IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received October 17, 2018, accepted November 6, 2018, date of publication November 29, 2018,
date of current version January 11, 2019.

*Digital Object Identifier 10.1109/ACCESS.2018.2883964*

# Minimizing Geo-Distributed Interactive Service Cost With Multiple Cloud Service Providers

**FEI HU[1], QINGCHUN LIU[1], JIAHONG WU[2], AND JIANGUO YAO[2], (Senior Member, IEEE)**
[1]Science and Technology on Avionics Integration Laboratory, Shanghai 200233, China
[2]School of Software, Shanghai Jiao Tong University, Shanghai 200240, China

Corresponding author: Jianguo Yao (jianguo.yao@sjtu.edu.cn)

The work of J. Wu was supported in part by the National Key Research and Development Program of China under Grant 2018YFB1003603, in part by the Medical Cross-Fund of SJTU under Grant YG2015MS47, in part by the Program for NSFC under Grant 61772339, and in part by the Shanghai Rising-Star Program under Grant 16QA1402200.

**ABSTRACT** Geo-distributed interactive service is a type of Internet cloud service. Since data centers belonging to multiple cloud service providers (CSPs) can provide lower resource provision cost for these services, we try to overcome the two challenges in selecting the data centers from multi-CSPs for our cost optimization. First, as the resource prices vary across data centers, scheduling data centers and geo-distributed interactive services have a tremendous consequence for optimizing the cost. Second, the resource demand is uncertain, and the geo-distributed interactive service needs enough resources to meet the quality-of-service constraint. A regularized stochastic decomposition algorithm, namely, two-dimension resource provision (2DRP), is proposed for solving the two-stage stochastic linear programming problems. In stage one, users express tail latency constraints for each geo-distributed interactive service, letting 2DRP choose the lowest price-per-instance data center group and determine the right amount of resources. In stage two, 2DRP estimates the impact of the request rate and the amount of resources on tail latency and uses the decomposition technique to adjust the resource provision plan to handle the demand uncertainty. We evaluate 2DRP over a set of workload scenarios on real clouds by reproducing the Internet request with data center workload traces, and the results show the availability of our algorithm for cost minimization geo-distributed interactive services with multi-CSPs.

**INDEX TERMS** Cloud computing, cost minimization, geo-distributed interactive service, tail latency.

## I. INTRODUCTION

These days, companies and organizations in diverse industries have adopted MapReduce-based systems for interactive service [1]. Important new workloads in the form of geo-distributed interactive service have emerged, which feature geographically distributed and latency-sensitive. For instance, scalable geo-distributed interactive service on data sets has become the main concern for several teams at Google [2]. Some of them aim at creating business intelligence for marketing which have features ranging from simple Internet Cloud Services (such as Google Analytics Real-time) to more advanced types, such as Insights for the Google Advertisers.

Apart from the Google, the flexible and cost-effective Cloud infrastructure becomes significant for Cloud consumers like common companies and organizations to handle the problems of the growing number of data volumes, latency-sensitive characteristic and resource provision cost minimization [3], [4]. Besides, there are some other requirements. Cloud consumer requires less stringent latency constraint so as to save resource provision cost [5]; Cloud consumers prefer the lower price data centers although these data centers belong to multi-CSPs; resource demand depends on the service request rate. These requirements determine the Cloud infrastructure must have the capability to provide optimized distribution and resource provision plan for the geo-distributed interactive services, multiple Cloud Service Providers (multi-CSPs), diverse resource price policies and changing resource demand.

These works [6]–[8] have made encouraging progress in optimizing the resource provision cost. However, multi-CSPs environment and diverse resource pricing policies, even for single geo-distributed interactive service, result in non-convex optimization and no efficient solution for the distributions and resource provision planning. Existing researches [9], [10] require working with specified service

2169-3536 © 2018 IEEE. Translations and content mining are permitted for academic research only.
Personal use is also permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

profiles and making assumptions about which interactive service will be distributed to where, but both factors are more dynamic than expected. Resource prices differ from multi-CSPs and shift constantly, which causes the assumptions about the distribution can become ineffective in months. So, interactive service must switch to data center group with the lower price in regular interval to maintain that the resource provision cost is minimized. Also, the Internet request rates of interactive service change dynamically, and the allocated resource must meet the resource demand under different request rates [11], [12]. A comprehensive cost model and a credible tail latency model are necessary for adjusting the resource provision plan.

To summarize, our objective is to minimize the resource provision cost for geo-distributed interactive service with multi-CSPs. There are two challenges that we have to consider. The resource provision cost varies widely with diverse resource price policies, and the geo-distributed interactive service needs to meet the Quality of Service (QoS) constraints with changing resource demand. Besides, CSPs do not provide a centralized view of semantic-rich interaction service distribution. The ideal algorithm needs to decide where to place the interactive service's workloads, how many resources are needed to meet the QoS constraint. We overcome the two challenges by optimizing the distribution and resource provision planning. The key intuition is by balancing the price, the number of resources and the tail latency. The distribution and the resource provisioning can respectively be formulated as a master problem and finite scenario subproblems by modeling the uncertain resource price and the uncertain resource demand with Stochastic Programming. To that end, we can greedily distribute the geo-distributed interactive service to low price data centers and iteratively adjust the resource provision plan. Our contributions in this paper include:

1) We construct several optimization functions to work out the distribution and the resource provision planning strategy. The resource provision cost comprises the reservation resource and the on-demand resource for multi-CSPs. Importantly, we achieve complex distribution and resource provision planning by modeling tail latency constraint of the geo-distributed interactive service with high-percentile Service Level Agreement (SLA).

2) We novelly propose an algorithm, namely Two-dimension Resource Provision (2DRP). With resource price uncertainty, 2DRP distributes the Internet requests to the lowest resource price data center group with Stochastic Programming technique in every distribution decision period coarse-grained. With resource demand uncertainty, 2DRP estimates the resource demand through historical execution of the geo-distributed interactive service's Internet requests; then it uses decomposition technique to develop fixed optimized resource provision plan in every provision planning period in target data center group fine-grained.

3) Under a variety of settings, we conduct extensive simulation-based experiments using multi-CSPs, real Clouds and Google data center workload traces. The result turns out that the resource provision cost of geo-distributed interactive services is reduced up to 24% compared with the On-demand based infrastructure, and up to 10% compared with both the Reservation based infrastructure and the OCRP infrastructure. Besides, with 2DRP, the geo-distributed interactive services can meet the high-percentile SLA tail latency constraint.

## II. BACKGROUND AND MOTIVATION

In this section, we provide necessary background for the motivation and challenges behind minimizing geo-distribution interactive service cost with multi-CSPs.

### A. GEO-DISTRIBUTED INTERACTIVE SERVICE

Geo-distributed interactive service, such as Mapreduce Interactive Analysis (MIA) and web search, is a type of Internet Cloud Service that provides end-users business decisions or plentiful contents. In MIA, analyzing the data gathered across data center regions is a critical workload. Geo-distributed interactive service for the real-time data analytics can provide up-to-the-minute information about an enterprise's customers and present it so that better and quicker business decisions can be made -- perhaps even within the time span of a customer interaction. Also, in a data warehouse context, geo-distributed interactive service supports unpredictable ad hoc queries against large data sets [13]. Examples include querying user logs to make advertisement decisions. Since the data analysts and operators use results of these analytics queries for real-time decision, these services are latency-sensitive. To ensure high performance for geo-distributed interactive service, Internet requests of these interactive services need to be distributed to data center groups and utilize enough data center resources to accomplish the running of data center workloads in collaborative.

### B. CLOUD PRICING OF MULTI-CSPS

Google Compute Engine (GCE), Amazon AWS and Microsoft Azure globally deployed a certain number of data centers, which are organized as several data center regions for business purpose. These CSPs provide diverse resource price policies for different Internet Cloud Services [14]–[16]. For data centers deployed in different locations, GCE and Azure provide different compute capacity VM instances with different unit resource prices (e.g., Google North America: 2 VCPU, 4G RAM, 250G SSD, $1.424/hour; Azure North America: 2 VCPU, 4G RAM, 250G SSD, $1.324/hour) [14], [16]. With the On-demand VM instances, the cloud consumer pays the compute capacity by the hour with no long-term commitments or upfront payments. The cloud consumer can increase or decrease the compute capacity depending on the demands and only pays the specified hourly rate for the VM instances that used [15]. Besides, AWS not only provides On-demand VM instances similar to GCE and Azure but also allows the cloud consumer to use the convertible reserved VM instances. A cloud consumer can set typical configuration VM instances for one year or three years and make the upfront payment, and then charged a discounted hourly rate for the duration of the reserved VM instance term (approximately 50%) [15].

### C. DATA CONSISTENCY

Naturally, applications running in the data center can produce the corresponding raw data (such as user access records or shipping information). With the Facebook data center as an example of this tremendous data growth, one has to just look at the fact that while today they load between 10-15TB of compressed data every day, just six months back this number was in the 5-6TB range [17]. Significantly, the geo-distributed interactive service requires the corresponding raw data from different data centers. Needless to say, such a rapid growth places very strong scalability requirements on the data processing infrastructure. For instance, MIA needs dozens of VM instances to finish the execution of the data center workload [18]. However, the resource prices are different from data centers, even though these data centers belong to the same CSP. The good news is the raw data is replicated among the data centers in the same region with the aim of providing high availability. Internet requests for the geo-distributed interactive service can be distributed to the data centers with low resource price to reduce the resource provision cost.

Our work focuses on minimizing the resource provision cost of geo-distributed interactive service. A potential approach would be to distribute the interactive service to the lowest price data center group and make the maximum resource provision for the changing demand. However, the resource price and demand can be highly heterogeneous, and the distributions could be dramatically inefficient. If the cloud consumer sets the maximum resource in every moment, the resources are always overprovisioned. In so far, it is possible to achieve good performance and truthful resource provision with the consideration of price uncertainty and demand uncertainty.

## III. AN OVERVIEW OF THE 2DRP FRAMEWORK

### A. RESOURCE PROVISION STRATEGIES

2DRP considers both of the Reservation resource and the On-demand resource. Hence, there are three resource provision situations: reservation only, on-demand only and the hybrid. These situations present in different times with different events as stipulated in the agreements: If data center provides the reservation resource and the cloud consumer prefers the reservation resource for long term service deployment, the cloud consumer takes action of making reservation contract and paying up-front at first. After the service online, the resource price and the resource demand are profited, 2DRP then allocates resources with the resource provision plan in advance and the reserved resources can be utilized. Without knowing the actual demand. As a result, the resources could be observed either overprovisioned or underprovisioned. If the actual demand exceeds the maximum reserved resource limitation of the reservation contracts, QoS cannot be guaranteed. If data center provides the on-demand resource, 2DRP can pay for additional resources to achieve the QoS requirement, and then the hybrid stage (reservation and on-demand) starts. If data center only provides the on-demand resource or the cloud consumer prefers the on-demand resource for short term application deployment, then 2DRP dynamically provides the on-demand resource.

### B. WORKLOAD DISTRIBUTION

2DRP is easy to understand. The Internet requests from the traffic source represent a list of executable data center workloads. At first, 2DRP makes optimized workload distribution decisions by choosing the lowest price data center, and with these decisions, the Internet requests will be distributed to the target data center group. Before workloads executed in this data center group, 2DRP develops resource provision plan with estimated resource demand. Then resources are allocated to workloads according to the resource provision plan, and the resources are provided as the data center VM instances. Figure 1 shows a simple case, Internet requests distribute to low price data center **B** rather than data center **A**. Workloads will run in the target data center group with proper resources. Especially, after finishing the Internet request, traces of latency and resource utilization send back to the workload dispatcher for the next resource provision planning.

We list the three main components of 2DRP as below:

**Traffic source**: Let $S$ denote set of traffic sources for $m$ geo-distributed interactive services. Traffic source sends Internet requests to access geo-distributed interactive service, and $I$ denotes set of data center workloads for each service.

**Data center**: Let $J$ denote set of data centers in every region. Each data center provides a pool of resources to the Cloud consumers. Let $\Phi$ denote all the resource types. Resource types can be computing power, memory, disk, and WAN network for data transmission [19]–[22].

**Time slot**: Workload distribution decisions are made at the beginning of every decision period $T$, namely WDD (Workload Distribution Decision) time slot. Due to the time-varying feature of the Internet request rates, 2DRP needs to periodically change the resource provision plan after every billing period $t$, namely RPP (Resource Provision Planning) time slot. Importantly, $t \in t_1, t_2, \ldots, t_n; \forall t \in T$.

## IV. PROBLEM FORMULATION

### A. DATA CENTER GROUP

Assume $R$ regions provide data center computing resources for the running of geo-distributed interactive services. In every region, there are $P$ cloud service providers and provider $i$ has deployed $DC_i$ $(i \in P)$ data centers for Internet Cloud Services. Let $J_r$ denote the number of data centers in region $r$ $(r \in R)$. Then, we can calculate all the available data

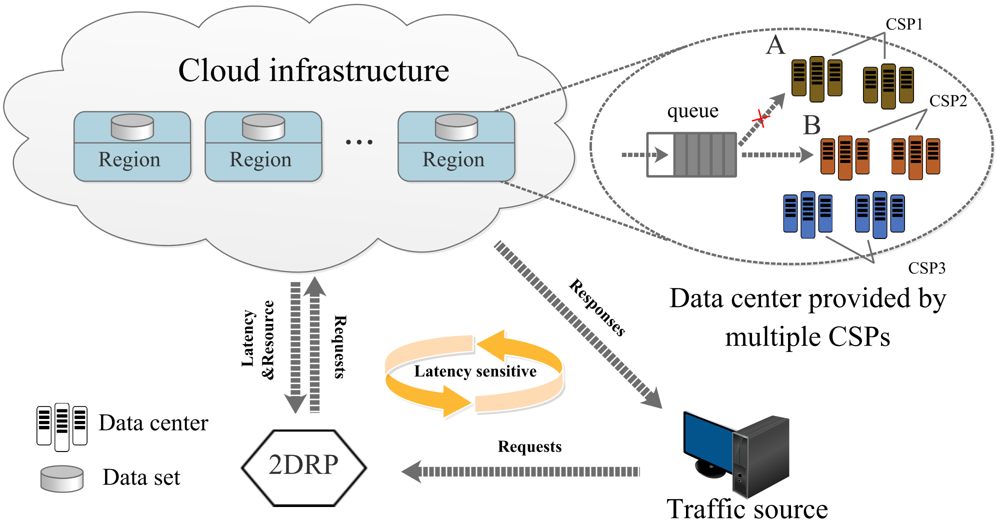


**FIGURE 1.** **Workload distribution of 2DRP: Internet requests are distributed to low resource price data centers.**

centers in region $r$ with equation

$$J_r = \sum\nolimits_{i\in P} DC_i. \tag{1}$$

We concern about $m$ geo-distributed interactive services, and each one contains more than $|R|$ data center workloads. Traffic source of geo-distributed interactive service sends Internet requests to $n$ ($n \geq |R|$) data centers, which we call the target data center group. These data centers are gathered from different regions, which means each region needs to provide at least one data center to accomplish the running of geo-distributed interactive service. So, let $G$ denote all the possible data center groups provided by $R$ data center regions. With the Combinatorial Mathematical Theory, we can calculate all data center groups as

$$G = \prod\nolimits_{r\in R} J_r = J_1 \times J_2 \times \ldots \times J_{|R|}. \tag{2}$$

In this paper, the theory of the data center group is feasible, since the raw data are replicated among data centers in the same region. Ordinarily, the replication of raw data among data centers aims at providing high reliability and availability. If data centers do not replicate the raw data, we consider each un-replicated data center as an independent region.

### B. COST MODEL

When cloud consumer made reservation contract $k$ for workload $i$, the up-front of making reservation contract for typical VM instance is charged by a fixed one-time fee. Let $b_{i\phi}$ denote the amount of resources of resource type $\phi$ ($\phi \in \Phi$) required by the typical VM instance. Let $c^U_{j\phi k}$ denote the unit price (e.g., up-front for the reservation) of data center $j$ for resource type $\phi$ subscribed to reservation contract $k$. The up-front, denote as $p^U_{ijk}$, is the prepaid expenses for provisioning every resource type

$$p^U_{ijk} = \sum\nolimits_{\phi\in\Phi} b_{i\phi} c^U_{j\phi k}. \tag{3}$$

After the Internet Cloud Service online, the cloud consumer is charged with discounted hourly price for the running of the reserved VM instance, and we call it the expanding cost. Let $c^R_{j\phi kt}$ denote the hourly price of data center $j$ for resource type $\phi$ with reservation contract $k$ when running the reserved VM instance during $t$. Let $p^R_{ijkt}$, defined as the similar way of (3), denote the expanding cost for running the reserved VM instance.

Let $c^O_{j\phi t}$ denote the hourly price of the on-demand resource of data center $j$ for resource type $\phi$ during $t$. Also, the on-demand price can be changed by CPS; it is uncertain to the consumer when the resource is on-demand. Let $p^O_{ijt}$, defined as the similar way of (3), denote the on-demand cost of running the on-demand VM instance. Especially, given traffic source $i$, data center $j$, the expanding cost of any reserved resource is cheaper than the on-demand resource.

Different from the data center computing resource, the WAN network resource is charged according to the amount of network package transferred per month. When the data transmission exceeds the maximum amount of traffic that introduced in the network price policy, the WAN network resource will be charged at a different unit price. Let $K_n$ denote the relationship between the amount of data transferred and the unit price. For example, data transfer out from Amazon EC2 to Internet: $K_n$ = {"First 1 GB":"\$0.000/GB","Up to 10 TB":"\$0.090/GB","Next 40 TB":"\$0.085/GB","Next 100 TB":"\$0.070/GB","Next 350 TB":"\$0.050/GB", $\cdots$}. Hence, let $p^N_{ijk}$ denote the unit price of WAN network traffic of data center $j$ for workload $i$ in $k$ ($k \in K_n$) situation.

If we distribute geo-distributed interactive service to data center group, we must allocate a sufficient number of typical VM instances for the running of workloads (e.g., 16core CPU, 64G memory and 500G SSD per VM instance × 10). We have provided a novel method to calculate the unit price of adaptable VM instance and WAN network resource. Then, we can calculate the resource provision cost according to the occupancy of the total amount of the typical VM instances and the WAN network traffic. We use Formulation (4) to calculate the resource provision cost of geo-distributed interactive services.

$$\sum_{s}^{S}\sum_{i}^{I}\sum_{j}^{J_g} X_{ij}(\sum_{k}^{K} x_{ijk}^{U} p_{ijk}^{U} + \sum_{t}^{T_n}\sum_{k}^{K_n} x_{ijkt}^{N} p_{ijkt}^{N} + C^{Y}). \quad (4)$$

Note that $s$ ($s \in S$) denotes traffic source, $i$ ($i \in I$) denotes workload for traffic source $s$. Simply each workload needs to distribute to one data center. $j$ ($j \in J_g$) denotes the data center belongs to group $g$ ($g \in G$). $x_{ijk}^{U}$ denotes the number of reserved VM instances for workload $i$ in data center $j$ with reservation contract $k$. $x_{ijk}^{N}$ denotes the total amount of WAN network package transferred per month related to corresponding network utilization situation $k$ ($k \in K_n$) in each network resource billing period $t$. Note that $t \in T_n$, while $T_n$ can be 12 months in one year. $X_{ij}$ denotes a binary variable: it equals to 1 if workload $i$ distributed to data center $j$, and 0 otherwise.

The resource provision cost includes three parts: up-front of reserved $x_{ijk}^{U}$ VM instances; the expenses of WAN network traffic; the expenses of running reserved VM instance and on-demand VM instance in every billing period $t$. The third part is denoted as $C^{Y}$, which can be changed dynamically. If the running of reserved VM instances $x_{ijkt}^{R}$ cannot meet the tail latency requirements ($x_{ijk}^{U} = x_{ijkt}^{R} < \mathit{actual\ demand}$), we need extra $x_{ijt}^{O}$ VM instances to reach it. The dynamic resource provision cost is shown as

$$C^{Y} = \sum_{t}^{T}(\sum_{k}^{K} x_{ijkt}^{R} p_{ijkt}^{R} + x_{ijt}^{O} p_{ijt}^{O}). \quad (5)$$

### C. LATENCY MODEL

Workloads running in the data center has processing latency and WAN network package transmission latency [23]. This work [24] provides an extensible network-aware QoS prediction model to estimate the latency of Internet Cloud Service. Since the processing latency is related to the size of raw data and the amount of allocated data center resources, the mapping function and latency CDFs for historical Internet requests can be applied to predict the processing latency. Since the processing latency is usually in little change for the typical size of raw data and a certain amount of computing resources, especially for MIA services, the processing latency can be predicted by mapping functions with the workload traces. However, the interference of workload execution has a great impact on the processing latency. It can be eliminated by the latency CDFs. Let $l_{ij}^{C}(t)$ denote the data processing latency. The data transmission latency is positively related to the size of intermediate data being transmitted. Hence, the WAN network latency can be obtained as the similar way of the data processing latency. Let $l_{ij}^{N}(t)$ denote the WAN network transmission latency.

For different types of data center workload, computing latency and network package transmission latency vary widely. For some Internet Cloud Services, computing latency is the main contribution to the service latency when for other Internet Cloud Services, network transmission latency is the main contribution to service latency. So, we use (6) to calculate the fixed latency $l_{ij}(t)$.

$$l_{ij}(t) = l_{ij}^{C}(t) + l_{ij}^{N}(t). \quad (6)$$

Each Internet request has a corresponding latency constraint. We independently consider every Internet request and believe if all Internet requests meet the latency constraint, then the geo-distributed interactive service can satisfy the QoS requirements. Let $L_{ij}^{C}(t)$ denote the processing latency threshold. Let $L_{ij}^{N}(t)$ denote WAN network latency threshold. Let $L(t)$ denote the fixed latency threshold for Internet requests of $m$ geo-distributed interactive services sent to $n$ ($n = |J_g|$) data centers.

$$\begin{aligned} L(t) &= L_{ij}^{C}(t) + L_{ij}^{N}(t) \\ &= \begin{bmatrix} L_{11}^{C} + L_{11}^{N} & L_{12}^{C} + L_{12}^{N} & \dots & L_{1n}^{C} + L_{1n}^{N} \\ L_{21}^{C} + L_{21}^{N} & L_{22}^{C} + L_{22}^{N} & \dots & L_{2n}^{C} + L_{2n}^{N} \\ \vdots & \vdots & \ddots & \vdots \\ L_{m1}^{C} + L_{m1}^{N} & L_{m2}^{C} + L_{m2}^{N} & \dots & L_{mn}^{C} + L_{mn}^{N} \end{bmatrix}. \end{aligned} \quad (7)$$

Formulation (6) and (7) used for latency estimation are based on high-level workload execution which does not consider the interaction of workloads within the same data center. We will struggle this limitation in Section V.

### D. FORMULATION

Combining the cost model and latency model, we can formulate the minimizing resource provision cost problem as the following optimization function.

$$\begin{aligned} \mathit{Minimize} \ & \sum_{s}^{S}\sum_{i}^{I}\sum_{j}^{J_g} X_{ij}[\sum_{k}^{K} x_{ijk}^{U} p_{ijk}^{U} + \sum_{t}^{T_n}\sum_{k}^{K_n} x_{ijkt}^{N} p_{ijkt}^{N} \\ & + \sum_{t}^{T}(\sum_{k}^{K} x_{ijkt}^{R} p_{ijkt}^{R} + x_{ijt}^{O} p_{ijt}^{O})]. \end{aligned} \quad (8)$$

$$\text{Subject to} \ \sum_{j}^{J_g} X_{ij} = 1 \quad \forall s \in S, \ \forall i \in I, \quad (8a)$$

$$l_{ij}(t) \geq L_{ij}(t) \quad \forall L_{ij}(t) \in L(t), \ \forall i \in I, \ \forall j \in J_g, \quad \forall t \in T, \quad (8b)$$

$$\sum_{s}^{S}\sum_{i}^{I} X_{ij} b_{i\phi}(\sum_{k}^{K} x_{ijkt}^{R} + x_{ijt}^{O}) \leq a_{j\phi t} \quad \forall j \in J_g, \ \forall \phi \in \Phi, \quad \forall t \in T, \quad (8c)$$

$$x_{ijkt}^{R} \leq x_{ijk}^{U} \quad \forall i \in I, \ \forall j \in J_g, \ \forall k \in K, \ \forall t \in T, \quad (8d)$$

$$x_{ijk}^{U} \in N^{*} \quad \forall i \in I, \ \forall j \in J_g, \ \forall k \in K, \quad (8e)$$

$$x^R_{ijkt} \in N^* \quad \forall i \in I,\ \forall j \in J_g,\ \forall k \in K,\ \forall t \in T, \tag{8f}$$

$$x^O_{ijt} \in N^* \quad \forall i \in I,\ \forall j \in J_g,\ \forall t \in T, \tag{8g}$$

$$x^N_{ijkt} \in N^* \quad \forall i \in I,\ \forall j \in J_g,\ \forall k \in K_n,\ \forall t \in T_n. \tag{8h}$$

Constraint (8a) indicates that each workload must be distributed to one data center. $J_g$ denotes a set of data centers of the group $g$. Constraint (8b) indicates that latency of Internet requests must not exceed the latency threshold. Constraint (8c) indicates that the allocated data center resource must not exceed the maximum resource limitation for all resource types. $a_{j\phi t}$ denotes the maximum resource of data center $j$ for resource type $\phi$ during RPP time slot $t$. Constraint (8d) indicates that the running reserved VM instances must not exceed the limitation of maximum reserved VM instances. Constraint (8e), (8f), (8g), (8h) together indicate that variables of the number of VM instance and the usage of WAN network traffic take the values from a set of non-negative integer numbers.

## V. DESIGN

We have provided an optimization function for workload distribution and resource provision planning. However, with Formulations (6) and (7), the latency estimation are based on high-level workload execution, and the latency prediction error is unbounded. Accurate estimation of the latency for each Internet request is in high cost which results in the latency model are infeasible. So we change to concern about the probability of Internet requests not exceed the latency threshold and design an SLA fixed tail latency model. We only need to measure whether the Internet requests meet the latency constraints when the geo-distributed interactive service is finished, rather than estimate the latency for every future Internet request.

### A. HIGH-PERCENTILE SLA TAIL LATENCY

There is no simple closed-form method to compute the tail latency of geo-distributed interactive services. We have studied some global load balancing approaches [25], [26], which can estimate the tail latency of Internet request separately. However, quantifying accurate tail latency of each interactive service is difficult, especially the dynamic configuration of data center resources and time-varying feature of the request rates greatly affect the tail latency. Also, the shared data center resource causes interference [27].

This work [28] presents a data-driven approach to statically estimate the latency for each network route from traffic sources to data centers and redistributes the workloads to lower electricity price data center. In this case, we use *high-percentile tail latency* for Internet requests as SLA [29]. *For example, if we set x as the high-percentile SLA tail latency constraint, then the probability of Internet request not exceeding the latency threshold must not less than x%. Otherwise, the tail latency of these Internet requests cannot be guaranteed.* The high-percentile SLA tail latency provides a new way to let the geo-distributed interactive services satisfy the QoS requirement.

### B. TAIL LATENCY PROFILING

We focus on making probability tail latency estimation that can adapt to the optimization function shown as (8). In our design, we at first use the probabilistic method to estimate the probability distribution of processing latency and WAN network transmission latency. We then use the convolution function to integrate them. The estimation of probability tail latency is made by using the Internet request's traces because traces of Internet requests will send back to the workload dispatcher after finishing the execution of Internet request. These traces include the information of latency and resource utilization situation for each Internet request.

Consider one traffic source $s$ sends $\Omega$ Internet requests to data center $j$ to accomplish the execution of workload $i$ during time slot $t$. We define the method of estimating the high-percentile SLA tail latency as following definitions.

*Definition 1 (Probability Latency):* The probability of Internet request does not exceed the workload processing latency threshold and WAN network transmission latency threshold are denoted as $f^C_{ij}(t)$ and $f^N_{ij}(t)$, respectively. Note that operation $[l^C_{ij}(t) \geq L^C_{ij}(t)]$ represents a statistical method, and when $l^C_{ij}(t) \geq L^C_{ij}(t)$, the result is 1, otherwise 0. The same to the operation $[l^N_{ij}(t) \geq L^N_{ij}(t)]$.

$$f^C_{ij}(t) = \frac{\sum_{w\in\Omega}[l^C_{ij}(t) \geq L^C_{ij}(t)]_w}{|\Omega|}, \tag{9}$$

$$f^N_{ij}(t) = \frac{\sum_{w\in\Omega}[l^N_{ij}(t) \geq L^N_{ij}(t)]_w}{|\Omega|}. \tag{10}$$

*Remark 1:* According to the law of large numbers (LLN), the average of the latency obtained from a large number of latency traces should be close to the expected value. The latency estimation will tend to become more accurate as more traces are considered in the estimation.

*Definition 2 (Fixed Probability Latency):* The profiled latency of each Internet request for workload $i$ in data center $j$ is denoted as

$$f_{ij}(t) = f^C_{ij}(t) * f^N_{ij}(t), \tag{11}$$

where the operator "$*$" represents a convolution method.

*Remark 2:* Contributions of latency for Internet request include two parts: a) The data center processing latency. b) The WAN network package transmission latency. Convolution function mathematically produces a third function, that is typically viewed as the superposition of the computing latency and network latency.

*Definition 3 (Probability Tail Latency):* Internet requests from traffic source are sent simultaneously to data centers within its group, the probability tail latency for Internet requests from traffic source should be averaged across the data center group and be expressed as

$$F_{sg}(t) = \frac{\sum_{[i,j]}^{[I,J_g]} u_{ij}(t) f_{ij}(t)}{\sum_{[i,j]}^{[I,J_g]} u_{ij}(t)}, \tag{12}$$

where $F_{sg}(t)$ denotes the probability tail latency for Internet requests from traffic source $s$ sent to data center group $g$, and we use $F_{sg}(t)$ to emphasize that the probability tail latency estimation is a function of our workload distribution decision and resource provision planning.

With Formulation (9)-(12), we can get the probability tail latency of Internet requests sent to each data center group. Then, we change the latency constraint to the high-percentile SLA tail latency constraint, which is denoted as $\text{Pr}^{SLA}$.

$$F_{sg}(t) \geq \text{Pr}^{SLA} \tag{13}$$

Besides, previously established knowledge [30] provides high-percentile SLA tail latency modeling in multi-tenant environments. P95 ($Pr^{SLA} = 0.95$) is good enough in the situation of reality, since 5% probability of Internet request exceeds the latency threshold have little effects on the user experience and can be ignored by the end users. 2DRP can set different values as the high-percentile SLA tail latency constraint according to the needs of different QoS requirements. Also, we only focus on the high-level execution of geo-distributed interactive service, because we believe that the workloads are successfully executed when all Internet requests are expected to complete. We can ignore the details of their connections and only focus on the Internet requests from traffic sources sent to data center groups since existing Cloud computing framework already provides features so that we do not need to highly concern about these details.

### C. PRICE UNCERTAINTY AND DEMAND UNCERTAINTY

The unit price of data center resource is dynamic, and resource demand of geo-distributed interactive service will dynamically change following Internet request rates. To adapt to this situation, we utilize Stochastic Programming to handle the uncertainty situation for solving the problem of resource provision cost minimization. Stochastic Programming takes a set of uncertainty parameters (called scenarios), described by a probability distribution into account. So, we can use this feature to solve the problem. Let $\Lambda_t$ denote the set of price and demand scenarios in every RPP time slot $t$. For all scenarios during WDD time slot $T$, set $\Lambda$ is defined as the Cartesian product of $\Lambda_1, \Lambda_2, \ldots, \Lambda_{|T|}$.

$$\Lambda = \prod_{t \in T} \Lambda_t = \Lambda_1 \times \Lambda_2 \times \ldots \times \Lambda_{|T|}. \tag{14}$$

We know that the probability distribution of $\Lambda$ has finite support (e.g., set $\Lambda$ has a finite number of scenarios with respective probabilities $P_\lambda \in [0, 1]$). $\lambda$ is a composite variable defined as $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_{|T|}) \in \Lambda$. In this paper, uncertain price and uncertain demand are considered as scenarios in $\Lambda$ whose probability distribution is assumed to be available.

Given a probability distribution of all scenarios, we can reformulate the expanding reservation cost and on-demand cost, the original version is defined as (5), as Formulation (15).

$$C^Y = \sum_t^T \sum_\lambda^\Lambda P_\lambda (\sum_k^K x_{ijkt\lambda}^R p_{ijkt\lambda}^R + x_{ijt\lambda}^O p_{ijt\lambda}^O). \tag{15}$$

With the high-percentile SLA tail latency model and Stochastic Programming technique, the optimization function shown in (8) can be reformulated as (16). $P_\lambda$ denotes the probability of scenario $\lambda$. To solve this optimization function, probability distributions of both price and demand must be available.

$$\begin{aligned} \text{Minimize } & \sum_s^S \sum_i^I \sum_j^{J_g} X_{ij} [\sum_k^K x_{ijk}^U p_{ijk}^U + \sum_t^{T_n} \sum_k^{K_n} x_{ijkt}^N p_{ijkt}^N \\ & + \sum_t^T \sum_\lambda^\Lambda P_\lambda (\sum_k^K x_{ijkt\lambda}^R p_{ijkt\lambda}^R + x_{ijt\lambda}^O p_{ijt\lambda}^O)]. \end{aligned} \tag{16}$$

$$\text{Subject to } \sum_j^{J_g} X_{ij} = 1 \quad \forall s \in S,\ \forall i \in I,\ \forall \lambda \in \Lambda, \tag{16a}$$

$$F_{sg}(t) \geq \text{Pr}^{SLA} \quad \forall s \in S,\ \forall i \in I,\ \forall j \in J_g,\ \forall g \in G, \quad \forall t \in T, \tag{16b}$$

$$\sum_s^S \sum_i^I X_{ij} b_{i\phi} (\sum_k^K x_{ijkt\lambda}^R + x_{ijt\lambda}^O) \leq a_{j\phi t} \quad \forall j \in J_g, \quad \forall \phi \in \Phi,\ \forall t \in T,\ \forall \lambda \in \Lambda, \tag{16c}$$

$$x_{ijkt\lambda}^R \leq x_{ijk}^U \quad \forall i \in I,\ \forall j \in J_g,\ \forall k \in K,\ \forall t \in T, \quad \forall \lambda \in \Lambda, \tag{16d}$$

$$x_{ijkt\lambda}^R \in N^* \quad \forall i \in I,\ \forall j \in J_g,\ \forall k \in K,\ \forall t \in T, \quad \forall \lambda \in \Lambda, \tag{16e}$$

$$x_{ijt\lambda}^O \in N^* \quad \forall i \in I,\ \forall j \in J_g,\ \forall t \in T,\ \forall \lambda \in \Lambda. \tag{16f}$$

Note that $x_{ijkt\lambda}^R$ denotes the reserved VM instances, and $x_{ijt\lambda}^O$ denotes the on-demand VM instances under the uncertainty parameter of price and demand with scenario $\lambda$. Constraint (16b) indicates that the probability tail latency of each geo-distributed interactive service distributed to target data center group $g$ must meet the high-percentile SLA tail latency constraint. Other constraints are defined as the similar way of the constraints for optimization function (8) but with price and demand uncertainty scenario characteristics, which means that these constraints should be met in any scenario.

### D. TWO-DIMENSIONAL RESOURCE PROVISION DECISIONS

When the number of data centers and the request rate for geo-distributed interactive service keep in low, the above solution works well. With the increasing number of data centers in every region, the optional data center groups become huge, and tail latency estimations for all the scenarios become the bottleneck. We split the original optimization function into two-stage optimization functions to handle the price uncertainty and demand uncertainty. The workload distribution decision maker takes some action for making workload distribution and developing primary resource provision plan at first, after which a random event occurs affecting the outcome of the primary resource provision plan. An improved recourse provision plan can then be developed in next RPP time slot that compensates for any bad effects that might have been experienced as a result of the workload distribution and the primary resource provision plan.

### 1) DECISION MATRIX

We know that the resource provision of every data center does not affect each other. Hence, we can ignore the data center capacity and calculate resource provision cost by developing the primary optimized resource provision plan parallelly in regions. Compared with developing the resource provision plan for all data center group and all workloads with consideration of data center capacity, this method can save 99% computing time. After that, we select data centers with the lowest resource provision cost in every region and reconstruct as the data center group with the consideration of data center capacity. In the original optimization functions (8) and (16), we concern about the data center group $J_g$. We now concern about the set of data centers $J$ in every region. At the beginning of WDD time slot $T$, we use the Formulation (17) to obtain the workload distribution that covers all possible data center in every region and calculate its corresponding resource provision cost. In particular, the cost will be used for making workload distribution decision. For example, there are $\beta$ candidate data centers for workload $i$ in the region $r$; then we can get $\beta$ resource provision cost as weight values for workload distribution decision. We distributed workload $i$ to the data center which holds the lowest value.

$$\text{Minimize } C_{ij}(s) = \sum_{k}^{K} x_{ijk}^{U} p_{ijk}^{U} + \sum_{t}^{T_n} \sum_{k}^{K_n} x_{ijkt}^{N} p_{ijkt}^{N} + \sum_{t}^{T} \sum_{\lambda}^{\Lambda} P_\lambda \Big(\sum_{k}^{K} x_{ijkt\lambda}^{R} p_{ijkt\lambda}^{R} + x_{ijt\lambda}^{O} p_{ijt\lambda}^{O}\Big). \tag{17}$$

$$\text{Subject to } F_{sg}(t) \geq \Pr^{SLA} \quad \forall s \in S,\ \forall i \in I,\ \forall j \in J,\ \forall g \in G, \quad \forall t \in T, \tag{17a}$$

$$x_{ijkt\lambda}^{R} \leq x_{ijk}^{U} \quad \forall i \in I,\ \forall j \in J,\ \forall k \in K,\ \forall t \in T, \quad \forall \lambda \in \Lambda. \tag{17b}$$

Note that $C_{ij}(s)$ denotes the optimized resource provision cost for workload $i$ distributed to data center $j$. $J$ denotes set of data centers belong to the same region which is different from $J_g$ when $J_g$ is defined as Formulation (2).

In general, we separately calculate the resource provision cost of every geo-distributed interactive service workload in possible data centers. $C_{ij}(s)$ denotes the resource provision cost for workload $i$, which belongs to geo-distributed interactive service $s$, distributed to data center $j$. For all geo-distributed interactive services, we use $C$ as weight value matrix for the workload distribution decision. With the weight value matrix, we greedily choose the lowest cost data center.

### 2) WORKLOAD DISTRIBUTION DECISION AND RESOURCE PROVISION PLANNING

To solve the optimization function (17), we can obtain the primary resource provision plan and its corresponding resource provision cost in a set of candidate data centers for each interactive service workload. Then we use decision variable matrix $X$ to choose data centers from the candidate data centers to construct the target data center group. 2DRP is divided into two stages for solving this complex problem. The stage-one of 2DRP for workload distribution decision is by resolving the master problem shown in (18). In stage-two, 2DRP resolves set of scenario subproblems, shown as (19a), to develop the resource provision plan for every workload in target data center group.

**Master problem**

$$\text{Minimize } \sum_{s}^{S} \sum_{i}^{I} \sum_{j}^{J} X_{ij} C_{ij}(s). \tag{18}$$

$$\text{Subject to } \sum_{j}^{J} X_{ij} = 1 \quad \forall s \in S,\ \forall i \in I,\ \forall \lambda \in \Lambda, \tag{18a}$$

$$\sum_{s}^{S} \sum_{i}^{I} \sum_{j}^{J} X_{ij} b_{i\phi} \Big(\sum_{k}^{K} x_{ijkt\lambda}^{R} + x_{ijt\lambda}^{O}\Big) \leq a_{j\phi t} \quad \forall j \in J, \quad \forall \phi \in \Phi,\ \forall t \in T,\ \forall \lambda \in \Lambda. \tag{18b}$$

**Scenario subproblems**

$$\left\{ \begin{matrix} \sum_{\lambda}^{\Lambda_1} P_\lambda C_{i'j't_1}^{1\lambda} & \sum_{\lambda}^{\Lambda_2} P_\lambda C_{i'j't_2}^{1\lambda} & \cdots & \sum_{\lambda}^{\Lambda_{|T|}} P_\lambda C_{i'j'|T|}^{1\lambda} \\ \sum_{\lambda}^{\Lambda_1} P_\lambda C_{i'j't_1}^{2\lambda} & \sum_{\lambda}^{\Lambda_2} P_\lambda C_{i'j't_2}^{2\lambda} & \cdots & \sum_{\lambda}^{\Lambda_{|T|}} P_\lambda C_{i'j'|T|}^{2\lambda} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{\lambda}^{\Lambda_1} P_\lambda C_{i'j't_1}^{w\lambda} & \sum_{\lambda}^{\Lambda_2} P_\lambda C_{i'j't_2}^{w\lambda} & \cdots & \sum_{\lambda}^{\Lambda_{|T|}} P_\lambda C_{i'j'|T|}^{w\lambda} \end{matrix} \right. , \tag{19}$$

$$C_{i'j't}^{w\lambda} = X_{i'j'} \Big(\sum_{k}^{K} x_{i'j'kt\lambda}^{R} p_{i'j'kt\lambda}^{R} + x_{i'j't\lambda}^{O} p_{i'j't\lambda}^{O}\Big), \tag{19a}$$

$$X_{i'j'} \in X_{ij}, \quad X_{i'j'} = 1. \tag{19b}$$

In the master problem, constraints (18a) and (18b) indicate that workloads should be assigned to lowest cost data center but only if it needs to meet the capacity of the data center with the primary resource provision plan of all the workloads. In the scenario subproblems, the column elements in the matrix represent the optimization functions of all the workloads ($w = n \times m$) in specified RPP time slot $t$. The row elements represent the optimization function of the specified workload during WDD time slot $T$. (19aa) and (19ab)

**Algorithm 1** Pseudo-Code for 2DRP, Stage One

```
Input: Traffic source:$S$; Region:$R$; SLA Tail latency
       constraint:$\Pr^{SLA}$; WDD time slot $T$.
Output: Distribution matrix:$X$.
Step 1: Generate $\Lambda$ ($\Lambda_t \in \Lambda$) independent samples each
of size N, i.e., $(\Lambda_t^1, \dots, \Lambda_t^N)$ for $t \in 1, 2, \dots, |T|$.
Step 2: Formulate the high-percentile SLA tail latency
model and the master problem with $\Lambda$ samples.
Step 3: Solve the master problem with MathProg and
GLPK. Let $C$ denote the incumbent solution of the
master problem.
  $d \leftarrow size(R)$; Note that If d = 3, there are three data
center regions, e.g. US, EU and Asia.
  $C[m][d][n] \leftarrow 0$;
  For $(s, i, j)$ to (#, #, #) for all collections $(S, I, J)$ do
      $C_{sij}^U \leftarrow \sum_{k \in K} p_{ijk}^U x_{ijk}^U$;
      $C_{sij}^N \leftarrow \sum_{t \in T_n} \sum_{k \in K_n} x_{ijkt}^N p_{ijkt}^N$;
      $C_{sij}^Y \leftarrow \sum_{t \in T} \sum_{\lambda \in \Lambda} P_\lambda (\sum_{k \in K} x_{ijkt\lambda}^R p_{ijkt\lambda}^R + x_{ijt\lambda}^O p_{ijt\lambda}^O)$;
      $C[s][i][j] \leftarrow Min \ \{C_{sij}^U + C_{sij}^N + C_{sij}^Y\}$;
  end
Step 4: Initialize the workload distribution decision
matrix with the corresponding optimal objective value
and LP.
  $X[m][d][n] \leftarrow 0$;
  $Min \ \ X \cdot C$;
```

together indicate that 2DRP only develops the resource provision plan for the target data center group and ignores the other groups. These subproblems are independent of each other so can be solved in parallel.

Benders decomposition algorithm can be applied to solve the master problem and scenario subproblems. The formal description of stage-one is presented in Algorithm 1. The optimized resource provision cost for each data center will be calculated. Variable $C[m][d][n]$ denotes the optimized resource provision cost for the workload distributed to candidate data centers. Note that $m$ represents the number of geo-distributed interactive services, $d$ denotes the number of regions, $n$ denotes the number of data centers belongs to the same region. Then, the workload distribution decision variable $X[m][d][n]$ can be obtained by selecting the lowest cost data center with weight matrix $C[m][d][n]$. We can decide whether to develop a resource reservation plan in data center $j$ for workload $i$, which is reflected by the value of $x_{ijk}^U$. The formal description of stage-two is presented in Algorithm 2. With estimated probability tail latency, we only calculate the dynamic cost $C_{sij}^Y$ for the selected data center group and update the resource provision plan. The plan can be retrieved by the variables $x_{ijkt}^R$ and $x_{ijt}^O$.

In particular, we use GLPK to solve the master problem (18), then we can find the optimized workload distribution in the beginning of WDD time slot $T$. We distribute

**Algorithm 2** Pseudo-Code for 2DRP, Stage Two

```
Input: Distribution matrix: $X$; RPP time slot: $t$.
Output: Resource provision plan: $x$.
Step 1: Generate $\Lambda_t$ independent samples with the
corresponding workload traces with the same RPP time
slot $t$.
Step 2: Bind the workload and target data center with
workload distribution decision matrix $X$.
$(i', j') \leftarrow \{(i, j)|_{X_{ij}=1}\}$;
Step 3: Formulate the high-percentile SLA tail latency
model and the scenario subproblems in time slot $t$ with
Formulation (19a). Let $C$ denote the incumbent solution
of scenario subproblems.
$C_{i'j't}^{w\lambda} \leftarrow X_{i'j'} (\sum_{k \in K} x_{i'j'kt\lambda}^R p_{i'j'kt\lambda}^R + x_{i'j't\lambda}^O p_{i'j't\lambda}^O)$;
Step 4: Solve the scenario subproblems with MathProg
and GLPK formulated in Step 2. Let $x$ denote the
optimized resource provision plan.
  For $w$ to # for all the workloads do
      $Min \ \sum_{\lambda \in \Lambda_t} P_\lambda C_{i'j't}^{w\lambda}$;
  end
```

the workload to target data center and the workload will reside in target data center during $T$. The resource demand of workload may dynamically change in every RPP time slot $t$. So, we use (19a) to update resources provision plan for each workload after every RPP time slot $t$.

### 3) SYSTEM OVERVIEW

The dispatcher shown in Figure 2 is the core module of 2DRP. It is responsible for the workload distribution decision and resource provision planning. Inputs of the dispatcher include *Price* and $S$. The parameter *Price* includes three parts: the VM instance types, reservation contracts and the unit price of the resource. $S$ represents the Internet requests for geo-distributed interactive services. With these inputs, the dispatcher can generate the optimized resource provision plan and distributed the Internet requests to the target data center group.

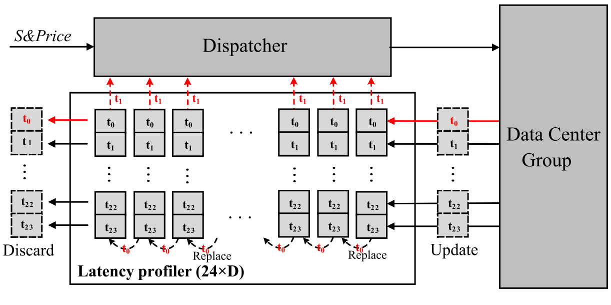


**FIGURE 2.** Dispatcher architecture.

In the first scheduling, we manually set target data center group and develop resource provision plan under the expected QoS requirements for geo-distributed interactive

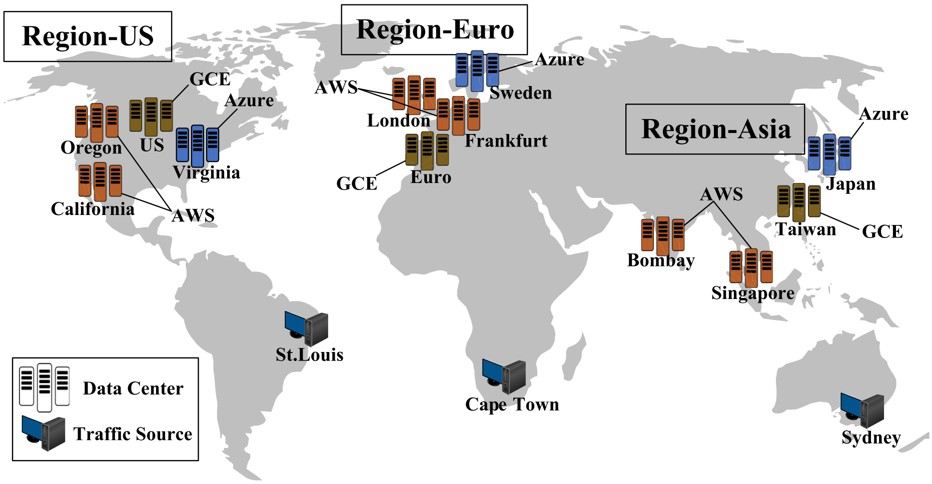


**FIGURE 3.** Location distribution information of data centers for multi-CSPs.

**TABLE 1.** Price policy of common VM instances for AWS, Azure and GCE in Western US data centers.

| CSP | Price($) | | | VM Instance Configuration | | | |
|---|---|---|---|---|---|---|---|
| | Up-front | Reservation | On-demand | Instance name | VCPU(Core) | Memory(G) | Disk(G) |
| AWS (one year) | 4350 | 0.497 | 0.84 | c3.4xlarge | 16 | 30 | 2 × 160 |
| AWS (three years) | 8265 | 0.315 | 0.84 | c3.4xlarge | 16 | 30 | 2 × 160 |
| Azure | N/A | N/A | 0.997 | F16 | 16 | 32 | 256 |
| GCE | N/A | N/A | 0.8727 | n1-standard-8 | 8 | 30 | 320 |

service. In next WDD time slot, the data center group and plan are under the historical records of resource utilization and probability tail latency. The latency profiler that showed in Figure 2 is responsible for the estimation of probability tail latency. We firstly record 24 × $D$ historical execution information in 2DRP. Each record includes the information: allocated computing resource, WAN network traffic and the latency for Internet request. With probability distribution of these records, the estimated resource demand can be obtained through probability theory. For example, the latency gaps between estimation and actual can be eliminated by adding or reducing a fixed amount of resources. The numbers of historical records $D$ can be set to a different value in different estimation precision requirements since more records can increase prediction accuracy, but the performance will decrease. Especially, 2DRP takes the historical execution records as $D$ scenarios to find optimized workload distribution and develop resource provision plan.

## VI. EVALUATION

After investigating resource price policies for several CSPs, we evaluated 2DRP using simulations by replaying Internet requests for geo-distributed interactive services that were collected from a large production Google data center workload traces. The main results are listed as follows.

1) By comparing across a set of alternatives, 2DRP achieves up to 24% of the optimized welfare compared with On-demand based infrastructure, and up to 10% compared with both Reservation based infrastructure and the OCRP infrastructure.

2) With 2DRP, more than 97% of Internet requests for geo-distributed interactive services can meet the high-percentile SLA tail latency constraint.

3) We demonstrate that the data center groups selected by 2DRP adapt to load variations and that 2DRP is robust to variations in network and request characteristics.

### A. METHODOLOGY

#### 1) MULTI-CLOUDS

We conducted some trace-driven experiments on data centers of real-world CSPs including Amazon AWS, Microsoft Azure and GCE. Data centers owned by these CSPs are geography distributed in different locations. Figure 3 shows the general location information of several common data centers and CSPs. Data centers are mainly structured in three regions: Region-US, Region-Euro and Region-Asia. In every region, different CSPs provide more than one resource provision policy. For example, AWS provides one year and three years reservation resource or the on-demand resource, while Azure and GCE only provide the on-demand resource. The unit price of the resource is different from data centers. We investigated the unit price of VM instance of three CSPs mentioned above. We list part of the price information of typical VM instances for data centers located in the Western US as Table 1. Details about the resource provision policy and price information are shown in [14]–[16].

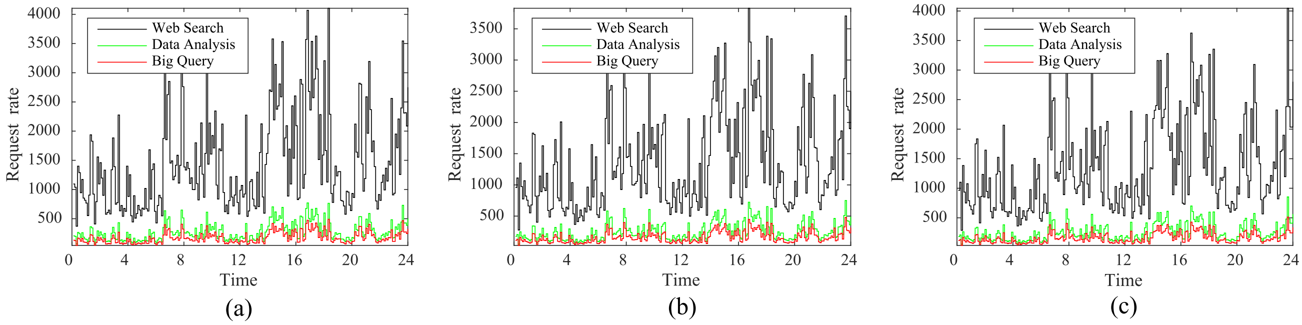


**FIGURE 4.** Google 24h data center workload traces. (a) Region-US; (b) Region-Euro; (c) Region-Asia.

### 2) REAL-WORLD GEO-DISTRIBUTED INTERACTIVE SERVICE

We take the Google data center workload traces as the basic trace-driven experimental data to replay the Internet requests for geo-distributed interactive services. We mainly concern about the following type of geo-distributed interactive service: Real-time Data Analysis, Big Query and Web Search. Each type of interactive service includes several workloads, denoted as *job* in the traces, which are distributed to one or more data centers in different regions. They are latency sensitive and take no more than a few seconds to complete. Google records CPU usage, memory usage and disk usage for Internet requests. Each Internet request represents one *task*, and one *job* contains several *tasks* in the workload traces. It reflects that one workload requires several Internet requests to complete. We use the traces of resource utilization and request rate for the corresponding workload to predict the primary resource demand. The requests rate of three types of geo-distributed interactive service is shown in Figure 4. After the geo-distributed interactive service online, traces of resource utilization and corresponding latency will be increased. Then in the upcoming WDD time slot, the workload distribution and resources allocation will become more accurate.

### 3) SOLVER

We built 2DRP with AMPL linear programs modeling language and resolve the optimization functions using the GLPK solver.

### 4) BASELINE

We configured 2DRP with $\Pr^{SLA} = 0.95$, $t = 1h$ and $T = 24h$ in the basic experiment. We can obtain the workload distributions $X$, resource provision plan $x$ and the corresponding resource provision cost $C$. To illustrate our design is efficient, we compared 2DRP with:

1) **OCRP**

In the aim of making an optimized resource provision decision, OCRP takes into account the demand uncertainty from cloud consumer and price uncertainty from CSPs to adjust the tradeoff between reservation and on-demand resources. The resource provision decision is achieved by formulating a cost minimization problem with multistage recourse, which solves by benders decomposition and sample average approximation. Compared with 2DRP, OCRP estimates resource requirements for workloads coarse-grained and does not concern about the tail latency constraint.

2) **Reservation based**

Compared to the on-demand VM instance, CSPs provide a significant discount for the reserved VM instances. Cloud consumer must make reservation contract to reserve VM instance in data center group. The geo-distributed interactive service can use the reserved VM instances during the reservation term. With the **All Upfront** option, the cloud consumer pays for the entire reserved instance term with one upfront payment. Also, when reserved VM instances are assigned to a specified availability zone, they provide a capacity reservation, giving the geo-distributed interactive service additional confidence to launch VM instances when need them to provide better QoS.

3) **On-demand based**

With the on-demand VM instances, cloud consumers pay for the compute capacity hourly with no long-term commitments or upfront payments. The compute capacity can increase or decrease depending on the constraint of the high-percentile SLA tail latency and only pay the specified hourly rate for the VM instances the data center workload used.

## B. THE OPTIMIZATION RESULTS

### 1) DEMAND UNCERTAINTY AND CONVERGENCE ANALYSIS

Google data center workload traces are the execution records of Internet requests. We have carried on a thorough study of the workload traces. We found that Internet request for the same geo-distributed interactive service is periodic, so we counted the Internet request for the same *jobid* in a day. After that, we can get the total relevant data center resource utilization and WAN network traffic. The primary estimation of resource demand is based on Poisson distribution. For each Internet request, we generate up to 1000 demand estimations. Each possible resource requirement corresponds to a specified probability $P_\lambda$. Figure 5 (a)-(d) shows the complex and diverse of resource requirement estimation of geo-distributed interactive service.

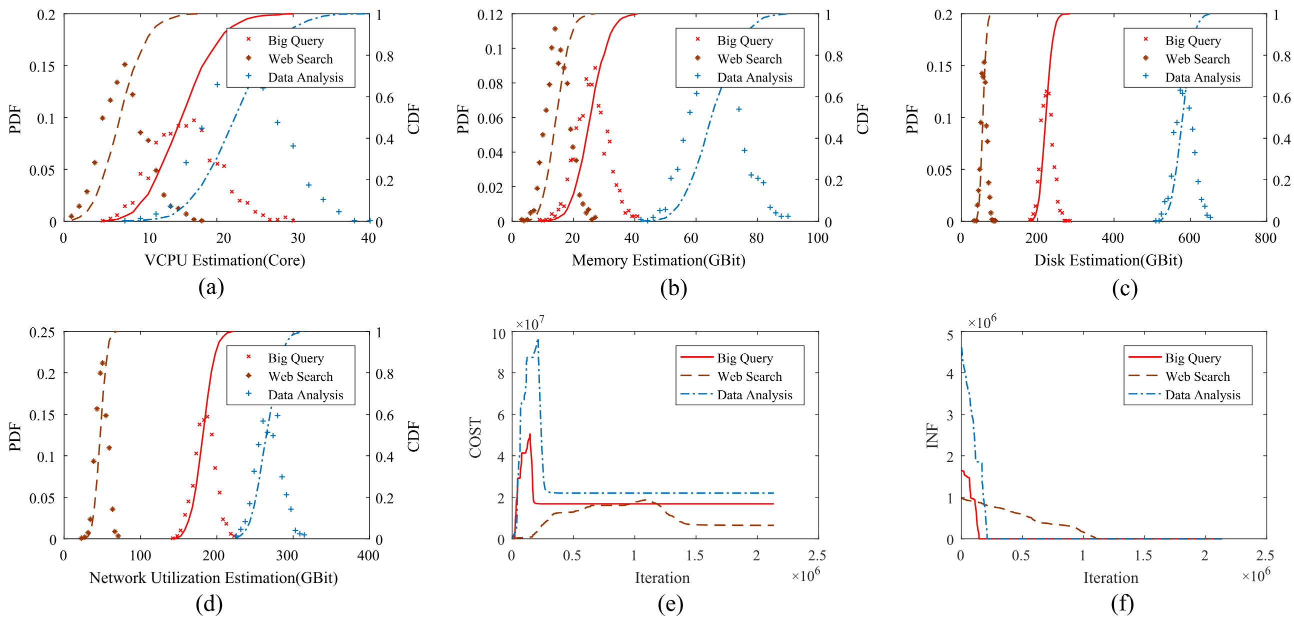


**FIGURE 5.** Uncertain resource demand and performance. (a) VCPU; (b) memory; (c) disk; (d) network utilization; (e) resource provision cost; (f) offset of the cost.

After generating estimated resource demand for more than 10000 Internet requests. We configure the 2DRP with the current data center price of CSPs mentioned above and solve the two-stage problem with GLPK. We can get the optimized workload distribution and resource provision plan for all geo-distributed interactive services. The convergence performance is shown as Figure 5 (e) and (f). Through limited iterations, three type of geo-distributed interactive services (Real-time Data Analysis, Big Query and Web Search) can be distributed to the lowest cost data center group and can obtain the optimized resource provision plan.

### 2) RESOURCE PROVISION COST AND PERFORMANCE

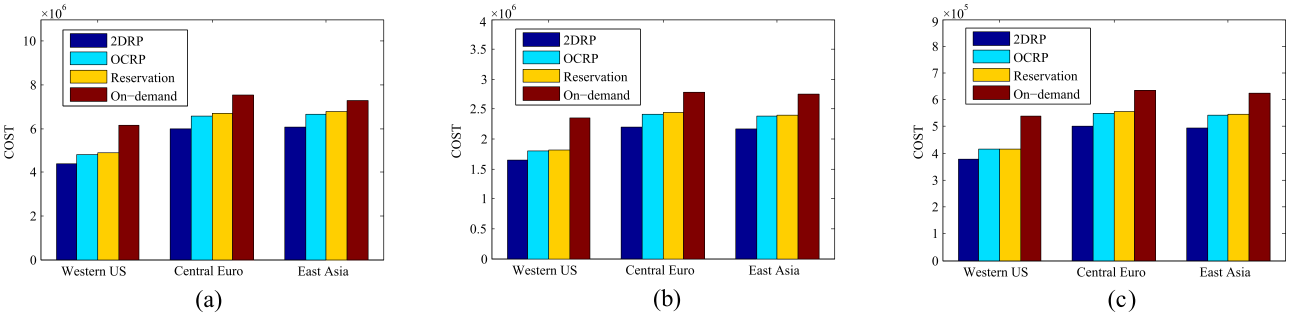
The characteristics of 2DRP determine the features of workload distribution: (1) If the workload is assigned to one data center and make reservation contract, the workload will reside in the data center at the reservation term. In every RPP time slot, 2DRP allocates a number of running reserved VM instances to workloads, and on-demand VM instances if needed, for reaching Internet request's high-percentile SLA tail latency constraint. (2) If the workload is assigned to one data center at the beginning of WDD time slot, and the resource provision plan is on-demand based, 2DRP allocates a number of running on-demand VM instances to workloads, no reserved VM instance, for meeting the high-percentile SLA tail latency constraint. Workload will reside in that data center during WDD time slot iff 2DRP finds another lower cost data center in next WDD time slot because there is no reservation contract restriction. Of course, if workloads scheduling to other data center can reduce more cost, then 2DRP can break restriction of the reservation contract.

To better calculate the resource provision cost, we restrict the deployment time of geo-distributed interactive services as three years. Hence, we calculate the resource provision cost totally in three years. The resource provision cost of geo-distributed interactive services under several distribution strategies are shown in Figure 6. In every region, we can know that 2DRP performs better than other workload distribution infrastructure and can save resource provision cost in varying degrees.

The geo-distributed interactive service needs data centers of every region to complete the latency sensitive service. So we sum the resource provision cost in all regions. The total cost is shown in Table 2. With 2DRP, we can save more than 9% resource provision cost compared with the OCRP infrastructure and save up to 10% compared with Reservation based infrastructure. 2DRP can save up to 24% compared with On-demand based infrastructure. Reservation based is more cost-effective than On-demand based, because if deployment time of geo-distributed interactive service is long enough and the resource price is relatively stable, the more the reserved VM instances allocate to interactive service workload, the less the resource provision cost.

**TABLE 2.** Resource provision cost for different solutions.

| Solution | Resource Provision Cost ($) | | |
|---|---|---|---|
| | Big Query | Data Analysis | Web Search |
| 2DRP | 10011458.57 | 3669256.49 | 1383849.16 |
| OCRP | 11012604.42 | 4036182.14 | 1522234.08 |
| Reservation | 11184702.94 | 4067186.36 | 1532462.91 |
| On-demand | 12793880.14 | 4822396.26 | 1818654.56 |



(a) (b) (c)

**FIGURE 6.** **Comparison of resource provision cost for three types of geo-distributed interactive service. (a) Real-time Data Analysis; (b) Big Query; (c) Web Search.**

### 3) THE HIGH-PERCENTILE SLA TAIL LATENCY CONSTRAINT

One important feature of 2DRP is that Internet requests for geo-distributed interactive service can meet the high-percentile SLA tail latency constraint. The high-percentile SLA tail latency enables 2DRP to minimize the resource provision cost while fulfilling the QoS requirement of Cloud consumer. We use P95 as the basic configuration to perform 2DRP in the basic experiment. Mapping function for probability distribution and latency CDFs is used to estimate the probability of Internet request met the latency constraint, whether the Internet request exceeds the latency threshold in operation. Figure 7 shows the high-percentile tail latency satisfaction condition for 10000+ Internet requests for three types of geo-distributed interactive services. We can know that more than 97% of Internet requests can meet the P95 constraint. We can ensure that the tail latency of geo-distributed interactive service has a great probability not exceeding the latency threshold and can satisfy the QoS requirement.

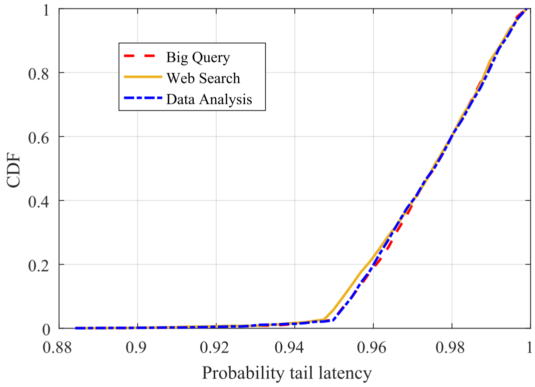


**FIGURE 7.** **CDF for probability tail latency.**

## C. SLA TAIL LATENCY CONSTRAINT

Naturally, the less stringent the high-percentile SLA tail latency of data center group can provide, the more resource provision cost can be saved. Higher resource provision cost can ensure higher percentile of tail latency because it is difficult to provide better QoS at low cost in business. We primitively believe that low price of data center resources weaken the probability tail latency while higher resource price can provide better QoS.

To configure the value of $Pr^{SLA}$ and evaluate the high-percentile SLA tail latency model of 2DRP, we primarily take the benefit of P95 as the high-percentile SLA tail latency constraint. The value of $Pr^{SLA}$ reflects the QoS satisfaction levels. If we set a bigger value of $Pr^{SLA}$, the customer is satisfied, but the resource provision cost of geo-distributed interactive service increase. If we set a lower value of $Pr^{SLA}$, the resource provision cost is maintained a low level, but QoS cannot be guaranteed which is reflected in the decrease of the probability tail latency. We change the value of $Pr^{SLA}$ and perform 2DRP to evaluate how the value of $Pr^{SLA}$ influences resource provision cost.

We take the value of $Pr^{SLA}$ as 0.90, 0.95 and 0.98 to configure 2DRP. Figure 8(a) shows the relationship between the high-percentile SLA tail latency constraint and the resource provision cost. The resource provision cost is decreased if the geo-distributed interactive service has loosed tail latency constraints. Data center resources with a higher price can provide better QoS and at last improve the high-percentile tail latency. In contrast, if we choose the data center group with lower resource price, we must realize the unexpected risk of not meeting the expected high-percentile SLA tail latency.

We use the 0.95 (P95) as the criterion of workload distribution and resource provision planning in the evaluation. It balances the cost minimizing and QoS requirement. The cost keeps at minimum while the tail latency can meet the high-percentile SLA tail latency constraint. However, if customers can tolerance bad QoS to save memory or willing to spend more money to gain better QoS, 2DRP can be configured to distinct these requirements. The value of $Pr^{SLA}$ compromises the probability tail latency and resource provision cost.

## D. SENSITIVITY ANALYSIS

### 1) LATENCY PREDICTION ERROR

We now consider the relationship between latency prediction error and the high-percentile SLA tail latency constraint; the result is shown in Figure 8(b). Latency prediction error is decreased with loose $Pr^{SLA}$. The reason is that if Internet requests can fulfill the $Pr^{SLA} = 0.95$ tail latency constraint,

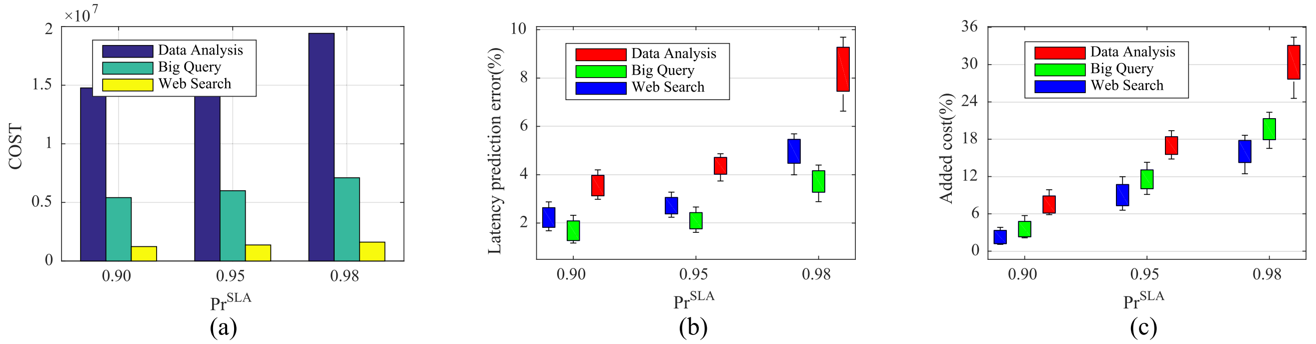


**FIGURE 8.** Cost comparisons and latency prediction error. (a) cost comparisons; (b) latency prediction error; (c) added resource provision cost.

it can satisfy the $\Pr^{SLA} = 0.90$ tail latency constraint as well. The tail latency prediction becomes accuracy when the $\Pr^{SLA}$ keeps in low level. With higher prediction error ratio, the added cost for selecting the wrong data center is increased. Figure 8(c) shows the increased resource provision cost due to more stringent $\Pr^{SLA}$. We can realize that the error prediction of latency increases slowly as the $\Pr^{SLA}$ increases, but reflects a noticeable increase in resource provision cost. The reason is that the reservation resources reduce and on-demand resources with more interference growth rapidly, the increased error tail latency prediction becomes more pronounced.

### 2) INTERFERENCE

We expect to allocate data center resources with the lowest price to interactive service workloads as many as possible. However, since it does not take data center capacity and interference into consideration, a data center may become overloaded and hence may not meet the high-percentile SLA tail latency constraint. The data center utilization can influence the performance of geo-distributed interactive services. If the data center utilization keeps at low, the error ratio of the probability tail latency estimation can tolerate. Otherwise, the error ratio will increase and as a result reduces the precision of workload distribution and resource provision planning. It finally causes the added resource provision cost. Hence, 2DRP attempts to distribute workloads to data centers with low resource price and the data center under possible low utilization. If the data center has many pre-running workloads, then the interference may enlarge the error ratio of the tail latency prediction.

## VII. RELATED WORK

Zhao *et al.* [8] proposed an online algorithm for distributing the workload to lowest cost data center. They achieve the distribution with VM migration. Because the running time of workloads is longer than the pre-scheduling service time, with precise workload arrival prediction, workloads can complete the execution within the deadline. Altmann and Kashef [31] proposed COMBSPO to minimize the resource provision cost in federated hybrid clouds. With VM migrations technology, it can compute the deployment cost of services and the data transmission cost in public clouds. However, both methods do not consider the cost of VM migration which is the main contribution to the resource provision cost. Quasar [32] is a cluster management system that performs coordinated resource allocation and assignment for distributed analytics frameworks and web-serving applications. Quasar achieves the resource allocation by using the big data analysis tools to analyze the impact of allocation (scale-up and scale-out), resource type (heterogeneity), and interference on workload's performance. However, Quasar only concerns about the single data center, while 2DRP concerns about multiple CSPs.

Lucas-Simarro *et al.* [33] proposed a cloud broker architecture to minimize the resource provision cost. This architecture is based on a prediction model in dynamic pricing scenarios. They concern about the performance of workloads with minimum data center resources. Xiao *et al.* [34] proposed a concept of "skewness" to measure the unevenness of the servers in data center. Applications running in the data center have different states, which result in dynamic resource requirements. With dynamic resource, Cloud can provide minimum resources to support the cluster service. However, both methods may not be able to meet the QoS requirement of Cloud consumers, and the workload performance cannot ensure the Internet Cloud Service met the QoS requirements because tail latency not only relates to the processing efficiency of data center resources but also the WAN network condition.

OCRP [6] is an efficient resource provision algorithm that obtained by formulating and solving optimization function with multistage recourses. OCRP applies the SAA approach to address the problem of a large set of workloads, which can effectively save the total resource provision cost. However, OCRP does not consider QoS requirement of Cloud consumers and does not design for geo-distributed interactive service. Greenberg *et al.* [35] detailed the contribution of the performance of data center computing resources and the infrastructure of WAN network to the resource provision cost. They addressed the problem of how to improve the efficiency of the data center resources to reduce resource provision cost. However, they do not concern about dynamic pricing strategy of data center resource. Gu *et al.* [7] primarily studied an optimization function to save over resource provision cost for workloads deployed in geo-distributed data centers. They take advantage of 2-D Markov chain to drive the average workload completion time in closed-form and develop resource provision plan. However, the cost model is complicated and not easy to apply to real-world data centers. Furthermore, these approaches [36], [37] focus on the costs of Get/Put operation of data sets and the network packet transmission which do not apply to the geo-distributed interactive service workloads.

## VIII. CONCLUSION

This work aims to minimize the resource provision cost of geo-distributed interactive services with multi-CSPs. We have made a contribution to developing a dynamic cost model and a high-percentile SLA tail latency model. With the cost model and the latency model, we design 2DRP with the Stochastic Programming and the decomposition technique to distribute the workloads to data center group, and the geo-distributed interactive service needs minimum resource provision cost to meet the high-percentile SLA tail latency constraint. At first, 2DRP makes workload distribution decision and develops primary resource provision plan with uncertain resource price. Recourse provision plan can then be adjusted to compensate for any bad effects that might have been experienced with uncertain resource demand. The performance evaluation of 2DRP is conducted with several studies and simulations. Internet requests of geo-distributed interactive services can meet the high-percentile SLA tail latency constraint by using the real Clouds belonging to multi-CSPs and under variety resource provision policies. Compared with other optimization methods, 2DRP can save up to 10% of the resource provision cost compared with the OCRP infrastructure, the reservation based infrastructure and up to 24% compared with the on-demand based infrastructure.

## REFERENCES

[1] Y. Chen, S. Alspaugh, and R. Katz, "Interactive analytical processing in big data systems: A cross-industry study of MapReduce workloads," *Proc. VLDB Endowment*, vol. 5, no. 12, pp. 1802–1813, 2012.

[2] Google. (2017). *Google Cloud DataLab*. [Online]. Available: https://cloud.google.com/datalab/

[3] Y. Xu, J. Yao, H.-A. Jacobsen, and H. Guan, "Cost-efficient negotiation over multiple resources with reinforcement learning," in *Proc. 25th IEEE/ACM Int. Symp. Qual. Service*, Jun. 2017, pp. 1–6.

[4] J. Yao, H. Zhou, J. Luo, X. Liu, and H. Guan, "COMIC: Cost optimization for Internet content multihoming," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 7, pp. 1851–1860, Jul. 2015.

[5] T. Deng, J. Yao, and H. Guan, "Maximizing profit of cloud service brokerage with economic demand response," in *Proc. 37th IEEE Conf. Comput. Commun.*, Apr. 2018, pp. 1907–1915.

[6] S. Chaisiri, B.-S. Lee, and D. Niyato, "Optimization of resource provisioning cost in cloud computing," *IEEE Trans. Services Comput.*, vol. 5, no. 2, pp. 164–177, Apr./Jun. 2012.

[7] L. Gu, D. Zeng, P. Li, and S. Guo, "Cost minimization for big data processing in geo-distributed data centers," *IEEE Trans. Emerg. Topics Comput.*, vol. 2, no. 3, pp. 314–323, Sep. 2014.

[8] J. Zhao, H. Li, C. Wu, Z. Li, Z. Zhang, and F. C. M. Lau, "Dynamic pricing and profit maximization for the cloud with geo-distributed data centers," in *Proc. 33rd IEEE Conf. Comput. Commun.*, Apr./May 2014, pp. 118–126.

[9] M. Al-Ayyoub, M. Wardat, Y. Jararweh, and A. A. Khreishah, "Optimizing expansion strategies for ultrascale cloud computing data centers," *Simul. Model. Pract. Theory*, vol. 58, pp. 15–29, Nov. 2015.

[10] B. Martens, M. Walterbusch, and F. Teuteberg, "Costing of cloud computing services: A total cost of ownership approach," in *Proc. 45th Hawaii Int. Conf. Syst. Sci.*, Jan. 2012, pp. 1563–1572.

[11] Y. Zhang, J. Yao, and H. Guan, "Intelligent cloud resource management with deep reinforcement learning," *IEEE Cloud Comput.*, vol. 4, no. 6, pp. 60–69, Nov./Dec. 2017.

[12] J. Yao, Q. Lu, H. Jacobsen, and H. Guan, "Robust multi-resource allocation with demand uncertainties in cloud scheduler," in *Proc. 36th IEEE Symp. Reliable Distrib. Syst.*, Sep. 2017, pp. 34–43.

[13] J. Yao, X. Liu, X. Zhu, and H. Guan, "Control of large-scale systems through dimension reduction," *IEEE Trans. Services Comput.*, vol. 8, no. 4, pp. 563–575, Jul. 2015.

[14] Google. (2017). *Google Computing Engine Pricing*. [Online]. Available: https://cloud.google.com/pricing/

[15] Amazon. (2017). *Amazon EC2 Pricing*. [Online]. Available: https://aws.amazon.com/emr/pricing/

[16] Microsoft. (2017). *Azure Pricing*. [Online]. Available: https://azure.microsoft.com/en-us/pricing/

[17] A. Thusoo *et al.*, "Data warehousing and analytics infrastructure at Facebook," in *Proc. 36th ACM Int. Conf. Manage. Data (SIGMOD)*, 2010, pp. 1013–1020.

[18] Y. Chen, S. Alspaugh, D. Borthakur, and R. Katz, "Energy efficiency for large-scale mapreduce workloads with significant interactive analysis," in *Proc. 7th ACM Eur. Conf. Comput. Syst. (EuroSys)*, 2012, pp. 43–56.

[19] J. Yao, H. Guan, J. Luo, L. Rao, and X. Liu, "Adaptive power management through thermal aware workload balancing in Internet data centers," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 9, pp. 2400–2409, Sep. 2015.

[20] H. Guan, J. Yao, Z. Qi, and R. Wang, "Energy-efficient SLA guarantees for virtualized GPU in cloud gaming," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 9, pp. 2434–2443, Sep. 2015.

[21] B. Peng, H. Zhang, J. Yao, Y. Dong, Y. Xu, and H. Guan, "MDev-NVMe: A NVMe storage virtualization solution with mediated pass-through," in *Proc. 24th USENIX Annu. Tech. Conf.*, 2018, pp. 665–676.

[22] Y. Xu, J. Yao, Y. Dong, K. Tian, X. Zheng, and H. Guan, "Demon: An efficient solution for on-device MMU virtualization in mediated pass-through," in *Proc. 14th ACM Int. Conf. Virtual Execution Environ. (VEE)*, 2018, pp. 57–70.

[23] B. Peng, J. Yao, Z. Qi, and H. Guan, "HybridPass: Hybrid scheduling for mixed flows in datacenter networks," in *Proc. 32nd IEEE Int. Parallel Distrib. Process. Symp.*, May 2018, pp. 1000–1009.

[24] X. Wang, J. Zhu, and Y. Shen, "Network-aware QoS prediction for service composition using geolocation," *IEEE Trans. Services Comput.*, vol. 8, no. 4, pp. 630–643, Jul. 2015.

[25] M. Lin, Z. Liu, A. Wierman, and L. L. H. Andrew, "Online algorithms for geographical load balancing," in *Proc. 3rd Int. Green Comput. Conf.*, Jun. 2012, pp. 1–10.

[26] Z. Liu, M. Lin, A. Wierman, S. Low, and L. L. H. Andrew, "Greening geographical load balancing," *IEEE/ACM Trans. Netw.*, vol. 23, no. 2, pp. 657–671, Apr. 2015.

[27] X. Zhao, J. Yao, P. Gao, and H. Guan, "Efficient sharing and fine-grained scheduling of virtualized GPU resources," in *Proc. 38th IEEE Int. Conf. Distrib. Comput. Syst.*, Jul. 2018, pp. 742–752.

[28] M. A. Islam, A. Gandhi, and S. Ren, "Minimizing electricity cost for geo-distributed interactive services with tail latency constraint," in *Proc. 7th Int. Green Sustain. Comput. Conf.*, Nov. 2016, pp. 1–8.

[29] D. Serrano *et al.*, "Towards QoS-oriented SLA guarantees for online cloud services," in *Proc. 13th IEEE/ACM Int. Symp. Cluster, Cloud, Grid Comput.*, May 2013, pp. 50–57.

[30] T. Zhu, D. S. Berger, and M. Harchol-Balter, "SNC-meister: Admitting more tenants with tail latency SLOs," in *Proc. 7th ACM Symp. Cloud Comput. (SoCC)*, 2016, pp. 374–387.

[31] J. Altmann and M. M. Kashef, "Cost model based service placement in federated hybrid clouds," *Future Gener. Comput. Syst.*, vol. 41, pp. 79–90, Dec. 2014.

[32] C. Delimitrou and C. Kozyrakis, "Quasar: Resource-efficient and QoS-aware cluster management," in *Proc. 19th Int. Conf. Archit. Support Program. Lang. Oper. Syst. (ASPLOS)*, 2014, pp. 127–144.

[33] J. L. Lucas-Simarro, R. Moreno-Vozmediano, R. S. Montero, and I. M. Llorente, "Cost optimization of virtual infrastructures in dynamic multi-cloud scenarios," *Concurrency Comput., Pract. Exper.*, vol. 27, pp. 2260–2277, Jun. 2015.

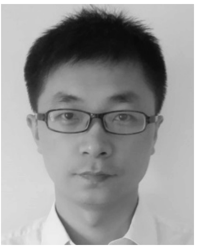
[34] Z. Xiao, W. Song, and Q. Chen, "Dynamic resource allocation using virtual machines for cloud computing environment," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 6, pp. 1107–1117, Jun. 2013.

[35] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, "The cost of a cloud: Research problems in data center networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 39, no. 1, pp. 68–73, 2009.

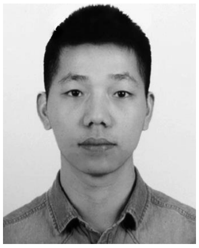
[36] G. Liu and H. Shen, "Minimum-cost cloud storage service across multiple cloud providers," *IEEE/ACM Trans. Netw.*, vol. 25, no. 4, pp. 2498–2513, Aug. 2017.

[37] Z. Wu, M. Butkiewicz, D. Perkins, E. Katz-Bassett, and H. V. Madhyastha, "SPANStore: Cost-effective geo-replicated storage spanning multiple cloud services," in *Proc. 24th ACM Symp. Oper. Syst. Princ. (SOSP)*, 2013, pp. 292–308.

**FEI HU** received the M.A. degree and the Ph.D. degree in computer science from Northwestern Polytechnical University, Xi'an, Shaanxi, China. He is currently the Vice Director of the Science and Technology on Avionics Integration Laboratory. His recent research interests include system architecture and applications of model-based systems engineering.



**QINGCHUN LIU** received the M.A. degree in electrical engineering from the Nanjing University of Aeronautics and Astronautics, Jiangsu, China. He is currently a Senior Systems Engineer with the Science and Technology on Avionics Integration Laboratory. His recent research interests include system architecture and applications of model-based systems engineering.



**JIAHONG WU** received the B.E. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2015. He is currently a Graduate Student with the Shanghai Key Laboratory of Scalable Computing and Systems, School of Software, Shanghai Jiao Tong University. His research interests mainly include cloud computing, real-time system, and networking.

**JIANGUO YAO** (SM'15) received the B.E., M.E., and Ph.D. degrees from Northwestern Polytechnical University, Xi'an, Shaanxi, China, in 2000, 2007, and 2010, respectively. He is currently an Assistant Professor with Shanghai Jiao Tong University. His research interests include feedback control applications, real-time and embedded computing, power management of data centers, and cyber-physical systems.