} \lambda_i D_i}$$

subject to

$$\lambda < S,$$

$$\lambda_i F_i < C_i, \quad \forall i = 1, \ldots, M,$$

$$\sum_{i=1}^{M} \lambda_i D_i < B,$$

$$(\alpha S + \beta \sum_{i=1}^{M} C_i + \gamma B)T \leq C_{\text{max}}.$$
where $C_{\text{max}}$ is the upper bound of the resource cost.

In the optimization problem (17), the objective function is the total service response time in multiple-class service case. The constraint, $\lambda < S$, represents the queuing stability constraint at the master server. The constraints, $\lambda_i F_i < C_i$ ($\forall i = 1, \ldots, M$), represent the queuing stability constraints at the computing servers. The constraint, $\sum_{i=1}^{M} \lambda_i D_i < B$, represents the queuing stability constraint at the transmission server. The constraint, $(\alpha S + \beta \sum_{i=1}^{M} C_i + \gamma B)T \leq C_{\text{max}}$, represents the resource cost constraint.

The service response time minimization problem (17) is a convex optimization problem [21]. Efficient solution methods for convex optimization problems are well developed. In this paper, we use the primal-dual interior-point methods [21] to solve the convex optimization problem (17).

Furthermore, based on the service response time minimization problem (4) in single-class service case and (17) in multiple-class service case, we have the following theorem.

**Theorem 3**: The minimum service response time obtained in the optimization problem (4) for the single-class service case is equal to the minimum service response time obtained in the optimization problem (17) for the multiple-class service case with the number of the classes equal to 1.

**Proof**: When the number of classes is equal to 1 in multiple-class service case (suppose that the only provided service is class-$i$ service), the total requests arrival rate $\lambda = \lambda_i$, and according to Equation (14) the total service response time can be formulated as follows.

$$T_{\text{mult}} = \frac{1}{S} + \frac{\lambda_i F_i / C_i}{1 - \lambda_i / S} + \frac{\lambda_i D_i}{B^2 - B \lambda_i D_i} + \frac{\lambda_i D_i}{\lambda_i B}$$

$$= \frac{1}{S} + \frac{F_i / C_i}{1 - \lambda_i / S} + \frac{D_i / B}{1 - \lambda_i D_i / B}.$$

Therefore, the optimization problem (17) can be changed to the following:

$$\text{Minimize} \quad T_{\text{mult}} = \frac{1}{S} + \frac{F_i / C_i}{1 - \lambda_i / S} + \frac{D_i / B}{1 - \lambda_i D_i / B}$$

subject to

$$\lambda_i < S,$$

$$\lambda_i F_i < C_i,$$

$$\lambda_i D_i < B,$$

$$(\alpha S + \beta C_i + \gamma B)T \leq C_{\text{max}}.$$

Comparing (18) with the optimization problem (4) in single-class service case, we can find that when the scheduling probability settings in (4) are $p_i = 1, (\forall i = 1, \ldots, N)$, $p_j = 0, (j \neq i, j = 1, \ldots, N)$, the optimization problem (4) is identical to the optimization problem (18), and according to Theorem 1, the minimum service response time will be achieved under such scheduling probability settings in (4). Therefore, the minimum service response time obtained in the optimization problem (4) for the single-class service case is equal to the minimum service response time obtained in the optimization problem (17) for the multiple-class service case with the number of the classes equal to 1.

\[ \diamond \]
2) Resource Cost Minimization Problem: As different types of multimedia applications have different service response time requirements, the multimedia cloud not only supplies various multimedia services, but also provides QoS provisioning to satisfy these different requirements on service response time. However, it’s challenging for cloud service providers to satisfy all service response time requirements at the minimum resource cost. Thus, we formulate the resource cost minimization problem, which can be stated as: to minimize total resource cost in multiple-class service case by jointly optimizing the scheduling rate at the master server, the computing rate at each computing server, and the transmission rate at the transmission server, subject to the queuing stability constraint in each queuing system and the constraint on the service response time for each class of service. Mathematically, the problem can be formulated as follows.

\[
\begin{align*}
\text{Minimize} & \quad C_{\text{tot mult}}^\text{S} = (\alpha S + \beta \sum_{i=1}^{M} C_i + \gamma B) T \\
\text{subject to} & \quad \lambda < S, \\
& \quad \lambda_i \lambda_i F_i < C_i, \quad \forall i = 1, \ldots, M, \\
& \quad \sum_{i=1}^{M} \lambda_i D_i < B, \\
& \quad \frac{1}{1 - \lambda_i} + \frac{F_i}{C_i} + \frac{\sum_{i=1}^{M} \lambda_i D_i^2}{B^2} \sum_{i=1}^{M} \lambda_i D_i + \frac{\sum_{i=1}^{M} \lambda_i D_i}{\lambda B} \leq \tau_i, \quad \forall i = 1, \ldots, M,
\end{align*}
\]

where \(\tau_i\) is the upper bound of the service response time for class-\(i\) service.

In the optimization problem (19), the objective function is the total resource cost in multiple-class service case. The constraint, \(\lambda < S\), represents the queuing stability constraint at the master server. The constraints, \(\lambda_i F_i < C_i, \quad \forall i = 1, \ldots, M\), represent the queuing stability constraints at the computing servers. The constraint, \(\sum_{i=1}^{M} \lambda_i D_i < B\), represents the queuing stability constraint at the transmission server. The constraints, \(\frac{1}{1 - \lambda_i} + \frac{F_i}{C_i} + \frac{\sum_{i=1}^{M} \lambda_i D_i^2}{B^2} \sum_{i=1}^{M} \lambda_i D_i + \frac{\sum_{i=1}^{M} \lambda_i D_i}{\lambda B} \leq \tau_i, \quad (\forall i = 1, \ldots, M)\), represent the service response time constraints of class-\(i\) service.

The resource cost minimization problem (19) is also a convex optimization problem, which can be solved efficiently using the primal-dual interior-point methods [21]. To examine the relationship between the resource cost optimization problem (9) in single-class service case and (19) in multiple class service case, we provide the following theorem.

**Theorem 4**: The minimum resource cost obtained in the optimization problem (9) for the single-class service case is equal to the minimum resource cost obtained in the optimization problem (19) for the multiple-class service case with the number of the classes equal to 1.

**Proof**: When there is only one class of service (suppose that the only provided class is class-\(i\) service), the resource cost
minimization problem (19) is changed to the following:

\[
\begin{align*}
\text{Minimize} & \quad C^\text{tot}_\text{mult} = (\alpha S + \beta C_i + \gamma B) T \\
\text{subject to} & \quad \lambda_i < S, \\
& \quad \lambda_i F_i < C_i, \\
& \quad \lambda_i D_i < B, \\
& \quad \frac{1}{1 - \lambda_i / S} + \frac{F_i / C_i}{1 - \lambda_i F_i / C_i} + \frac{D_i / B}{1 - \lambda_i D_i / B} \leq \tau_i.
\end{align*}
\]  

(20)

Comparing (20) with the optimization problem (9) in single-class service case, we can find that the optimization problem (9) is the same as the optimization problem (20) when the scheduling probability settings in (9) are \(p_i = 1, (\forall i = 1, \ldots, N)\), \(p_j = 0, (j \neq i, j = 1, \ldots, N)\). According to Theorem 2, the minimum resource cost in (9) is achieved under such scheduling probability settings. Therefore, the minimum resource cost obtained in the optimization problem (9) for the single-class service case is equal to the minimum resource cost obtained in the optimization problem (19) for the multiple-class service case with the number of the classes equal to 1.

\[\Box\]

C. Priority Service Case

The master server, computing servers and transmission server in the multiple-class service case process requests in the first-come first-served (FCFS) order, which is not suitable for the urgent applications, such as real-time health monitoring. In this subsection, we study the resource allocation at the multimedia cloud in priority service case, in which multiple classes of services with different priorities are simultaneously provided by the data center, and the requests for the higher-priority services are processed ahead of those for the lower-priority services.

In the priority service case, we employ the preemptive priority queuing discipline, in which the requests with a higher priority obtain the service immediately even if other requests with a lower priority are being served, and the preempted requests will be later resumed from the last preemption point.

Suppose that there are \(M\) classes of services with different priorities, which are denoted as class-1, 2, \ldots, \(M\), respectively. A smaller class number corresponds to a higher priority. The arrivals of the class-\(i\) requests are modeled as a Poisson Process with average arrival rate \(\lambda_i\). According to Poisson composition property, the total arrivals of all requests also follow a Poisson Process with average arrival rate \(\lambda = \sum_{i=1}^{M} \lambda_i\). When arriving at data center, all requests are firstly stored in the schedule queue. The master server always schedules the requests with the highest priority first. The lower-priority requests can be scheduled only after all higher-priority requests have left the schedule queue. The schedule queue is modeled as an \(M/M/1\) queuing system with preemptive priority service. Therefore, the mean service response time for scheduling class-\(i\) priority requests is given by

\[
T^\text{sche}_{\text{prio}}^{\text{(i)}} = \frac{1}{1 - \sigma^\text{sche}^{(i)}} + \frac{\sum_{j=1}^{i-1} (\lambda_j / S^2)}{(1 - \sigma^\text{sche}^{(i)})(1 - \sigma^\text{sche}^{(i)})}, \quad (\forall i = 1, \ldots, M),
\]

where \(\sigma^\text{sche}^{(i)} = \sum_{j=1}^{i} \lambda_j / S\). To make the schedule queue stable, \(\sigma^\text{sche} = \sum_{j=1}^{M} \lambda_j / S < 1\) should be satisfied. Since the service rate distributions are the same for all classes of requests in the schedule queue, the total response time is the same as an \(M/M/1\) queue. Thus, the mean service response time at the master server is given by

\[
T^\text{sche}_{\text{prio}} = \frac{1}{1 - \lambda / S}.
\]
In the computing phase, there are $M$ computation queues to store the requests with the corresponding priority. Moreover, the total computing resources are aggregated as a computing server. Requests with the highest priority have preemptive right to obtain service immediately. The total computing resource is denoted as $C$, and the average task size of class-$i$ service is $C_i$. The service time for computing class-$i$ requests is assumed to be exponentially distributed with mean time of $C_i/C$. Thus, the service at the computing server is modeled as a preemptive priority queuing system with unequal service rates for multiple classes customers. According to [22], the mean service response time for computing the requests with class-$i$ priority is given by $T_{comp(i)}^{\text{prior}} = \frac{C_i/C}{1 - \sigma_{comp(i)}^2} + \sum_{i=1}^M \left( \frac{\lambda_i C_i^2}{C^2(1 - \sigma_{comp(i)}^2)} \right)$, $(\forall i = 1, \ldots, M)$, where $\sigma_{comp}^2 = \sum_{i=1}^M \lambda_i C_i^2$, and $\sigma_{comp}^2 = \sum_{j=1}^M \frac{\lambda_j C_j^2}{C^2} < 1$ should be satisfied to maintain a stable queue. Since the service rate distributions are unequal for different classes requests, the mean service response time at computing server is given by [22] $T_{comp}^{\text{prior}} = \sum_{i=1}^M \frac{\lambda_i T_{comp(i)}^{\text{prior}}}{\lambda}$.

After processing, all service results are sent to the transmission queue. The results of higher-priority classes are transmitted prior to those of lower-priority classes. The transmission rate at the transmission server is $B$, and the average result size of class-$i$ service is $D_i$. The mean transmission time for class-$i$ result is exponentially distributed with mean service time $D_i/B$. According to [22], the mean response time for transmitting service results with class-$i$ priority is given by $T_{\text{tran}(i)}^{\text{priority}} = \frac{D_i/B}{1 - \sigma_{\text{tran}(i)}^2} + \sum_{i=1}^M \left( \frac{\lambda_i D_i^2}{B^2(1 - \sigma_{\text{tran}(i)}^2)} \right)$, $(\forall i = 1, \ldots, M)$, where $\sigma_{\text{tran}}^2 = \sum_{i=1}^M \frac{\lambda_i D_i^2}{B^2}$. To ensure the transmission queue stable, $\sigma_{\text{tran}}^2 = \sum_{j=1}^M \frac{\lambda_j D_j^2}{B^2} < 1$ should be satisfied. Thus, the mean service response time at the transmission server is formulated as $T_{\text{tran}}^{\text{prior}} = \sum_{i=1}^M \frac{\lambda_i T_{\text{tran}(i)}^{\text{prior}}}{\lambda} = \sum_{i=1}^M \left( \frac{\lambda_i D_i/B}{\lambda(1 - \sigma_{\text{tran}(i)}^2)} + \frac{\lambda_i D_i^2/B^2}{\lambda(1 - \sigma_{\text{tran}(i)}^2)} \right)$.

The total service time in multimedia data center is formulated as follows.

$$T_{\text{total}}^{\text{prior}} = T_{\text{total}}^{\text{prior}} + T_{\text{comp}}^{\text{prior}} + T_{\text{tran}}^{\text{prior}}$$

$$= \frac{1}{\lambda} + \sum_{i=1}^M \frac{\lambda_i \sum_{j=1}^i \frac{\lambda_j C_i^2}{C^2}}{\lambda(1 - \sum_{j=1}^{i-1} \frac{\lambda_j C_j^2}{C^2})(1 - \sum_{j=1}^i \frac{\lambda_j C_j^2}{C^2})} + \sum_{j=1}^M \frac{\lambda_j B_j D_j^2}{B^2} \frac{\lambda_j}{\lambda(1 - \sum_{j=1}^{i-1} \frac{\lambda_j B_j D_j^2}{B^2})(1 - \sum_{j=1}^j \frac{\lambda_j B_j D_j^2}{B^2})}$$

$$= \frac{1}{\lambda} + \sum_{i=1}^M \frac{\lambda_i \sum_{j=1}^i \frac{\lambda_j C_i^2}{C^2}}{\lambda(1 - \sum_{j=1}^{i-1} \frac{\lambda_j C_j^2}{C^2})(1 - \sum_{j=1}^i \frac{\lambda_j C_j^2}{C^2})} + \sum_{j=1}^M \frac{\lambda_j B_j D_j^2}{B^2} \frac{\lambda_j}{\lambda(1 - \sum_{j=1}^{i-1} \frac{\lambda_j B_j D_j^2}{B^2})(1 - \sum_{j=1}^j \frac{\lambda_j B_j D_j^2}{B^2})}$$

Furthermore, we can formulate the mean service response time for processing class-$i$ priority service as follows.

$$T_{\text{total}}^{\text{prior}} = T_{\text{total}}^{\text{prior}} + T_{\text{comp}}^{\text{prior}} + T_{\text{tran}}^{\text{prior}}$$

$$= \frac{1}{\lambda} + \sum_{i=1}^M \frac{\lambda_i \sum_{j=1}^i \frac{\lambda_j C_i^2}{C^2}}{\lambda(1 - \sum_{j=1}^{i-1} \frac{\lambda_j C_j^2}{C^2})(1 - \sum_{j=1}^i \frac{\lambda_j C_j^2}{C^2})} + \sum_{j=1}^M \frac{\lambda_j B_j D_j^2}{B^2} \frac{\lambda_j}{\lambda(1 - \sum_{j=1}^{i-1} \frac{\lambda_j B_j D_j^2}{B^2})(1 - \sum_{j=1}^j \frac{\lambda_j B_j D_j^2}{B^2})}$$

$$\forall i = 1, 2, \ldots, M.$$
In addition, the total resource cost in priority service case is formulated as

\[ C_{\text{tot}}^{\text{prio}} = (\alpha S + \beta C + \gamma B)T. \]  

(23)

1) Service Response Time Minimization Problem: In multimedia cloud, priority queuing discipline has been used in many applications. The cloud providers need to support different priority services and minimize the average service response time for all service requests. The service response time minimization problem in priority service case can be stated as: to minimize the mean service response time for all requests in priority service case by jointly optimizing the scheduling rate at the master server, the computing rate at the computing server, and the transmission rate at the transmission server, subject to the queuing stability constraint in each queuing system and the resource cost constraint. Mathematically, the service response time minimization problem can be formulated as follows.

Minimize \[ T_{\text{tot}}^{\text{prio}} \]

subject to

\[ \lambda < S, \]
\[ \sum_{i=1}^{M} \lambda_i F_i < C, \]
\[ \sum_{i=1}^{M} \lambda_i D_i < B, \]
\[ (\alpha S + \beta C + \gamma B)T \leq C_{\text{max}}, \]

(24)

where \[ T_{\text{tot}}^{\text{prio}} \] is given in Equation (21).

In the optimization problem (24), the objective function is the total service response time in priority service case. The constraint, \( \lambda < S \), represents the queuing stability constraint at the master server. The constraint, \( \sum_{i=1}^{M} \lambda_i F_i < C \), represents the queuing stability constraint at the computing server. The constraint, \( \sum_{i=1}^{M} \lambda_i D_i < B \), represents the queuing stability constraint at the transmission server. The constraint, \( (\alpha S + \beta C + \gamma B)T \leq C_{\text{max}} \), represents the resource cost constraint. The service response time minimization problem (24) in priority service case is also a convex optimization problem, which can be solved efficiently using the primal-dual interior-point methods [21].

By examining the priority service case and the multiple-class service case, we have the following theorem.

**Theorem 5:** With the same resource cost constraint, the same requests arrival rate, the same task size and result size, the service response time of the highest priority service (class-1 service) in priority service case is always no larger than the service response time of the same service (class-1 service) in multiple-class service case.

**Proof:**

To prove Theorem 5, we first prove that the optimal solution for optimization problem (17) is a candidate solution for optimization problem (24). Suppose that the optimal resource allocation for optimization problem (17) is \( \{ S, C_1, \ldots, C_M, B \} \), which satisfies all constraints in optimization problem (17). With the same resource cost constraint, the same request arrival rate, the same task size and result size, the corresponding resource allocation \( \{ S, C = \sum_{i=1}^{M} C_i, B \} \) is a candidate solution for optimization problem (24), since it can meet all constraints in problem (24). Therefore, with the optimal resource allocation \( \{ S, C_1, \ldots, C_M, B \} \), the service response time for processing class-1 service in multiple-class service case is given by \( T_{\text{mult}}^{(1)} = T_{\text{sche}}^{(1)} + T_{\text{comp}}^{(1)} + T_{\text{tran}}^{(1)} = \)
The service response time for processing class-1 service in priority service case is given by

\[ T_{prio}^{\text{tot}} = T_{prio}^{\text{sche}} + T_{prio}^{\text{comp}}(1) + T_{prio}^{\text{tran}(1)} \]

Next we will compare the service response time of schedule, computation, and transmission in \( T_{prio}^{\text{tot}(1)} \) and those in \( T_{prio}^{\text{tot}} \), respectively. In schedule phase, since \( \lambda = \sum_{i=1}^{M} \lambda_i \geq \lambda_1 \), \( T_{prio}^{\text{sch}(1)} = \frac{1}{S-\lambda_1} \geq T_{prio}^{\text{sch}(1)} = \frac{1}{S-\lambda} \); in computation phase, since \( C_1 \geq \sum_{i=1}^{M} C_i \), \( T_{mult}^{\text{comp}(1)}(1) = \frac{F_i}{c_1-\lambda_1 F_1} \geq T_{prio}^{\text{comp}(1)} = \frac{F_i}{\sum_{i=1}^{M} c_1-\lambda_1 F_1} \); in transmission phase, according to the conclusion in [20] that the class with the highest priority in priority queue has smaller service response time than its counterpart in \( M/H_M/1 \) queue, we can get that \( T_{mult}^{\text{trans}(1)} = \frac{\sum_{i=1}^{M} \lambda_i D_i^2}{B^2-B \sum_{i=1}^{M} \lambda_i D_i} + \frac{\sum_{i=1}^{M} \lambda_i D_i}{MB} \geq T_{prio}^{\text{trans}(1)} = \frac{D_1}{B-\lambda_1 D_1} \). Therefore, based on the above comparisons, we can get that \( T_{mult}^{\text{tot}(1)} \geq T_{prio}^{\text{tot}(1)} \). Therefore, with the same resource cost constraint, the same requests arrival rate, the same task size and result size, the service response time of the highest priority service (class-1 service) in priority service case is always no larger than the service response time of the same service (class-1 service) in multiple-class service case.

With priorities, the mean number of higher priority customers waiting in the queue decreases while the mean number of lower priority customers increases [20]. Moreover, to address the issue how the imposition of priorities affects the overall service response time, Schrage and Miller [23] propose the shortest processing time (SPT) rule, which is described as follows. If the design objective of a queue is to reduce the total number of waiting customers, or equivalently the overall mean delay, the priority should be given to the group of customers that has the faster service rate. In our work, the SPT rule is that the higher priorities in computation queue should be given to the services with smaller task size, while the higher priorities in transmission queue should be given to the services with smaller result size. Therefore, if the overall objective in multimedia cloud data center is to reduce the service response time for one specific service, like the health monitoring, this service should be given the highest priority. However, if the objective is to reduce the average delay in data center for all customers, the SPT rule should be employed to give a higher priority to the class with a faster service rate.

2) Resource Cost Minimization Problem: In multimedia cloud, different priority services have different requirements on service response time. It is challenging for cloud providers to support multiple QoS provisioning to meet all requirements on service response time at the minimum cost. Therefore, we formulate the resource cost minimization problem in priority service case, which can be stated as: to minimize the total resource cost for all requests in priority service case by jointly optimizing the scheduling rate at the master server, the computing rate at the computing server, and the transmission rate at the transmission server, subject to the stability constraint in each queuing system and the service response time constraint for each class of service. Mathematically, the resource

\[
\frac{1}{\lambda} + \frac{F_i}{c_1-\lambda_1 F_1} + \frac{\sum_{i=1}^{M} \lambda_i D_i^2}{B^2-B \sum_{i=1}^{M} \lambda_i D_i} + \frac{\sum_{i=1}^{M} \lambda_i D_i}{MB}.
\]
cost minimization problem can be formulated as follows.

\[
\text{Minimize } \quad C_{\text{tot, prior}} = (\alpha S + \beta C + \gamma B)T
\]

subject to

\[
\begin{align*}
\lambda &< S, \\
\sum_{i=1}^{M} \lambda_i F_i &< C, \\
\sum_{i=1}^{M} \lambda_i D_i &< B, \\
T_{\text{tot, prior}}^{(i)} &\leq \tau_i, \quad \forall i = 1, \ldots, M,
\end{align*}
\]

where \(T_{\text{tot, prior}}^{(i)}\), given in Equation (22), is the service response time for class-\(i\) service, and \(\tau_i\) is the upper bound of the service response time for class-\(i\) service.

In the optimization problem (25), the objective function is the total resource cost in priority service case. The constraint, \(\lambda < S\), represents the queuing stability constraint at the master server. The constraint, \(\sum_{i=1}^{M} \lambda_i F_i < C\), represents the queuing stability constraint at the computing server. The constraint, \(\sum_{i=1}^{M} \lambda_i D_i < B\), represents the queuing stability constraint at the transmission server. The constraints, \(T_{\text{tot, prior}}^{(i)} \leq \tau_i, \quad \forall i = 1, \ldots, M\), represent the service response time constraints of class-\(i\) service. The resource cost minimization problem (25) can also be solved efficiently using the primal-dual interior-point methods [21].

V. SIMULATIONS

A. Simulations for Single-Class Service Case

1) Simulation Settings: We perform simulations to evaluate the proposed methods in the single-class service case. Windows Azure [5] is a cloud platform developed by Microsoft, which provides on-demand computation, storage, and networking resources for services through Microsoft data centers. We employ the device configuration and pricing rate of Microsoft Azure in our simulations. One medium instance server is employed as the master server to schedule requests. According to Theorem 1 and Theorem 2, the minimum total service response time and resource cost are achieved when only one scheduling probability is 1 and all others are 0. Thus, we employ only one extra large instance server as computing server in single-class service case to process all service requests. The cloud resources of the master server, the computing server, and the transmission server are charged by the scheduling cost rate \(\alpha = 5 \times 10^{-4}\) dollars/request, the computing cost rate \(\beta = 6 \times 10^{-6}\) dollars/MIs, and the transmission cost rate \(\gamma = 0.08\) dollars/gigabit, respectively. The occupied time \(T\) is set to 1 hour. The mean arrival rate \(\lambda\) of customers’ requests is set in the range of 50 to 150 requests/s. In single-class service case, the task size \(F\) of the service is set as 500 MIs, and the result size \(D\) is set as 15 megabit.

2) Simulation Results: We first compare the performance between the proposed optimal allocation scheme, in which the resources for the master, computing, and transmission servers are optimally allocated by solving the optimization problem (4) or (9), and the equal allocation scheme, in which the resource cost for schedule, computation, and transmission are allocated equally.
Comparison of the mean service response time between the proposed optimal allocation scheme and the equal allocation scheme is shown in Fig. 4. The resource cost constraint is set to $C_{max} = 5000$ dollars. When the requests arrival rate reaches 150 requests/s, the proposed allocation scheme achieves a much smaller service response time compared to the equal allocation scheme under the same resource cost constraint. Fig. 5 shows the service rate of schedule, computation, and transmission in two schemes when $\lambda$ is set to 150 requests/s. As shown in Fig. 5, the equal allocation scheme allocates a smaller portion of resource in the computation phase, thus leading to a higher service response time.

We next evaluate the resource cost between the proposed optimal allocation scheme and the equal allocation scheme in the data center. We set the upper bound of the service response time $\tau = 0.1$ seconds. From Fig. 6, we can see that the proposed optimal allocation scheme achieves a much lower resource cost compared to the equal allocation scheme under same service response time constraint. We can get the reason from the service rates shown in Fig. 7, when arrival rate is 150 requests/s. Compared to the proposed optimal allocation scheme, a large amount of resource is deployed for schedule and transmission in the equal allocation scheme. Thus customer’s requests cannot be computed in time and all tasks have to wait longer in computation queue, which degrades the system performance.
B. Simulations for Multiple-Class Service Case

1) Simulation Settings: In this subsection, we perform simulations to evaluate our proposed resource allocation methods in the multiple-class service case. Five classes of services are provided in the data center. Each class of service has a different arrival rate, a different task size, a different result size, and a different requirement on service response time. Table I shows the settings of the percentage of arrival rate, task size, result size and the upper bound of service time for each class. The pricing rates and the range of total arrival rate are the same as those configured in Section V-A1.

<table>
<thead>
<tr>
<th>Service Class</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Percentage of arrival rate</td>
<td>10%</td>
<td>15%</td>
<td>20%</td>
<td>25%</td>
<td>30%</td>
</tr>
<tr>
<td>Task size (Mh)</td>
<td>300</td>
<td>350</td>
<td>400</td>
<td>450</td>
<td>500</td>
</tr>
<tr>
<td>Result size (megabit)</td>
<td>7</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>Upper bound of service response time (seconds)</td>
<td>0.1</td>
<td>0.15</td>
<td>0.2</td>
<td>0.25</td>
<td>0.2</td>
</tr>
</tbody>
</table>

2) Simulation Results: We first compare the performance between the proposed optimal allocation scheme, in which the resources for schedule, computation, and transmission are allocated optimally by solving the optimization problem (17) or (19), and the equal allocation scheme, in which the resource cost for schedule, computation, and transmission are allocated equally.

Fig. 8 shows the comparison of service response time. The resource cost constraint is still set to 5000 dollars for the one hour period. From Fig. 8, we can see that the proposed optimal allocation scheme takes a smaller service response time than the equal allocation scheme under the same resource constraint. Fig. 9 shows the detailed service rates when the arrival rate of requests is 150 requests/s. The master server is allocated too much resource in the equal allocation scheme, which results in the less resource in the computing servers and the transmission server.
The comparison of resource cost in the multiple-class service case is shown in Fig. 10, from which we can see that the proposed optimal allocation scheme achieves a much lower resource cost than the equal allocation scheme. The detailed service rate of each server is shown in Fig. 11, when arrival rate is 150 requests/s. Compared with the proposed scheme, the equal allocation scheme assigns less resource on the transmission server, which causes longer waiting time in transmission queue and degrades the system performance.

C. Simulations for Priority Service Case

1) Simulation Settings: In this subsection, we perform simulations to evaluate the proposed resource allocation method in the priority service case. Five classes of services with different priorities are provided in the data center. The percentage of arrival rate, the task size, the result size and the upper bound of response time of each class are the same as the parameter settings in Table I. Moreover, a smaller class number corresponds to a higher priority. The class-1 service has the highest priority for service.

2) Simulation Results: We compare the mean service response time and resource cost in priority service case between the proposed optimal resource allocation scheme, in which the resources for schedule, computation, and transmission are allocated by solving the optimization problem (24) or (25), and the equal allocation scheme, in which the resource cost for schedule, computation and transmission are allocated equally.
Fig. 14. Comparison of resource cost between the proposed optimal allocation scheme and the equal allocation scheme in priority service case.

Fig. 15. Comparison of the service rates between the proposed optimal allocation scheme and the equal allocation scheme in the resource cost minimization problem in priority service case.

Fig. 12 shows the comparison of the mean service response time. In the simulation, the resource cost constraint is still set to 5000 dollars for one hour period. From Fig. 12, we can find that the proposed resource allocation scheme can take a smaller service response time than the equal allocation scheme under the same resource constraint. Fig. 13 shows the detailed resource allocation when the arrival rate of requests is 150 requests/s. From Fig. 13, we can see that the equal allocation scheme allocates a smaller service rate in the computation phase, which results in a larger service response time.

The comparison of the resource cost in priority service case is shown in Fig. 14, from which we can see that the proposed optimal resource allocation scheme achieves a much lower resource cost than the equal allocation scheme. The detailed resource allocation is shown in Fig. 15. The equal allocation scheme has a higher cost because much resource is allocated in the master server and the transmission server while less resource is allocated in the computing server.

VI. Conclusions

In this paper, we study the resource optimization problem in multimedia cloud. We first model the service process at multimedia cloud data center as three concatenated queuing systems, which are schedule queue, computation queue and transmission queue. We then employ the queuing models to study the relationship between the service response time and the allocated resources. Moreover, we examine the resource allocation problem in single-class service case, multiple-class service case and priority service case, respectively. In each case, we formulate and solve the service response time minimization problem and resource cost minimization problem, respectively. The simulation results demonstrate that the proposed optimal allocation scheme can improve the performance of multimedia cloud data center in terms of service response time or resource cost compared to the equal allocation scheme.

References


