2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing

RO-BURST: A Robust Virtualization Cost Model for Workload Consolidation over Clouds

Jianzong Wang[1], Rui Hua[1], Yifeng Zhu[2], Jiguang Wan[1], Changsheng Xie[⊠1], Yanjun Chen[1]
[1]School of Computer Science and Technology, Huazhong University of Science and Technology, China
[1] Wuhan National Laboratory for Optoelectronics, China
[1] Key Laboratory of Data Storage Systems, Ministry of Education of China
[2] Department of Electrical and Computer Engineering, University of Maine, USA
[⊠] Corresponding Author: cs_xie@hust.edu.cn

***Abstract*—As more public cloud computing platforms are emerging in the market, a great challenge for these Infrastructure as a Server (IaaS) providers is how to measure the cost and charge the Software as a Service (SaaS) clients for the cloud computing services. This problem is compounded as virtualization technology is deployed in many cloud platforms to consolidate servers and improve their utilization. This paper studies three different but related models for apportioning costs in a private or public cloud environment supported by virtualized data centers. With given workload placement scenarios and randomly selected workloads, these models estimate the cost for each workload. Through simulations and thorough comparisons of the results, we finally champion the RO-BURST model tailored for the service providers' need, that is characterized by robustness and burstiness. What is more, we import Cost Volatility Factors to ensure that our model is able to adjust itself to the market and multiform demands in power and hardware components, such as disks and CPU, showing its compatibility and extensibility. We also come up with a pricing strategy with respect to servers the workload employs, which generates an applicable and less placement-sensitive fee for the clients.**

***Keywords*-IaaS and SaaS, Cost model, Pricing strategy, Workloads, and Clouds;**

I. Introduction

A. Preface

In the past, many enterprises, especially the small ones, had to host their computing service and software applications on infrastructure they owned and maintained at high costs. Recently they start to resort to an advanced technology for more cost-efficient and flexible IT service. The bringing forward of cloud makes the computing power a kind of new-style commodity possible with the computing power of 10 trillion times per second, enough to simulate nuclear explosion and predict climate fluctuations. In addition, virtualization helps to expand hardware capacity and effectively reduces the expenses through dividing an integrated server into considerable different virtual resources. Therefore, under the pay-as-you-go model [1], IaaS providers must utilize the virtualization to obtain the flexibility and stability that those end-users require in both private or public clouds. One of those present challenges for IaaS providers is that when virtual machines and different resource demands are assigned into resource pools, how to accurately estimate the costs and place reasonable charges to SaaS clients according to the demand of each workload in the competitive cloud environments.

B. Example of Motivation

Why is an efficient and reliable cost estimate model of great importance to both IaaS providers and SaaS clients?First of all, workload characteristics can influence the service charge. Two applications with the same average demand of computation resources could be charged differently. Figure 1 shows the demands of disks of two different workloads. Workload A shares 22% peak demands in disks, and its average demands lies in 0.2%. Workload B also has 0.2% average demands but the peak value only amounts to 6%. We can easily infer that they both share similar average demands in disks but huge difference in peak demands, which conducts dissimilarity in bursty demands. Such dissimilarity makes the number of servers that host each workload varied. As a result, workload A must be assigned to more servers than workload B to meet its much higher peak demands.Secondly, workload placement strategies also impact the service costs. For example, we assumed that 10 workloads are allocated into a resources pool with two servers. In the first scenario, 8 workloads are assigned to one server and the remaining 2 are assigned to the other server. In the second scenario, 5 workloads are assigned to each server. We have same amount of servers and workloads here, but the cost of each workload under such two placement scenarios may vary significantly.Thirdly, unallocated part of each server resulting from the influence of burstiness and workload placement strategies should also be taken into account when evaluate whether the cost model is reliable and robust enough.

C. Contributions

In this paper, we study and compare three different cost estimate models. We obtain 20 workloads on a company's shared service platform. After analyzing the costs and characteristics of various workloads in different workload placement strategies, we find that the third model, named as RO-BURST, is the best one and is deployable in real systems. This model has the following four salient features.

- RO-BURST finely reflects the bursty attribute of each workload over the cloud.
- RO-BURST takes into account the unallocated resources, this model is much less sensitive to workload placement and is more robust.
- Such model has strong malleability and compatibility,

978-0-7695-4691-9/12 $26.00 © 2012 IEEE
DOI 10.1109/CCGrid.2012.87

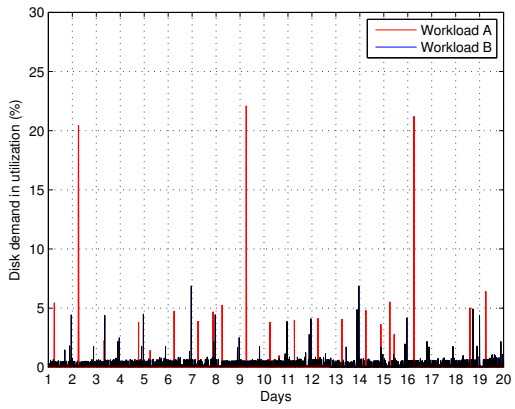

Fig. 1. Disk demands under workload A and B

which can not only be applied to disks, but also CPU, memory, network and power, etc.

- Based on the result of RO-BURST model, the service for SaaS clients can be appropriately priced by their IaaS provider, and such price is more competitive than others because RO-BURST is able to adjust itself to market trend.

The remainder of this paper is organized as follows. Section II presents background knowledge. Three cost models and pricing strategy are well defined in Section III. We discuss our experiments and evaluate the model and pricing strategy in Section IV. Section V introduces the related works. Conclusion and description of our future work are presented in the last section.

II. Background

A. SaaS and IaaS

Software as a Service (SaaS) providers, which are partly deployed on the basis of Infrastructure as a Service (IaaS), such as Google and Oracle, offer consumers the convenience to request applications directly over the cloud through Internet. Consumers do not need to install or manage infrastructure such as network, servers, operating system and storage, and avoid high cost of locally invested software and hardware. The IaaS model guarantees basic resources for the clients to host their applications, which involves complicated infrastructure management that could be handled by IaaS providers themselves like Amazon and Microsoft.

B. Cloud Storage

Cloud storage is a notion developed and extended from cloud computing. By employing cluster and grid technology, and distributed file systems, cloud storage leverage massive storage devices work together to provide the capacity and throughput required by end applications. Cloud computing users do not need to own the space of a specific storage device over the clouds but the data access service brought by whole cloud storage system. Strictly speaking, cloud storage is a kind of service rather than pure 'storage' itself. The core of cloud storage is to convert storage devices into storage service by integrating application software and storage devices. There are some key advantages for cloud storage, e.g., cloud storage providers offer distant system and data backup to keep the most important data of clients from being destroyed by natural disasters or man-made damage. As the emergence of PaaS and SaaS, in most circumstances, it is always an excellent choice for people to transfer local storage to the cloud for its simplification of management, reliability and cost-effectiveness.

C. Workload Consolidation

Before virtualization came into being, thousands of servers were used together, which could generate a lot of heat and thus shorten the working lifetime of these machines, and of course, accompanied with an increased costs on the hardware acquisitions and maintenance. In order to make IT infrastructure such as servers, network and database better meet the current and predictable future demands of different kinds of applications in the cloud environment, we often choose to simplify and optimize end-to-end infrastructures by bringing workload consolidation. Nowadays, there exist many tools to simulate such consolidation, one of which is HP capacity advisor [2]. The capacity advisor's purpose is to find optimal solution for workload placement. It firstly traverses former workloads to predict upcoming peak demands, and then finishes the search by applying genetic algorithm [3] while ensuring peak demands less than the capacity of the attribute for each server.

D. Cost Estimation in Cloud Environment

Among the biggest companies offering cloud computing, Amazon AWS [4] charges customers by the number of virtual instances an application occupies and how long it uses them: $cost=price \times t$ (t is the total running time, price is the price per VM), while Google's AppEngine charges by the number of CPU cycles a customer's application consumes, and recently Microsoft started to charge for its Azure cloud computing service. More and more large IT giants start to accelerate the development and cooperation of business model on cloud computing. Gartner Research predicts the market scale of cloud computing will amount to $150 billion by 2014, and small and medium will spend over $100 billion on cloud computing by 2014 [5].

This pay-as-you-go model makes resource-consumption based pricing particularly sensitive to how a system is designed, configured, optimized, monitored, and measured.

III. Cost Modeling

A. Service Framework

The architecture of our presented RO-BURST framework is shown in Figure 4. Numerous workloads of SaaS applications firstly go through the Consolidation Engine [6] which is able to find an appropriate workload placement [7] for each workload, and then the workloads will be assigned into different servers

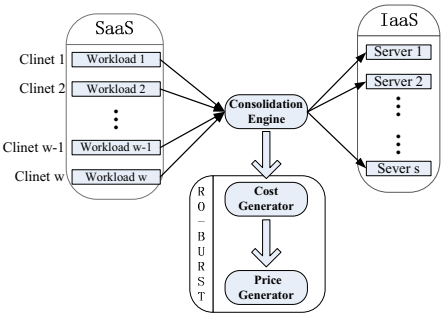

Fig. 2. Service Framework

lying in the resource pool.

Our RO-BURST system consists of two components, including Cost Generator and Price Generator. The Cost Generator takes the reports of server usage to automatically estimate the costs and the Price Generator implements the pricing strategy based on the result of cost estimates to come up with a reasonable and robust price for IaaS providers.

B. Parameter Explanations

In this paper, we limit our study to disk and CPU utilization when estimating the cost of an individual workload. However, our model can be easily extended to consider the utilization of other components, such as network and memory. Important parameters used in the three different models are elaborated in **Table I** and **Table II**

Complement explanations:

- If $U^{cpu}_{s,w}(t)$ is 20%, it means workload w takes 20% CPU resources of server s at time instant t; if $U^{cpu}_{s,w}(t)$ is 0, that means workload w does not use server s at time t.
- We introduce a new concept, named bursty utilization $\hat{U}^{cpu}_{s,w}$ and $\hat{U}^{disk}_{s,w}$ to model the burstiness of workloads. We define the burstiness as the difference between the peak and average utilizations. For example, the bursty utilization for CPU is $\hat{U}^{cpu}_{s,w} = max(U^{cpu}_{s,w}(t)) - mean(U^{cpu}_{s,w}(t))$.
- To incorporate the impact of burstiness of workloads and unallocated server resources of placement strategies, we propose the concepts of average cost ($\bar{C}^{cpu}_s$, $\bar{C}^{disk}_s$), bursty cost ($\hat{C}^{cpu}_s$, $\hat{C}^{disk}_s$,), and unused cost ($\check{C}^{cpu}_s$, $\check{C}^{disk}_s$).
- Our models also involve parameters α and β as the CPU and Disk volatility factors in order to adjust our models to current inconstant market trend (e.g., Thailand flood that occurred in November 2011 resulted in a considerable rise in hard disk price)
- We introduce the basic cost C^{cpu}_s and C^{disk}_s here for the consideration of importing volatility factors. In addition, according to current market, the costs of a 1TB disk and a CPU with 42.8MHz are \$128 and \$175 respectively, we set the value at \$100 for the convenience of calculations.

C. Three Models

Next, we introduce three cost estimate models for the IaaS providers. We refer to these as Average-based, Bursty-based, and RO-BURST.

Firstly, we consider the **Average-based** model (model 1) which only takes into account the average demand of a workload on a server. It is a model most widely employed by IaaS providers. For a given set of workloads W running a set of servers S, the model can be expressed as follows:

$$\bar{C}^{cpu}_{s,w} = C^{cpu}_s \times \frac{\bar{U}^{cpu}_{s,w}}{\sum_{w' \in W} \bar{U}^{cpu}_{s,w'}} \quad (1)$$

$$\bar{C}^{disk}_{s,w} = C^{disk}_s \times \frac{\bar{U}^{disk}_{s,w}}{\sum_{w' \in W} \bar{U}^{disk}_{s,w'}} \quad (2)$$

The total cost of Average-based model*(Model 1):*

$$\bar{C}_w = \sum_{s' \in S} \bar{C}^{cpu}_{s',w} \times \alpha + \sum_{s' \in S} \bar{C}^{disk}_{s',w} \times \beta \quad (3)$$

where $\bar{U}^{cpu}_{s,w}$ and $\bar{U}^{disk}_{s,w}$ are the average utilization of workload w on server s, and α and β are the volatility factors of CPU and disks.

The key weakness of the Average-based model is that it does not take the impact of burstiness and unallocated resources into considerations. To address this weakness, we propose the **Bursty-based** model (model 2). This model has three major steps. First of all, we calculate the average cost of a workload w on a server s. Next, we use both bursty demands of each workload hosted on server s and the result of first step to evaluate the bursty portion of the total cost of the server. In the third step, the unallocated cost of the server s is apportioned based on the bursty cost. At last, we sum up the results and bring volatility factors α and β to the model. The Bursty-based model is presented below.

Cost of CPU $\tilde{C}^{cpu}_{s,w}$***:***

$$\bar{C}^{cpu}_{s,w} = \bar{C}^{cpu}_s \times \frac{\bar{U}^{cpu}_{s,w}}{\sum_{w' \in W} \bar{U}^{cpu}_{s,w'}} \quad (4)$$

$$\hat{C}^{cpu}_{s,w} = \hat{C}^{cpu}_s \times \frac{\bar{C}^{cpu}_{s,w} + \hat{U}^{cpu}_{s,w} \times C^{cpu}_s}{\sum_{w' \in W} \bar{C}^{cpu}_{s,w'} + \hat{U}^{cpu}_{s,w'} \times C^{cpu}_s} \quad (5)$$

$$\check{C}^{cpu}_{s,w} = \check{C}^{cpu}_s \times \frac{\hat{C}^{cpu}_{s,w}}{\sum_{w' \in W} \hat{C}^{cpu}_{s,w'}} \quad (6)$$

$$\tilde{C}^{cpu}_{s,w} = \bar{C}^{cpu}_{s,w} + \hat{C}^{cpu}_{s,w} + \check{C}^{cpu}_{s,w} \quad (7)$$

where $\bar{C}^{cpu}_s$, $\hat{C}^{cpu}_s$, and $\check{C}^{cpu}_s$ are the average, bursty, and unused cost of CPU on server s, respectively, and $\hat{U}^{cpu}_{s,w}$ is the bursty CPU utilization of workload w on server s.

TABLE I
KEY PARAMETERS USED IN MODEL 1

Parameters	Descriptions	Remarks and Equations
$\bar{U}_{s,w}^{cpu}$, $\bar{U}_{s,w}^{disk}$	Mean CPU and disk physical utilization on server s by workload w	Value range : [0,100%]
C_s^{cpu}, C_s^{disk}	Basic costs of CPU and disk on server s.	We set the value at \$100 in this paper
α, β	Volatility index of CPU and disk costs, their values are actual cost/basic cost	The value of basic cost is \$100 in this paper

TABLE II
KEY PARAMETERS USED IN MODEL 2 AND 3

Parameters	Descriptions	Remarks and Equations
$\hat{U}_{s,w}^{cpu}$, $\hat{U}_{s,w}^{disk}$	Bursty CPU and disk physical utilization (i.e.,the difference between peak and average utilization) of workload w running on server s	Value range:[0,100%] $\hat{U}_{s,w}^{cpu} = max(U_{s,w}^{cpu}) - \bar{U}_{s,w}^{cpu}$, and $\hat{U}_{s,w}^{disk} = max(U_{s,w}^{disk}) - \bar{U}_{s,w}^{disk}$
$\check{U}_{s,w}^{cpu}$, $\check{U}_{s,w}^{cpu}$	Unused CPU and disk physical utilization of server s.	Value range: [0,100%] $\check{U}_{s,w}^{cpu} = 100\% - max(U_{s,w}^{cpu})$ and $\check{U}_{s,w}^{disk} = 100\% - max(U_{s,w}^{disk})$
$\bar{C}_s^{cpu}$, $\bar{C}_s^{disk}$, $\hat{C}_s^{cpu}$, $\hat{C}_s^{disk}$, $\check{C}_s^{cpu}$, $\check{C}_s^{disk}$	Average cost, bursty cost, and unused cost respectively.	$\bar{C}_s^{cpu} = C_s^{cpu} \times \bar{U}_{s,w}^{cpu}$ $\bar{C}_s^{disk} = C_s^{disk} \times \bar{U}_{s,w}^{disk}$ $\hat{C}_s^{cpu} = C_s^{cpu} \times \hat{U}_{s,w}^{cpu}$ $\hat{C}_s^{disk} = C_s^{disk} \times \hat{U}_{s,w}^{disk}$ $\check{C}_s^{cpu} = C_s^{cpu} \times \check{U}_{s,w}^{cpu}$ $\check{C}_s^{disk} = C_s^{disk} \times \check{U}_{s,w}^{disk}$
C_w	Total costs of workload w by using Average-based model	
$\hat{C}_w$	Total costs of workload w by using Bursty-based model	
$\check{C}_w$	Total costs of workload w by using RO-BURST model	

Cost of Disks $\tilde{C}_{s,w}^{disk}$***:***

$$\bar{C}_{s,w}^{disk} = \bar{C}_s^{disk} \times \frac{\bar{U}_{s,w}^{disk}}{\sum_{w' \in W} \bar{U}_{s,w'}^{disk}} \quad (8)$$

$$\hat{C}_{s,w}^{disk} = \hat{C}_s^{disk} \times \frac{\bar{C}_{s,w}^{disk} + \hat{U}_{s,w}^{disk} \times C_s^{disk}}{\sum_{w' \in W} \bar{C}_{s,w'}^{disk} + \hat{U}_{s,w'}^{disk} \times C_s^{disk}} \quad (9)$$

$$\check{C}_{s,w}^{disk} = \check{C}_s^{disk} \times \frac{\hat{C}_{s,w}^{disk}}{\sum_{w' \in W} \hat{C}_{s,w'}^{disk}} \quad (10)$$

$$\tilde{C}_{s,w}^{disk} = \bar{C}_{s,w}^{disk} + \hat{C}_{s,w}^{disk} + \check{C}_{s,w}^{disk} \quad (11)$$

where $\bar{C}_s^{disk}$, $\hat{C}_s^{disk}$, and $\check{C}_s^{disk}$ are the average, bursty, and unused cost of disks on server s, respectively, and $\hat{U}_{s,w}^{disk}$ is the maximum (peak) disk utilization of workload w on server s.

The total cost of Bursty-based model*(Model 2):*

$$\hat{C}_w = \sum_{s' \in S} \hat{C}_{s',w}^{cpu} \times \alpha + \sum_{s' \in S} \hat{C}_{s',w}^{disk} \times \beta \quad (12)$$

The third model, named as **RO-BURST** model, inherits the advantages of former two proposed models. However, this model differs from model 2 in that it uses measurement of a given set of S servers in the shared resource pool instead of the individual server to evaluate the average, bursty and unallocated cost respectively. RO-BURST model is well defined as follows.

Cost of CPU $\vec{C}_{s,w}^{cpu}$***:***

$$\bar{C}_{s,w}^{cpu} = \left(\sum_{s' \in S} \bar{C}_{s'}^{cpu} \right) \times \frac{\bar{U}_{s,w}^{cpu}}{\sum_{w' \in W}^{s' \in S} \bar{U}_{s',w'}^{cpu}} \quad (13)$$

$$\hat{C}_{s,w}^{cpu} = \left(\sum_{s' \in S} \hat{C}_{s'}^{cpu} \right) \times \frac{\bar{C}_{s,w}^{cpu} + \hat{U}_{s,w}^{cpu} \times C_s^{disk}}{\sum_{s' \in S, w' \in W} \bar{C}_{s',w'}^{cpu} + \hat{U}_{s',w'}^{cpu} \times C_s^{cpu}} \quad (14)$$

$$\check{C}_{s,w}^{cpu} = \left(\sum_{s' \in S} \check{C}_{s'}^{cpu} \right) \times \frac{\hat{C}_{s,w}^{cpu}}{\sum_{s' \in S, w' \in W} \hat{C}_{s',w'}^{cpu}} \quad (15)$$

$$\vec{C}_{s,w}^{cpu} = \bar{C}_{s,w}^{cpu} + \hat{C}_{s,w}^{cpu} + \check{C}_{s,w}^{cpu} \quad (16)$$

Cost of Disks $\vec{C}_{s,w}^{disk}$***:***

$$\bar{C}_{s,w}^{disk} = \left(\sum_{s' \in S} \bar{C}_{s'}^{disk} \right) \times \frac{\bar{U}_{s,w}^{disk}}{\sum_{s' \in S, w' \in W} \bar{U}_{s',w'}^{disk}} \quad (17)$$

$$\hat{C}_{s,w}^{disk} = \left(\sum_{s' \in S} \hat{C}_{s'}^{disk} \right) \times \frac{\bar{C}_{s,w}^{disk} + \hat{U}_{s,w}^{disk} \times C_s^{disk}}{\sum_{s' \in S, w' \in W} \bar{C}_{s',w'}^{disk} + \hat{U}_{s',w'}^{disk} \times C_s^{disk}} \quad (18)$$

$$\check{C}^{disk}_{s,w} = \left(\sum_{s'\in S}\check{C}^{disk}_{s'}\right) \times \frac{\bar{C}^{disk}_{s,w}}{\sum_{s'\in S, w'\in W}\hat{C}^{disk}_{s',w'}} \quad (19)$$

$$\vec{C}^{disk}_{s,w} = \bar{C}^{disk}_{s,w} + \hat{C}^{disk}_{s,w} + \check{C}^{disk}_{s,w} \quad (20)$$

The total cost of RO-BURST model*(Model 3):*

$$\vec{C}_w = \sum_{s'\in S}\vec{C}^{cpu}_{s',w} \times \alpha + \sum_{s'\in S}\vec{C}^{disk}_{s',w} \times \beta \quad (21)$$

D. Pricing Strategy

Apart from employing RO-BURST model to calculate the infrastructure costs of cloud service, we also provide an applicable pricing strategy based on our RO-BURST model. There are two advantages of RO-BURST for the IaaS's pricing: firstly, its robustness helps to stabilize cost estimates and make the pricing neither far beyond nor much lower than actual cost. Secondly, a price based on RO-BURST model owns more competitiveness for IaaS providers with its dynamic adjustment to the market trend. The equation for pricing strategy is introduced as below (θ represents the number of servers hosted by a workload w, S means the total number of servers in the pool ($1 \leq \theta \leq$ S), δ represents profit index and we set it 120% here).

$$Price = \vec{C}_w \times S/\theta \times \delta \quad (22)$$

where θ represents the number of servers hosted by a workload w, S means the total number of servers in the pool ($1 \leq \theta \leq S$), δ represents profit index and we set it 120%

In the next section, we will evaluate these three models under a variety of workloads in details.

IV. Model characterization and Pricing strategy

To testify whether the three proposed models are efficient, robust and feasible enough, we collected 20 workloads from one IT company's shared data center, each of which is assigned to different servers in real IT application environments, including ERP, OA, Code Tracking, Web Service, et al.

These workloads have been traced for 20 days and recorded every 5 minutes, so we have 5,760 records for each workload, and each trace describes a workload's resource usage. We conduct a comprehensive study to compare the competiveness, stability and robustness in term of cost estimates of these three models under different workload placement scenarios. Unless particularly specified, C^{cpu}_s and C^{disk}_s are set at \$100, and α and β are 100%.

A. Workload Characteristics

Figure 3 and 4 show the relationship between the peak and average demand in CPU and disks of 20 workloads respectively. We summarize three characteristics about workload as follows:

- If we define a workload as one with bursty demand when the peak-average ratio is higher than 5, for 100% of these workloads, they share such feature in disk usage. For example, the ratio of disk could range from hundreds to tens and has a mean value of 110.27. As for CPU, 65% of the workloads show burstiness and the average ratio amounts to 10.93.
- For those workloads with high disk demands, they may not have high usage in CPU (e.g., workload 2, 5, 19). However, we also find that workload 14 and 15 have both high CPU and disk requirements, so there is no apparent relevance between disks and CPU demands.
- For 20% of these workloads, both peak and average disk demands are much more intensive than others, and they conduct burstiness more easily

B. Influence of Burstiness on Cost Estimates

Without loss of generality, this paper considers an experiment setting with 5 servers in a shared resource pool and 3 workload placement scenarios.

- **Scenario I** is an equal distributed strategy. It assigns 20 workloads to 5 servers, and ensures that a server consistently keeps hosting 4 workloads and the peak demands less than the capacity of each server.
- **Scenario II** is a greedy distributed strategy. A second server could not be assigned a workload until the first server is used up.
- **Scenario III** is a balanced strategy, employing 4 servers to host 20 workloads, with each server assigned 5 workloads.

Due to space limitation, this paper only presents the detailed research results of two workloads, named as ***C*** and ***D***, in this paper. The results of the other traces are very similar to the ones presented. Figure 5 shows their CPU demands for the cost interval(20 days).

For workload ***C*** and ***D***, they have almost the same average CPU usage, i.e., 0.2625% and 0.2569%. However, their peak CPU demands, vary significantly, with 4.00% and 1.15% respectively. Apparently, workload ***C*** is much more bursty than workload ***D***. For most of workloads, such performance results in complication of workload placement, so workload ***C*** calls need for more servers and the charge of that should be much higher. Table III presents CPU cost estimates of workload ***C*** and ***D*** calculated by three proposed models under scenario III. The close results of Average-based model fail to reflect workload's burstiness, whereas Bursty-based and RO-BURST models noticeably show the difference between two workloads and objectively incorporate the impact of burstiness and unallocated utilization.

TABLE III
Cost of Workload ***C*** and ***D*** in Scenario III

Workload	Model 1	Model 2	Model 3
Workload ***C***	\$33.5	\$37.5	\$26.2
Workload ***D***	\$32.7	\$11.0	\$7.7

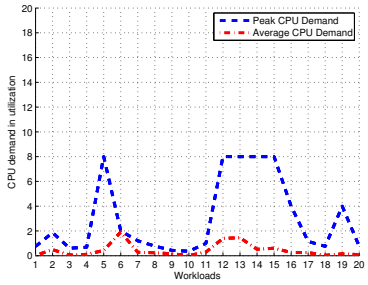

Fig. 3. The peak and average CPU demands of 20 workloads

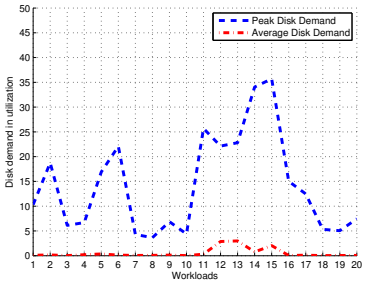

Fig. 4. The peak and average disk demands of 20 workloads

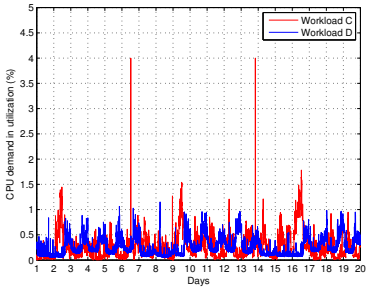

Fig. 5. CPU demands under workload C and D

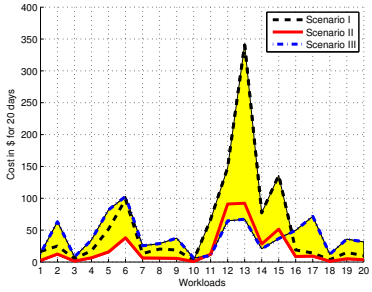

Fig. 6. Costs of 20 workloads calculated by the **Average-based model**

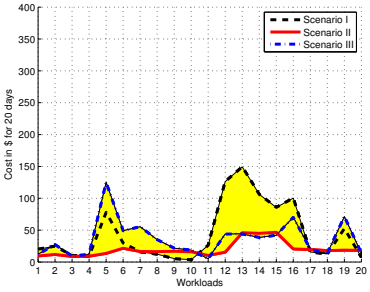

Fig. 7. Costs of 20 workloads calculated by the **Bursty-based model**

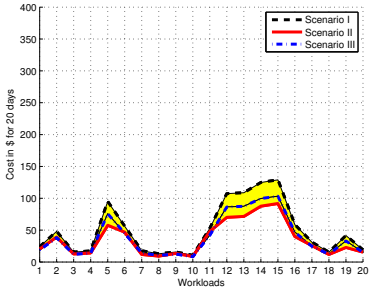

Fig. 8. Costs of 20 workloads calculated by the **RO-BURST model**

C. Robustness in Proposed Models

Figure [6, 7 and 8] show the costs of these 20 workloads reported by these three models (The cost range was shown in yellow). It is observed that the cost range of 20 workloads under Average-based and Bursty-based models are considerably wider than RO-BURST.

For model 1 and 2, the maximum costs exceed minimum cost by 90% and 53%, while the percentile of RO-BURST only reaches 20%, as shown in Figure 9. In other words,Average-based and Bursty-based model are much more sensitive to workload placement and thus lack robustness. In contrast, RO-BURST is a more reliable, predictable, and robust cost estimate model with less sensitivity to placement decisions.

D. Cost Volatility Factors

Due to numerous factors contributing to hardware and power (e.g, CPU, disks, Memory and network) costs, the fluctuation in their price could be frequent. Hence, we import α and β as volatility factors in order to appropriately temper our cost estimates to better press to market. Thailand takes up 60% shares of global hard disk production. The recent (Nov. 2011) floods in Thailand led to severe shortage of resources and thus resulted in huge rise to the acquisition price (e.g., most 500GB and 1TB disk prices have risen by an average of 80%). According to current market prices, the costs of a 1TB disk and a CPU with 4 x 2.8 MHz are $128 and $175 respectively, so we set α 1.75 and β 1.28 here. Their costs are presented in Figure 10. On the contrary, if real-time updates fail to be made, and we still use the volatility factors a month ago with α 1.75 and β 0.7, the results of cost estimates would be not objectively conducted, making IaaS providers suffer from a average loss of 23% (shown in Figure 11)

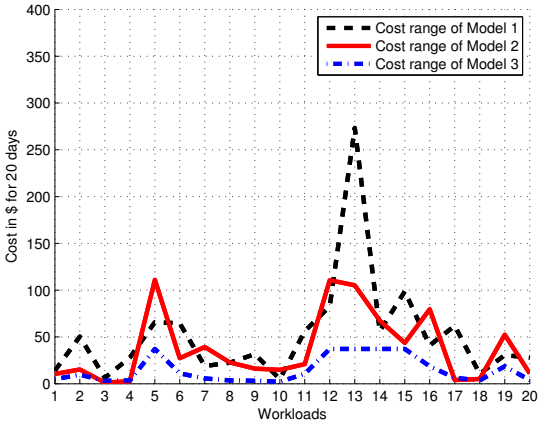

Fig. 9. The difference between maximum and minimum cost of three models, i.e. the cost range of each model

E. Pricing Strategy Characterizations

Figure 12 and 13 show the performance of 20 workloads calculated by Eq.(22) in section III in scenario I and scenario

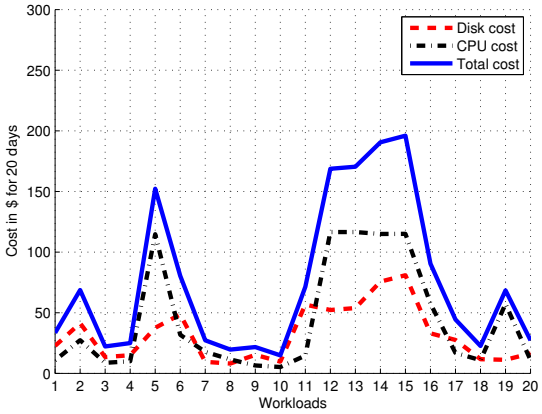

Fig. 10. CPU, disk and total cost of 20 workloads calculated by model 3 in scenario I when α = 1.75 and β = 1.28

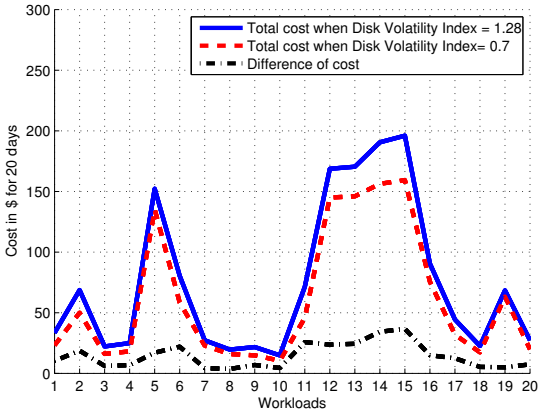

Fig. 11. CPU, disk and total cost of 20 workloads calculated by model 3 in scenario I when α = 1.75 and β = 0.7

III respectively (the introduction of scenario I and scenario III is presented in section 4). The prices that IaaS providers charge SaaS clients which are depicted by blue lines are almost the same between scenario I and scenario III. The costs of infrastructure that SaaS applications occupy, which are depicted by red line, however, are apparently dissimilar due to different workload placement. Scenario III, a better placement choice with fewer servers and less resources occupied save more money for IaaS providers. As a result, IaaS providers which employ a more favorable placement can earn an additional profit than those do not.

V. Related Work

Recently, there are many researches on the cost of applications over the cloud by discussing and evaluating workload consolidation in virtual environment. Ref. [8] provides a general approach, which is able to estimate the compensation (i.e., additional resource requirement) incurred by an application's transition from real hardware to a virtual machine. To achieve service level objectives, a dynamic AutoControl system that involves an online model estimator and multi-input, multi-output (MIMO) resource controller is raised in Ref. [9].

In Ref. [10], it comes up with an approach to determine the beneficial time frame for VM reassignment, aiming to save energy consumption to maximize the possible profit by reducing the complexity of resource and workloads management in the data center.

In addition, it is also important to determine what kind of application placement should be used for enhancing reliability and efficiency of cloud computing system, which necessarily brings a reduction in cost for IaaS providers. By comparing centralized and distributed decision making, Ref. [11] presents a distributed capacity agent manager to make service providers process and assign their resource more quickly and accurately.

Concerning workload characterization in the cloud, Ref. [12] uses data trace in the real data center to realize the prediction of workloads hosted on servers by offering a prediction model. The focus on their work lies in workload characterization based on each server and aims to predict workload changes for co-clusters. Our paper, however, presents workload characteristics based on workload itself in section IV, and mainly concentrates on the generality of a workload's burst attribute and its considerable influence on cost estimates.

Amazon.com Inc.'s Elastics Compute Cloud, or EC2, charges clients by fixed-size virtual machines requested per hour. They omit the variation among different workloads, which could generate a waste of fixed machine size and thus result in more infrastructure cost for IaaS providers. Another Usage-based pricing strategy based on clients' actual usage, which is applied by many companies, such as Cisco [1]; also fail to take into account the burst attribute of workload. However, our RO-BURST is a more robust cost estimate model compared with Usage-based charge model, for ours less sensitivity to workload placement and burst consideration.

Closest in spirit to our work is HP lab's [13]. Their work also put forward a model with robustness, while our RO-BURST not only considers robustness, but also involves iteration thus making our estimates more reliable. In addition, our model has extensibility for requirement of Disk, CPU, memory and etc. Moreover, RO-BUST adjusts itself to the market trend by importing volatility factors, based on which the IaaS providers could offer competitive pricing strategy for their SaaS clients.

VI. Conclusion and Future Work

We firstly raise the issue that an important task for IaaS providers is to seek a refined cost model to estimate the cost of the infrastructure consumed by SaaS applications. Then we present three different and somewhat connected cost models. We experiment 20 workloads in the shared pool and consider the impact made by bursty and unallocated resources, aiming to find a robust model with less sensitivity to consolidation placement. Features of workload performance and cost estimates calculated by each model are elaborated and com-

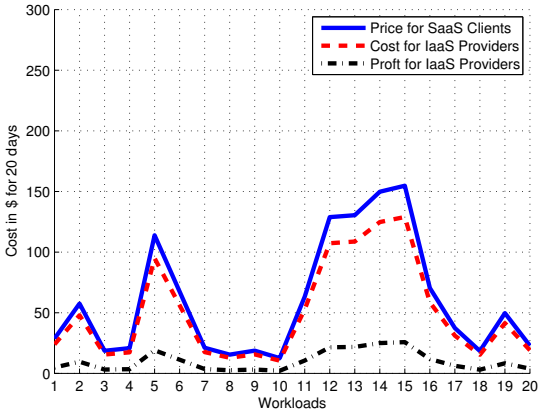

Fig. 12. The charge and cost for 20 workloads under Scenario I

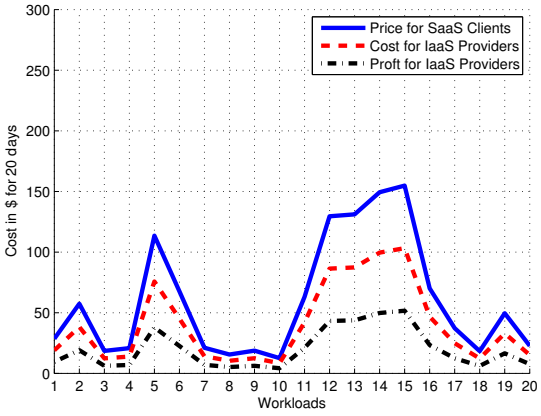

Fig. 13. The charge and cost for 20 workloads under Scenario III

parisons are made in this paper. Eventually, based on shared resources pool rather than resources within a server, we propose a RO-BURST model that enjoys properties of reliability and generality. To take into considerations the market impacts, we also import volatility factors of key hardware components, such as disks and CPU, into our model. At last, we offer IaaS providers a pricing strategy to better charge the SaaS for the infrastructure rent.

Complexity and multiformity are major issues for thousands of applications hosted in data centers. Our introduced model, RO-BURST, based on considerations of burstiness and shared resource pool, aims to help IaaS providers to evaluate the costs of numerous kinds of workloads with strong robustness. RO-BURST is a model that congregates robustness - less sensitive to the workload placement, and scalability - can be applied to all kinds of hardware resource demands, such as disks, CPU, memory, and network. In addition, based on our model, IaaS providers are able to set up reliable charges for their clients with the imported volatility factors, which take into considerations the volatility of market prices of hardware components. We also demonstrate that, the better workload placement scenarios our RO-BUST system employs, the more cost-effective and profitable the IaaS and SaaS will achieve by using the pricing strategy proposed in this work.

Our future work includes: improving our RO-BURST model, making the volatility factors α,β automatically updated rather than manually regulated; extending our model to other realms, i.e., telecommunication, power plants and hydro-electric stations, etc.; finding a better expression of burstiness in workload instead of difference between peak and average value. We also seek to propose a more sophisticated and refined pricing strategy based on our RO-BURST model to help IaaS providers and SaaS clients achieve win-win.

ACKNOWLEDGEMENT

We thank the anonymous reviewers of CCGRID for their feedback on previous versions of this paper. This Project supported by the National Basic Research Program (973) of China (No. 2011CB302303), the National Natural Science Foundation of China (No. 60933002), the Natural Science Foundation of Hubei province (NO. 2010CDB01605), the HUST Fund under Grant (Nos.2011QN053 and 2011QN032), the Fundamental Research Funds for the Central Universities and National Science Foundation (CNS #1117032, EAR 1027809, IIS #091663, CCF #0937988, CCF #0737583, CCF #0621493).

REFERENCES

[1] Cisco white paper, Managing the Real Cost of On-Demand Enterprise Cloud Services with Chargeback Models.

[2] HP Capacity Advisor Version 4.0 User's Guide. *http://h10032.www1.hp.com/ctg/Manual/c02015766.pdf*

[3] J. H. Holland, Adaptation in Natural and Artificial Systems. *Ann Arbor: University of Michigan Press*, 1975.

[4] Amazon web services. *http://aws.amazon.com/*

[5] Jitendra Kumar, Vernon, and Sandeep Kumar, The Economic Perspective of Cloud Computing. *UACEE International Journal of Advances in Computer Networks and Security*,Page 145-152.

[6] D. Gmach, J. Rolia, and L.Cherkasova, Resource and Virtualization Costs up in the Cloud: Models and Design Choices. *Proc. of the International Conference on Dependable Systems and Networks, (DSN'20011)*,Hong Kong, China, June 27-30, 2011.

[7] L.Cherkasova and J. Rolia, R-Opus: A Composite Framework for Application Performability and QoS in Shared Resource Pools. in Proc. of the Int. Conf. on DependableSystems and Networks (DSN),Philadelphia, USA, 2006.

[8] Timothy Wood, Ludmila Cherkasova, Kivanc Ozonat, and Prashant Shenoy, Predicting Application Resource Requirements in Virtual Environments. *HP Laboratories, Technical Report*.

[9] Pradeep Padala, Kai-Yuan Hou,Xiaoyun Zhu, and Mustafa Uysal, Automated Control of Multiple Virtualized Resources. *EuroSys 2009*,Nuremberg, Germany, April 1-3.

[10] Thomas Setzer and Alexander Stage, Decision support for virtual machine reassignments in enterprise data centers. *IEEE/IFIP Network Operations and Management Symposium, NOMS* ,2008,7-11 April, Salvador, Bahia, Brazil.

[11] Trieu C. Chieu and Hoi Chan, Dynamic Resource Allocation via Distributed Decisions in Cloud Environment. *IEEE International Conference on e-Business Engineering, ICEBE 2009*, Macau, China, 21-23 October.

[12] Arijit Khan, Xifeng Yan,Shu Tao, and Nikos Anerousis, Workload Characterization and Prediction in the Cloud: A Multiple Time Series Approach. *Technical report www.cs.ucsb.edu/arijitkhan/Papers*.

[13] D. Gmach, J. Rolia, and L. Cherkasova, Chargeback Model for Resource Pools in the Cloud. *Proc. of the 11th IFIP/IEEE Symposium on Integrated Management (IM'2011)*,Dublin, Ireland, May 23-27, 2011.