

# Distributed Resource Allocation for Data Center Networks: A Hierarchical Game Approach

Huaqing Zhang\*, Yong Xiao\*, Shengrong Bu<sup>†</sup>, Richard Yu<sup>‡</sup>, Dusit Niyato<sup>§</sup>, and Zhu Han\*  
 \*Department of Electrical and Computer Engineering, University of Houston, Houston, TX, USA.  
<sup>†</sup>School of Engineering, University of Glasgow, UK  
<sup>‡</sup>Department of Systems and Computer Engineering, Carleton University, Ottawa, ON, Canada  
<sup>§</sup>School of Computer Engineering, Nanyang Technological University, Singapore

**Abstract**—The increasing demand of data computing and storage for cloud-based services motivates the development and deployment of large-scale data centers. This paper studies the resource allocation problem for the data center networking system when multiple data center operators (DCOs) simultaneously serve multiple service subscribers (SSs). We formulate a hierarchical game to analyze this system where the DCOs and the SSs are regarded as the leaders and followers, respectively. In the proposed game, each SS selects its serving DCO with preferred price and purchases the optimal amount of resources for the SS's computing requirements. Based on the responses of the SSs' and the other DCOs', the DCOs decide their resource prices so as to receive the highest profit. When the coordination among DCOs is weak, we consider all DCOs are noncooperative with each other, and propose a sub-gradient algorithm for the DCOs to approach a sub-optimal solution of the game. When all DCOs are sufficiently coordinated, we formulate a coalition game among all DCOs and apply Kalai-Smorodinsky bargaining as a resource division approach to achieve high utilities. Both solutions constitute the Stackelberg Equilibrium. The simulation results verify the performance improvement provided by our proposed approaches.

**Index Terms** — Data center, hierarchical game, game theory, resource management.

## I. INTRODUCTION

Cloud computing attracts significantly interest in recent years due to the popularity of data services and applications. To meet the demand for the data-intensive cloud computing applications, huge investments have been made by companies, commonly referred to as the data center operators (DCOs) to build large-scale data centers [1]. Each data center consists of a large amount of facilities and computing resources, which can be accessed and shared by multiple service subscribers (SSs) at the same time. For example, companies such as eBay, Facebook, Google, Microsoft, and Yahoo [2]–[7] nowadays are able to provide data computing or data storage services for SSs with large-scale data centers, which significantly improve the SSs' efficiency and convenience.

As more and more companies join the cloud computing market, the competition among different DCOs becomes intense. How to optimize their services and prices so as to attract more SSs and maximize the revenues is one critical problem for DCOs. From the SS's perspective, different SSs have different expectations in price, delay, etc., of various cloud computing services provided by the DCOs. Therefore, how to choose

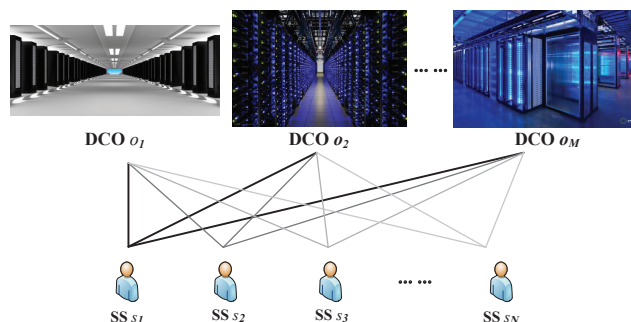


Fig. 1: The game structure

the optimal DCO that meets its requirements is an important problem for each SS.

Nevertheless, the existing works didn't consider any relationship among multiple DCOs, which motivates the work of this paper. In this paper, we investigate resource allocation among multiple competitive or cooperative DCOs, each of which processes resources that can be accessed by SSs at a certain price. The main objective of each DCO is to maximize its profit by adjusting the price offered to the SSs. Since each SS has its own service requirement and price affordability, the SS should decide the appropriate DCO and the number of resource blocks to procure based on the offered price. Therefore, we formulate a hierarchical game model to analyze the joint optimization of the decision making processes for both DCOs and SSs. In this game, all DCOs are the leaders that decide the prices first, and all SSs are the followers that can make their decisions based on the prices declared by the leaders. When the coordination among all DCOs is weak, we consider all DCOs to be noncooperative with each other and propose the sub-gradient algorithm for the DCOs to approach a sub-optimal solution of the game. When all DCOs are sufficiently coordinated with each other, we formulate a coalition game among all the DCOs. In order to guarantee fairness and avoid the competition among all DCOs in the coalition game, each DCO should obtain revenue proportional to its capability. Accordingly, Kalai-Smorodinsky bargaining is adopted as a resource division approach to achieve fair and efficient utilities. Based on the above, the contribution of this paper can be summarized as follows,

- A multi-DCO multi-user scenario with cooperative and competitive behaviors among DCOs is analysed in the paper.
- Based on the proposed scenario, a hierarchical game-based model has been established to analyze the interaction between the DCOs and the SSs. In the hierarchical game, the interaction between the DCOs and SSs are modeled as a Stackelberg game, and the cooperation and competition behaviors among the DCOs are modeled as a coalition game and non-cooperative game, respectively. To our best knowledge, it is the first paper to adopt a hierarchical game model in the data center networks.
- In the scenario where all DCOs are competitive with each other, a sub-gradient algorithm is adopted to reach a Stackelberg equilibrium solution where no DCO or SS can further improve their performance by unilaterally deviating from their decisions.
- In the scenario where all DCOs are coordinated with each other, Kalai-Smorodinsky bargaining is applied to achieve fair sharing of their utility among all DCOs.
- Simulation results have been presented to verify the performance improvements of our proposed approaches.

The rest of this paper is organized as follows. We describe the system model in Subsection II-A and formulate the problems in Subsection II-B. According to the formulated problem, we further analyze the game in Section III and present simulation results in Section IV. Finally, we show related works in Section V and summarize our work in Section VI.

## II. SYSTEM MODEL AND PROBLEM FORMULATION

### A. System Model

We consider a cloud computing system consisting of  $M$  DCOs, labeled as  $o_1, o_2, \dots, o_M$  with different amounts of computing facilities that can be accessed to and shared by  $N$  SSs, labeled as  $s_1, s_2, \dots, s_N$ . The computing facilities can be massive data centers built by each DCO or any public data centers rent and shared by DCOs. We suppose that all the DCOs offer computing resources over the Internet [34] to all SSs at the same time. We use the term “computing resource block” to denote the unit of time and computing speed measured by service rate that can be allocated to each SS. Let  $S_i$  be the total number of computing resource blocks allocated by DCO  $o_i$ . We use  $\mu_i$  to denote the service rate of each computing resource block of DCO  $o_i$ . Let  $\lambda_j$  be the workload arrival rate of  $s_j$ . Let  $\lambda_{ij}$  be the workload arrival rate generated from SS  $s_j$ , which will be processed using the resources of DCO  $o_i$ . We have  $\lambda_j = \sum_{i=1}^M \lambda_{ij}$ .

In this paper, we focus on the delay-sensitive applications in which quality of service (QoS) of each SS is relative with both the data rate and the total delay during the service.  $\forall i \in \{1, 2, \dots, M\}$ , DCO  $o_i$  charges price  $p_i$  to each SS for using one computing resource block. The main objective of each SS is to choose the DCO that can provide the highest QoS at the lowest price. Specifically, the cost incurred by the queuing delay for each SS  $s_j$  when served by DCO  $o_i$  is given by [35], [36],

TABLE I: List of Notations

Symbol	Definition
$M$	Total number of DCOs
$N$	Total number of SSs
$S_i$	Number of computing resource blocks for the DCO $o_i$
$\mu_i$	Service rate of computing resource blocks in the DCO $o_i$
$\lambda_j$	Workload arrival rate for the SS $s_j$
$p_i$	Price per unit of computing resource block of the DCO $o_i$
$p_i^{max}$	Maximum service price of the DCO $o_i$
$w_{ij}$	Cost of queuing delay of SS $s_j$ at DCO $o_i$
$d_j^{z_1 z_2 \dots z_N}$	Cost of network delay of SS $s_j$ when each SS $s_j$ is served by DCO $o_{z_k}$
$r_{ij}$	Cost of total delay of SS $s_j$ at DCO $o_i$
$m_{ij}$	Number of computing resource blocks from DCO $o_i$ to SS $s_j$
$e_i$	Energy cost of DCO $o_i$
$k_j$	Weight factor between the benefits from the workload and the total cost of SS $s_j$
$c$	Cost of purchasing one watt power
$q_i$	Power requirement of each computing resource block in DCO $o_i$
$\alpha_{ij}$	Probability for SS $s_j$ to choose DCO $o_i$
$b_{z_j j}$	Motivation of price reduction on one unit of network delay for the SS $s_j$
$u_i$	Utility of DCO $o_i$
$u_i^{max}$	Maximum utility of DCO $o_i$
$v_j$	Utility of SS $s_j$
$r_{th}$	Upper bound of total delay cost
$q_{ij,s}$	Static power requirement when DCO $o_i$ serves SS $s_j$
$q_{ij,c}$	Computing power requirement when DCO $o_i$ serves SS $s_j$
$x_{ij}$	Speed of computing workloads when DCO $o_i$ serves SS $s_j$
$\alpha$	Pairing outcomes between DCOs and SSs
$\mathbf{m}$	Strategies of all SSs
$\mathbf{p}$	Prices set by all DCOs

$$w_{ij} = \frac{\lambda_j}{\mu_i - \frac{\lambda_j}{m_{ij}}}, \quad (1)$$

where  $m_{ij}$  is the number of computing resource blocks offered by DCO  $o_i$  to SS  $s_j$ . This work can be easily extended to other delay models in DCOs.

The cost of total delay can be expressed as the summation of the delay in data processing by the DCOs plus the delay in data transmission in the network, i.e.,

$$r_{ij} = w_{ij} + d_j^{z_1 z_2 \dots z_N}, \quad (2)$$

where  $d_j^{z_1 z_2 \dots z_N}$  is the cost of network delay for SS  $s_j$  when each DCO  $o_{z_k}$  serves each SS  $s_k$ ,  $\forall k \in \{1, 2, \dots, N\}$ . The network delay can be affected by the time spent on uploading the computing data from SS  $s_j$  to DCO  $o_i$ , as well as that spent on a feedback of the computing results from the DCOs to the SSs. As we can obtain the value of the network delay by sending training sequence timely, in this paper, we assume the value of  $d_j^{z_1 z_2 \dots z_N}$ ,  $\forall j \in \{1, 2, \dots, N\}$  is known.

Furthermore, we consider resource allocation for each data center where each DCO needs to take into account its power consumption when providing services to SSs. Specifically, we consider the energy cost of each DCO as the total amount of

power required by all the computing resource blocks as follows [37]:

$$e_i = c \sum_{j=1}^N \alpha_{ij} \beta_{ij} m_{ij}, \quad (3)$$

where  $\beta_{ij}$  is the energy consumption for each computing resource block, satisfying,

$$\beta_{ij} = \begin{cases} q_{ij,s} + q_{ij,c}(x_{ij}) \frac{\lambda_i}{x_{ij}}, & \text{if } x_{ij} > 0, \\ 0, & \text{if } x_{ij} = 0. \end{cases} \quad (4)$$

$q_{ij,s}$  is the static power requirement regardless of workloads as long as the corresponding computing resource block  $m_{ij}$  has been used by an SS.  $q_{ij,c}(x_{ij})$  is the computing power when the computing resource block  $m_{ij}$  has been applied to compute workloads  $x_{ij}$ .  $c$  is the cost of purchasing each unit of power, and  $\alpha_{ij}$  is the probability for SS  $s_j$  to choose DCO  $o_i$ , where  $\alpha_{ij} = 1$  (or  $\alpha_{ij} = 0$ ) means that SS  $s_j$  is (or is not) served by DCO  $o_i$ .

Accordingly, the profit of DCO  $o_i$ ,  $\forall i \in \{1, 2, \dots, M\}$ , is the total revenue obtained by selling resources to SSs minus the cost of power consumption, which can be written as

$$u_i(p_i | \boldsymbol{\alpha}, \mathbf{m}, \mathbf{p}_{-i}) = \sum_{j=1}^N \alpha_{ij} m_{ij} p_i - e_i, \quad \forall i \in \{1, 2, \dots, M\}, \quad (5)$$

where  $\boldsymbol{\alpha} = \langle \alpha_{ij} \rangle$ ,  $i \in \{1, 2, \dots, M\}$ ,  $j \in \{1, 2, \dots, N\}$  is the pairing probability between DCOs and SSs,  $\mathbf{m} = \langle m_{ij} \rangle$ ,  $i \in \{1, 2, \dots, M\}$ ,  $j \in \{1, 2, \dots, N\}$  contains the strategies of all SSs,  $\mathbf{p}_{-i}$  contains the optimal prices set by all other DCOs except DCO  $o_i$ , and  $\mathbf{p} = \langle p_i \rangle$ ,  $i \in \{1, 2, \dots, M\}$  contains the prices charged by the DCOs.  $p_i \geq 0$ ,  $\forall i$ .

Each SS tries to achieve a high utility from the service while minimizing both the service delay and payment to DCOs. We hence can define the utility of SS  $s_j$  as

$$v_j(\mathbf{m}_j | \mathbf{m}_{-j}, \mathbf{p}) = k_j \lambda_j - \sum_{i=1}^M \alpha_{ij} m_{ij} p_i - \sum_{i=1}^M \alpha_{ij} r_{ij}, \quad \forall j \in \{1, 2, \dots, N\}, \quad (6)$$

where  $\mathbf{m}_j$  is the strategy of purchasing computing resources from all DCOs for SS  $s_j$ ,  $\mathbf{m}_{-j}$  is the strategy of purchasing computing resources for all the other SSs, and  $k_j$  is the weight factor. Accordingly, the first term  $k_j \lambda_j$  represents the weighted revenues achieved by each SS  $s_j$  from the services.

The second term  $\sum_{i=1}^M \alpha_{ij} m_{ij} p_i$  is the amount of money that SS  $s_j$  is required to pay the DCO  $o_i$  for the service. The third term  $\sum_{i=1}^M \alpha_{ij} r_{ij}$  is the cost of the total delay in service. If the delay is large, the user experience of the SSs is poor, and the corresponding utilities are small. Let  $r_{th}$  be the maximum delay that can be tolerated by SSs. Thus,  $r_{ij} \geq 0$  and  $r_{ij} \leq r_{th}$ .

## B. Problem Formulation

In the cloud computing with multiple DCOs and multiple SSs, when each DCO sets its price for each computing resource block, the DCO needs to consider the prices offered by other DCOs as well as the strategies of all SSs. Therefore, the optimization problem for each DCO  $o_i$  is,

$$\begin{aligned} \max_{p_i} \quad & u_i(p_i | \boldsymbol{\alpha}, \mathbf{m}^*, \mathbf{p}_{-i}^*), \quad \forall i \in \{1, 2, \dots, M\}, \\ \text{s.t.} \quad & p_i \geq 0, \end{aligned} \quad (7)$$

where  $\mathbf{p}_{-i}^*$  is the optimal prices of all other DCOs except the DCO  $o_i$ .  $\mathbf{m}^*$  is the optimal strategies of all SSs.

Based on the price declared by the DCOs, each SS needs to compete with other SSs when choosing their optimal DCO. Each SS also decides the optimal number of computing resource blocks procured from its chosen DCO. Therefore, we can define the optimal problem for SS  $s_j$  as

$$\begin{aligned} \max_{\mathbf{m}_j} \quad & v_j(\mathbf{m}_j | \mathbf{m}_{-j}^*, \mathbf{p}^*), \quad \forall j \in \{1, 2, \dots, N\}, \\ \text{s.t.} \quad & \begin{cases} r_{ij} \geq 0, \\ r_{ij} \leq r_{th}, \end{cases} \end{aligned} \quad (8)$$

We assume that each of the DCOs or SSs is rational and autonomous when making its decision in a distributed fashion. To make full use of the resource provided by the DCOs and meet the computing requirements of all SSs, we model the scenario as a multi-leader multi-follower Stackelberg game. The DCOs act as the leaders, and the SSs act as the followers. In the following section, we will discuss and explore the optimal strategies for each player of the game, based on different settings and objectives.

## III. GAME ANALYSIS

According to the formulated problems in the modeled multi-leader multi-follower Stackelberg game, the optimal solutions can be achieved when the Stackelberg Equilibrium can be achieved between the DCOs and SSs. The concept of Stackelberg Equilibrium can be defined as follows.

**Definition 1.** [38] Let  $((\mathbf{X}, \mathbf{A}), (g, f))$  be the multi-leader multi-follower Stackelberg game with  $m$  leaders and  $n$  followers.  $\mathbf{X} = \mathbf{X}_1 \times \mathbf{X}_2 \times \dots \times \mathbf{X}_m$  and  $\mathbf{A} = \mathbf{A}_1 \times \mathbf{A}_2 \times \dots \times \mathbf{A}_n$  are the strategy profiles of leaders and followers, respectively.  $g = (g_1(\mathbf{x}), \dots, g_m(\mathbf{x}))$  is the payoff function of leaders for  $\mathbf{x} \in \mathbf{X}$ , and  $f = (f_1(\boldsymbol{\alpha}), \dots, f_n(\boldsymbol{\alpha}))$  is the payoff function of followers for  $\boldsymbol{\alpha} \in \mathbf{A}$ . Let  $\mathbf{x}_i$  be a strategy profile of leader  $i$ ,  $\mathbf{x}_{-i}$  be a strategy profile of all the leaders except leader  $i$ ,  $\boldsymbol{\alpha}_j$  be a strategy profile of follower  $j$ , and  $\boldsymbol{\alpha}_{-j}$  be a strategy profile of all the followers except leader  $j$ . A set of strategy profiles  $\mathbf{x}^* \in \mathbf{X}$  and  $\boldsymbol{\alpha}^* \in \mathbf{A}$  is the equilibrium of the multi-leader multi-follower game if  $\forall i, \forall j$ ,  $\mathbf{x}_i \in \mathbf{X}_i$ ,  $\boldsymbol{\alpha}_j \in \mathbf{A}_j$ ,

$$\begin{aligned} g_i(\mathbf{x}_i^*, \mathbf{x}_{-i}^*, \boldsymbol{\alpha}^*) &\geq g_i(\mathbf{x}_i, \mathbf{x}_{-i}^*, \boldsymbol{\alpha}^*) \geq g_i(\mathbf{x}_i, \mathbf{x}_{-i}, \boldsymbol{\alpha}^*), \\ f_j(\mathbf{x}, \boldsymbol{\alpha}_j^*, \boldsymbol{\alpha}_{-j}^*) &\geq f_j(\mathbf{x}, \boldsymbol{\alpha}_j, \boldsymbol{\alpha}_{-j}^*). \end{aligned}$$

In following parts of this section, we first consider a simplified version of our problem and analyze the Stackelberg

Equilibrium for a single-DCO and single-SS cloud computing system, in order to analyze the basic relationship between the DCO and SS. We then extend our model to the case with multiple DCOs and SSs, and discuss the competition or coalition among all DCOs.

#### A. Single-DCO single-SS Stackelberg game

Consider the scenario where there is only one DCO and one SS in the game, i.e.,  $M = 1$  and  $N = 1$ . Based on the cost of the SS, we have the following lemma.

**Lemma 1.** *In a single-DCO single-SS scenario, for a given price  $p_1$  of the DCO, the optimal number of computing resource blocks chosen by the SS is given by*

$$m_{11}^* = \frac{\lambda_1}{\mu_1 \sqrt{p_1}} + \frac{\lambda_1}{\mu_1}. \quad (9)$$

*Proof:* The proof is provided in Appendix A. ■

The optimal strategy for each SS has a closed form solution under the given price of the DCO, which can be pre-calculated and pre-stored in a table in the SS. In real scenarios, the SS can then use a simple table searching method to obtain its optimal solution with low implementation complexity. Therefore, for the SS in the game, the computation requirement is very low.

Furthermore, because of the first-move advantage in the Stackelberg game, the DCO is able to predict the optimal strategy of the SS. Therefore, we substitute (1) into (5), and the profit of the DCO can be derived as

$$u_1 = \frac{\lambda_1}{\mu_1} \sqrt{p_1} + \frac{\lambda_1}{\mu_1} p_1 - c\beta_{11} \frac{\lambda_1}{\mu_1 \sqrt{p_1}} - c\beta_{11} \frac{\lambda_1}{\mu_1}. \quad (10)$$

The utility of the DCO  $u_1$  is a monotonically increasing function with respect to  $p_1$ . Furthermore, according to the constraint that the delay of the SS cannot exceed  $r_{th}$ , i.e.,

$$r_{11} = \frac{\lambda_1}{\mu_1} \sqrt{p_1} + \frac{\lambda_1}{\mu_1} + \lambda_1 d_1^1 \leq r_{th}. \quad (11)$$

Thus in the single-DCO single-SS scenario, when the DCO sets price

$$p_1^{max} = (r_{th} - 1 - \mu_1 d_1^1)^2, \quad (12)$$

the profit of the DCO is maximized.

In the single-DCO single-SS scenario, the Stackelberg Equilibrium can be achieved. Both the DCO and the SS have their optimal utilities, respectively, and neither of them is able to change their strategy to achieve higher values.

#### B. Multi-DCO multi-SS hierarchical game

In a multi-DCO and multi-SS scenario, each SS has multiple choices on its serving DCOs, and each DCO may also be able to serve multiple SSs to receive higher profits. Accordingly, there exists the competition or coordination among DCOs and SSs. In this section, we analyze and propose strategies for each SS and DCO so as to receive the optimal utility.

We consider the data center network system with  $N$  SSs which can choose service from  $M$  DCOs. When SS  $s_j$  chooses DCO  $o_{z_j}$ ,  $\forall j \in \{1, 2, \dots, N\}$ , following the results in the single-DCO single-SS scenario, each SS makes its decision on the optimal number of computing resource blocks  $m_{z_j}^*$ , where

$$m_{z_j}^* = \frac{\lambda_j}{\mu_{z_j} \sqrt{p_{z_j}}} + \frac{\lambda_j}{\mu_{z_j}}. \quad (13)$$

Accordingly, the utility of SS  $s_j$  is denoted as  $v_j^{z_1 z_2 \dots z_N}$ , i.e.,

$$v_j^{z_1 z_2 \dots z_N} = k_j \lambda_j - \frac{\lambda_j}{\mu_{z_j}} (\sqrt{p_{z_j}} + 1)^2 - \lambda_j d_j^{z_1 z_2 \dots z_N}. \quad (14)$$

Based on the utility of each SS in different situations, we assume that each SS  $s_j$ ,  $\forall j \in \{1, 2, \dots, N\}$  is able to determine the probability to be served by each DCO  $o_{z_j}$ , i.e.,  $\alpha_{z_j j}$ ,  $\forall z_j \in \{1, 2, \dots, M\}$ . Therefore, in order to achieve high utilities with low service price and network delay for each SS  $s_j$ , we refer the incentive mechanism method in [39] and set the probability for each SS  $s_j$  to choose DCO  $o_{z_j}$  as

$$\alpha_{ij} = \frac{b_{z_j j}}{\sum_{z_j=1}^M b_{z_j j}}, \quad (15)$$

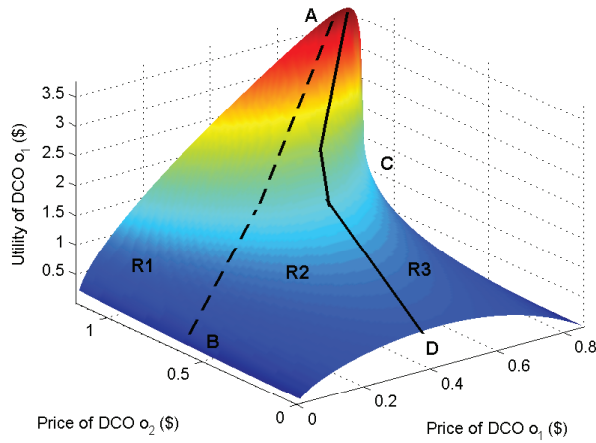
where  $b_{z_j j}$  is the motivation of price reduction on unit network delay, i.e.,

$$b_{z_j j} = \frac{(p_{z_j}^{max} - p_{z_j})}{d_j^{z_1 z_2 \dots z_N}}. \quad (16)$$

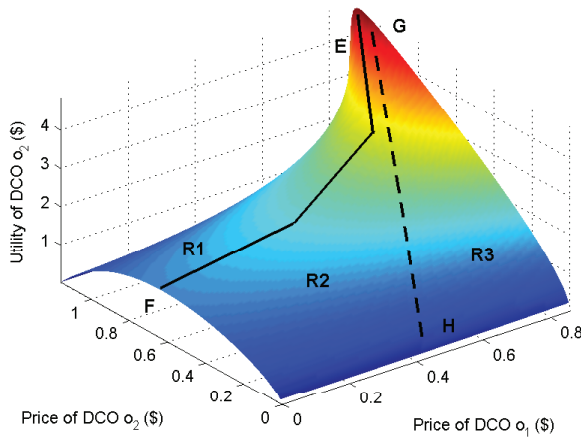
Following the results of the single-leader single-follower scenario, in order to satisfy the service delay requirements of all served SSs, the maximum setting price of DCO  $o_{z_j}$  is

$$p_{z_j}^{max} = \min\{(r_{th} - 1 - \mu_{z_j} d_j^{z_1 z_2 \dots z_N})^2\}, \quad \forall j \in \{1, 2, \dots, N\}. \quad (17)$$

Based on the above, if SS  $s_j$  experiences small network delay and relatively small service price compared with its maximum price constraint  $p_{z_j} \leq p_{z_j}^{max}$ , then the value of  $b_{z_j j}$  is small. When the value of  $b_{z_j j'}$  for other SS  $s_{j'}$  is relatively small, the probability for SS  $s_j$  to be served by the DCO  $o_{z_j}$  is large. Therefore, in order to attract more SSs, each DCO  $o_i$ ,  $\forall i \in \{1, 2, \dots, M\}$ , is motivated to set the price maximizing the gap  $p_i^{max} - p_i$  for each SS  $s_j$ ,  $\forall j \in \{1, 2, \dots, N\}$ . Compared with the behaviors of other DCOs  $o_{-i}$ , if the corresponding value of  $b_{ij}$  served by DCO  $o_i$  is relatively large, then the value of  $\alpha_{ij}$  is large, and thus SS  $s_j$  is more likely to be served by DCO  $o_i$ . On the other hand, each DCO needs to keep a high value of  $p_{z_j}$  to receive high revenues from the service. Accordingly, there is a tradeoff for setting prices of DCOs. Considering the behaviors of all other DCOs, if DCO  $o_i$  set a high service price for SS  $s_j$ , the revenues when SS  $s_j$  is served by DCO  $o_i$  is high, but the probability when SS  $s_j$  is served by DCO  $o_i$  is low. On the other hand, if the DCO set a low service price for SS  $s_j$ , SS  $s_j$  is more likely to choose the service of DCO  $o_i$ , but the revenues when SS  $s_j$  is served by DCO  $o_i$  are low.



(a) Utility of DCO  $o_1$  with different prices set by DCOs



(b) Utility of DCO  $o_2$  with different prices set by DCOs

Fig. 2: Utilities of the DCOs with different pricing behaviors

With the prediction of all SSs' behaviors, each DCO expects to set its service price so as to receive the optimal utility. Therefore, substitute (13) into (5) and the utility of DCO  $o_i$  can be denoted as

$$u_i = \sum_{j=1}^N \alpha_{ij} \left( \frac{\lambda_j}{\mu_i} \sqrt{p_i} + \frac{\lambda_j}{\mu_i} p_i - c\beta_{ij} \frac{\lambda_j}{\mu_i \sqrt{p_i}} - c\beta_{ij} \frac{\lambda_j}{\mu_i} \right). \quad (18)$$

In (18),  $\alpha_{ij}$  is related to the setting price of all DCOs. Accordingly, in order to obtain high utility, each DCO should also consider the behaviors of other DCOs. For ease of demonstration, we take an example of a two-DCO two-SS scenario and illustrate the relationships of setting prices from both DCOs in Fig. 2.

With different prices, the utilities of both DCOs are shown in Fig. 2. When the price of DCO  $o_2$  is fixed and DCO  $o_1$  increases its service price, the probability for the SSs to choose DCO  $o_1$  unilaterally decreases, but the utility of DCO  $o_1$  firstly increases and then decreases. Similarly, when the price of DCO  $o_1$  is fixed and DCO  $o_2$  increases its service price, the utility of DCO  $o_2$  firstly increases, then decreases.

In order to better analyze the problem, we separate the figures into three regions.

R1 In the region 1, the price of DCO  $o_2$  is relatively high, while the price of DCO  $o_1$  is relatively low. Because of the large gap in price between both DCOs, even though DCO  $o_2$  with a higher price is able to serve SSs with a lower delay and better performance, both SSs still prefer to choose DCO  $o_1$ , considering the total cost of the service. Accordingly, in order to serve the SSs and receive the highest revenues given strategies of the others, there are tradeoffs for both DCOs when they set their service prices. For DCO  $o_2$ , setting a high price allows it to receive high payment from serving the SSs. However, it should avoid setting a price far higher than that of DCO  $o_1$ . Otherwise, the probabilities for the SSs to choose DCO  $o_2$  are low, and DCO  $o_2$  cannot gain high revenues. For DCO  $o_1$ , setting low prices may help it serve more SSs, and receive more revenues. Nevertheless, as the service price is low, even though the amount of payment from SSs to DCO  $o_1$  is large, DCO  $o_1$  can only achieve limited revenue from a single SS. In general, the DCO may not receive high revenues.

R2 The price set by both DCOs at each point in this region is competitive. Accordingly, neither of the DCOs is able to be chosen by both SSs, and each DCO will serve one SS at the same time. In this situation, as the prices of both DCOs are competitive and each DCO needs to predict the behavior of the other DCO, and to set their serving prices optimally. On one hand, if the DCO is setting a price that is too high, its served SS will leave the service, resulting in a low revenue for the DCO. On the other hand, if the DCO sets a low price for its served SS, although the SS stays, the revenue of serving the SS is not maximized.

R3 In region 3, the price of DCO  $o_1$  is relatively high, while the price of DCO  $o_2$  is relatively low. With the same reason as in the case of region 1, even though DCO  $o_1$  with a higher price is able to serve the SSs with a lower delay and better performance, both SSs still prefer to choose DCO  $o_2$ , considering the total costs of the service. Therefore, on one hand, with prediction and estimation of the prices from DCO  $o_2$ , DCO  $o_1$  should avoid setting the price that is too high falling into the region 3. On the other hand, DCO  $o_2$  should compare and evaluate the total revenues, balancing the expected number of served SSs and setting a price.

Therefore, when setting the price for the SSs, there is the competition between DCO  $o_1$  and DCO  $o_2$ . Either DCO should estimate the behavior of the others, balance the setting price and number of the expected SSs and make decisions so as to achieve high revenues. In the general situation with  $M$  DCOs and  $N$  SSs, the figures become  $M + 1$  dimensions, which can be separated into  $M + 1$  regions. In each region, the tradeoff among DCOs' decisions works in a similar way.

However, for each DCO in the game, it is hard to predict the behaviors of other DCOs, causing unstable and unsatisfying

results for each other. In the following sections, based on the different benefits of the SSs and DCOs, we propose both the competitive and the coordinated strategies for the DCOs to achieve the highest utility given the actions of other DCOs. In the real situations where some DCOs coordinate with some other DCOs, and some still compete with other DCOs, the DCOs that can cooperate with each other will coordinated with each other first. Then, each coordinated DCOs can be regarded as a coalition and compete with other DCOs outside of the coalition using the competitive algorithm.

1) *The competitive behavior among DCOs:* In the noncooperative scenario where each DCO only considers its own revenue, in order to regulate the behaviors of the DCOs and obtain a stable and optimal solution, we propose a sub-gradient algorithm for DCOs of the game.

When DCOs adopt the sub-gradient algorithm, each DCO initially assumes that there is no competition with other DCOs and sets its service price at the maximum value to receive high utility. When the DCO discovers that there exist other DCOs trying to attract SSs with lower prices, the DCO predicts the reactions of the other DCOs on its own price and the tradeoff on its pricing strategy should be considered. If the DCO decreases its service price and competes with other DCOs, SSs are more likely to be served by the DCO. Thus, the expected number of served SSs increases. However, because of the low service price, the revenue the DCO obtain from each SS decreases. Similarly, when the DCO increases its service price, even though the DCO receives higher revenues from each served SS, as other DCOs serve SSs with lower prices and good performance, the number of SSs choosing the DCO decreases. Accordingly, in order to make optimal decisions based on the prediction other DCOs' strategies, we propose an iterated process for the DCOs to adjust their behaviors so as to receive optimal utilities. At each iteration, the DCO tries to increase or decrease its price with a small step and predict the resulting utility, if the adjustment increases its utility, the DCO will increase or decrease its price in the next iteration. Otherwise, the DCO will keep the current service price unchanged. Within finite number of iterations, all DCOs are able to determine the best decisions with the highest utilities. We assume that  $\mathbf{p} = \langle p_i \rangle$  is the pricing profile of all DCOs in the previous iteration,  $\mathbf{p}_{old} = \langle p_{old_i} \rangle$  is the pricing profile of all DCOs in the previous iteration, and  $\mathbf{p}_{old_{-i}}$  is the pricing profile of all DCOs except DCO  $o_i$  in the previous iteration. Accordingly, the detailed algorithm is shown as in Algorithm 1. Then, the following Lemma 2 holds for the proposed algorithm.

**Lemma 2.** *When the starting price and step size of DCO  $o_i$   $\Delta_i$ ,  $\forall i \in \{1, 2, \dots, M\}$  are fixed, the game can always converge to a unique outcome, which is also the Nash equilibrium of the game.*

*Proof:* The proof is provided in Appendix B. ■

For the proposed sub-gradient algorithm, it is not necessary for all the players to be strictly synchronized when they

---

**Algorithm 1** Strategy of each DCO in a multi-DCO multi-SS scenario.

---

```

1: Initially, each DCO ignores the existence of other DCOs,
   and sets its price at the maximum value to receive the
   highest revenue from all SSs.
2: while at least one DCO adjusts its price do
3:   for SS  $s_j$  do
4:     Based on the prices set by all DCOs, each SS
     determines the probability to be served by each DCO
     and purchases the optimal amount of resource blocks
     from each DCO.
5:   end for
6:   for DCO  $o_i$  do
7:     Each DCO stores the current value of service prices.
      $\mathbf{p}_{old} = \mathbf{p}$ ;
8:     Each DCO adjusts its price with a small step  $\Delta_i$  and
     calculate its own payoff based on the prediction of the
     followers' optimal strategies.
9:     if  $u_i(p_{old_i}, \mathbf{p}_{old_{-i}}) \leq u_i(p_{old_i} + \Delta_i, \mathbf{p}_{old_{-i}})$  then
10:       $p_i = \min\{p_i^{max}, p_{old_i} + \Delta_i\}$ ; % Increase the price
11:     else
12:       if  $u_i(p_{old_i}, \mathbf{p}_{old_{-i}}) \leq u_i(p_{old_i} - \Delta_i, \mathbf{p}_{old_{-i}})$  then
13:         $p_i = \max\{0, p_{old_i} - \Delta_i\}$ ; % Reduce the price
14:       else
15:         $p_i = p_{old_i}$ ; % Keep the price unchanged
16:       end if
17:     end if
18:   end for
19: end while

```

---

make decisions. More specifically, for each SS, based on the observation of the announced prices from all DCOs, each SS determines the optimal amount of computing resource block to purchase. In this case, all the SSs do not have to make decisions at the exactly same time. For each DCO, based on the observation of the announced prices of other DCOs and the observation of current behaviors of all SSs, it is able to follow our proposed algorithm to set its price to improve its utility. In other words, all the DCOs also do not need to make decisions at the same time, and the algorithm can still converge to the same Stackelberg equilibrium.

Based on the results of the iterated approaches, the computing resource blocks are allocated afterwards. Therefore, the utilities or revenues received by all DCOs and SSs follow the results of the iterated approaches. After one period, all the DCOs will perform the sub-gradient algorithm and do the resource allocation again. Therefore, for the iterative approach ahead of one period, the resulting Nash Equilibrium is still an equilibrium solution for a static game, not dynamic one.

2) *Coalition formation in DCOs:* Even though all DCOs are able to achieve Nash equilibrium outcomes with the proposed sub-gradient algorithm, the revenue of each DCO is still low due to the competition with all other DCOs. To improve their benefits, DCOs may seek to cooperate with each other and jointly decide the price offered to the SSs. Accordingly, in the upper layer of multi-leader multi-follower

hierarchical game, we propose the coalition game among all DCOs, in order to improve the utilities of the DCOs. However, some DCOs may be unwilling to cooperate with each other because of the unfair results within the coalition. We evaluate the capability of served SSs for each DCO as the revenue that the DCO receives when all the other DCOs stop serving SSs. Therefore, in order to guarantee fairness and avoid the competition among all DCOs in the coalition game, each DCO should set its price in the feasible region and obtain revenue proportional to its capability. In order to achieve the fair results and high revenues for all DCOs in the coalition game, we consider the Kalai-Smorodinsky bargaining as a resource division approach within all DCOs.

**Definition 2.** [41] [42] We denote  $\mathbf{U} = [u_1, u_2, \dots, u_M]^T$  as the feasible utility set, and let  $\mathbf{h} = [h_1, h_2, \dots, h_M]^T$  be the disagreement point set, which are the expectations of operators by joining the game without cooperation.  $\mathbf{Y}^* = F(\mathbf{U}, \mathbf{h})$  is regarded as a Kalai-Smorodinsky bargaining solution if the following six axioms, i.e., individual rationality, feasibility, pareto optimality, individual monotonicity, independence of linear transformations, and symmetry, are satisfied.

According to the conclusions of [43] and [44],  $u^* = [u_1^*, u_2^*, \dots, u_M^*]^T$  can be a unique solution satisfying all axioms in Definition 2, if the solution meets the following condition

$$\frac{u_1^* - h_1}{u_1^{\max} - h_1} = \frac{u_2^* - h_2}{u_2^{\max} - h_2} = \dots = \frac{u_M^* - h_M}{u_M^{\max} - h_M}, \quad (19)$$

where  $u_i^{\max}, \forall i \in \{1, 2, \dots, M\}$ , is the maximum utility of DCO  $o_i$  when all SSs are served by the DCO  $o_i$  with the highest price  $p_i^{\max}$ , i.e.,

$$u_i^{\max} = \sum_{j=1}^N u_{ij}^{\max}, \quad (20)$$

where

$$u_{ij}^{\max} = \left( \frac{\lambda_j}{\mu_i} \sqrt{p_i^{\max}} + \frac{\lambda_j}{\mu_i} p_i^{\max} - c\beta_{ij} \frac{\lambda_j}{\mu_i \sqrt{p_i^{\max}}} - c\beta_{ij} \frac{\lambda_j}{\mu_i} \right). \quad (21)$$

When all DCOs are noncooperative, the DCOs compete with each other by reducing their prices. Accordingly, we suppose that in the competition, the corresponding utility of each DCO  $o_i$  is zero, i.e.,  $h_i = 0, \forall i \in \{1, 2, \dots, M\}$  [42].

Based on the utilities of DCOs, we have the following lemma.

**Lemma 3.** For each DCO  $o_i$  in the coalition formation,  $\forall i \in \{1, 2, \dots, M\}$ , when all DCOs adopt Kalai-Smorodinsky bargaining as a resource division approach, the setting price of the DCO  $o_i$  satisfies

$$p_i^* = (p_i^{\max})^-, \quad (22)$$

where  $(\cdot)^-$  approaches the limit from the negative side. Correspondingly, the utility of each DCO  $o_i, \forall i \in \{1, 2, \dots, M\}$  is

$$u_i^* = \sum_{j=1}^N \frac{1}{N} u_{ij}^{\max}, \quad (23)$$

where

$$u_{ij}^{\max} = \frac{\lambda_j}{\mu_i} \sqrt{p_i^{\max}} + \frac{\lambda_j}{\mu_i} p_i^{\max} - c\beta_{ij} \frac{\lambda_j}{\mu_i \sqrt{p_i^{\max}}} - c\beta_{ij} \frac{\lambda_j}{\mu_i}. \quad (24)$$

*Proof:* The proof is provided in Appendix C. ■

#### IV. SIMULATION RESULTS AND DISCUSSIONS

In this section, we present the simulations to evaluate the performances of the proposed approaches with MATLAB. Without loss of generality, we follow the settings in [36] and assume that there are two DCOs accessible to three SSs at the same time. We set the service rate of each computing resource block in both DCOs as  $0.3 (ms)^{-1}$ . The workload arrival rates for SS  $s_1$ , SS  $s_2$  and SS  $s_3$  are  $0.7 (ms)^{-1}$ ,  $0.6 (ms)^{-1}$ , and  $0.6 (ms)^{-1}$ , respectively. We model the distance between source and data center, divided by the transmission speed of  $200km/ms$ , resulting in delay ranging in  $[0, 8]$  ms, and the delay tolerance of all SSs is set to 8 ms. The above settings are reasonable for the existing data center networks. [36] In the proposed sub-gradient method, we assume that the price step sizes for both DCOs are 0.01 dollar.

In Fig. 3, we compare our proposed sub-gradient algorithm with the ordinary noncooperative behavior where each DCO sets its optimal price based on the observation of other DCOs' behaviors in the previous iteration. The noncooperative behavior cannot guarantee the convergence of the game. In the simulation, we performed 1000 iterations and took the average of 10 latest results as an expected social welfare value. Moreover, we compare the proposed algorithm with the cooperative behavior, where each DCO can increase its served price regardless of the number of served SSs. Accordingly, in order to achieve the high utilities for all DCOs, each DCO sets its price as the maximum possible value. In our proposed Kalai-Smorodinsky bargaining method, both DCOs form a coalition with each other based on the bargaining strategy to achieve fair revenues. The social welfare achieved by the Kalai-Smorodinsky bargaining follows the results of cooperative behaviors. As shown in Fig. 3, with the increased delay tolerance  $r_{th}$  of SSs, all DCOs are able to set a high price to meet the service requirements of the SSs, and the performance of the SSs deteriorates. Therefore, the social welfare generally reduces with  $r_{th}$  increasing. Furthermore, at each value of  $r_{th}$ , the social welfare is always higher than the expected social welfare of the noncooperative strategy when the DCOs adopt the proposed sub-gradient algorithm. The social welfare when both DCOs are cooperative is the lowest, because when the DCOs cooperate with each other, even though the DCOs receive high revenues, all SSs suffer more because of the high service price and low quality of service. Based on the above results, from the perspective of SSs or the management of data center services, in order to achieve high social welfare, both DCOs should compete with

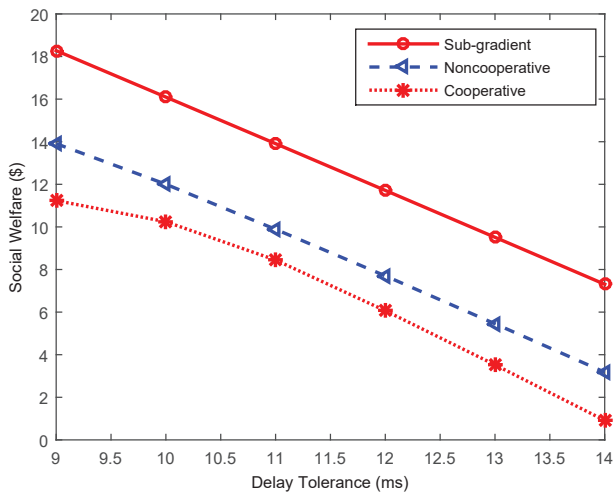


Fig. 3: Social welfare vs delay tolerance of SSs with different approaches

each other, and the proposed sub-gradient algorithm is a good tool to achieve stable and high revenues.

From the perspective of DCOs, the situation is different. In Fig. 4, we compare the utility of each DCO in coalition formation with Kalai-Smorodinsky bargaining, noncooperative scenario with sub-gradient algorithm, ordinary noncooperative behavior, and the situation where only DCO  $o_1$  or DCO  $o_2$  monopolizes the market. As shown in Fig. 4, we discover that when the DCOs form a coalition and adopt Kalai-Smorodinsky bargaining, the utilities of both DCOs are higher than the utilities when DCOs employ the proposed sub-gradient strategy or take the noncooperative behaviors. As both DCOs form a coalition, they do not need to compete with each other in order to achieve high revenues. They set their prices at high values to increase the total utilities for the DCOs so as to obtain high utilities through bargaining. The utilities of both DCOs when adopting the proposed sub-gradient strategy are also higher than the utilities in noncooperative behaviors. Furthermore, we discover in the single DCO situation, where all SSs have no choice on their serving DCOs, the DCO is able to receive higher utility than that when there are multiple DCOs. However, compared with the proposed Kalai-Smorodinsky bargaining strategy, it is hard to coordinate one DCO to give up all its services to the other DCO if only one DCO serves the SSs, and the quality of service of SSs is lower.

In Fig. 5, we evaluate the impact of the step size in the proposed sub-gradient strategy. In the simulation, we fix the step size of DCO  $o_2$  as 0.01, and change the step size of DCO  $o_1$  to 0.005, 0.01, 0.015, 0.02, respectively. According to the algorithm of the proposed sub-gradient strategy, with a different step size of DCO  $o_1$ , both DCOs compete with each other and gradually reduce their prices in the game. The game finally converges to the similar results, within different range to the same sub-optimal point of the game. Furthermore, when the step size increases, with the same starting point, the convergence speed of the corresponding DCO is faster, but the range of the sub-optimal point increases. As the starting

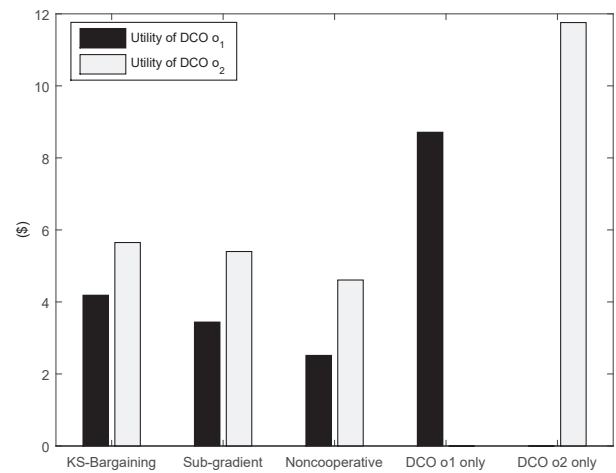


Fig. 4: The utility of each DCO with different approaches

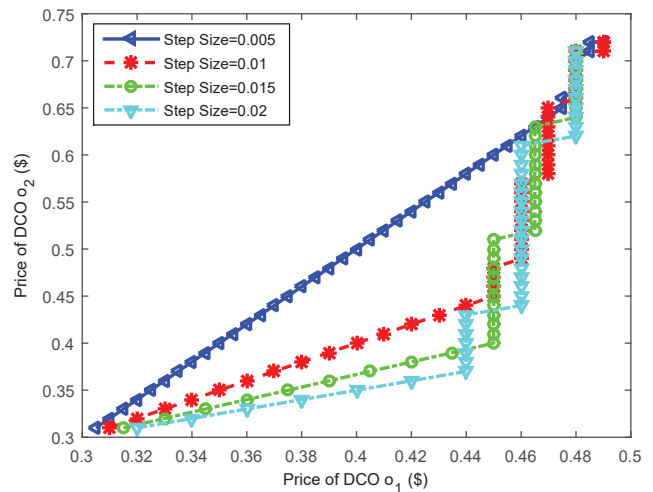


Fig. 5: The converge route in the proposed sub-gradient strategy with different steps

price of the proposed sub-gradient strategy is the maximum possible value of each DCO, we discover that when the step size increases, the service price of the DCO after convergence is higher, causing a lower utility for its served SSs. Therefore, there is a tradeoff for the SSs' choices. When the step size is large, the convergence time of service is small, but the gap between the optimal value and achieved value is large. When the step size is small, the gap between the achieving value and the optimal value is small, but the delay is large because of the slow convergence time with sub-gradient algorithm. Accordingly, if the SSs is more sensitive to the time delay of the service, it may require fast convergence of the DCO's strategies. However, the DCO may be able to set higher prices for the SSs for higher utilities. On the other hand, if the SSs expect lower prices from the DCO, the SSs may prefer a small step size of the DCO and tolerate higher convergence time in order to purchase the services with optimal prices.

In Fig. 6 and Fig. 7, we evaluate the impacts of the workload requirements of SSs and the service rates of both DCOs,



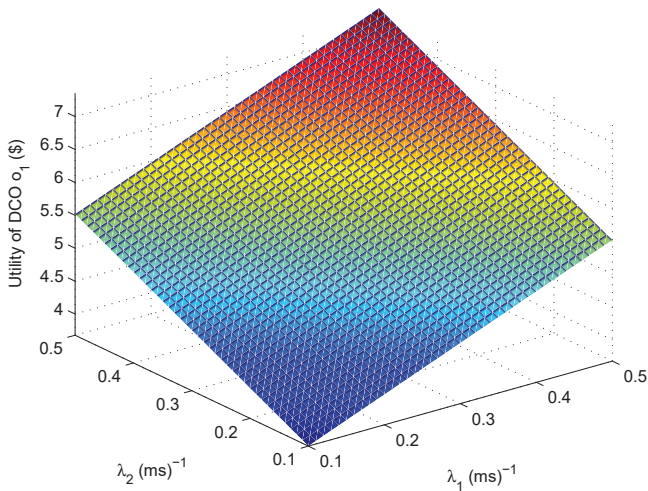


Fig. 6: Utility of DCO  $o_1$  vs the workload requirements of SS  $s_1$  and SS  $s_2$

respectively, for the proposed sub-gradient strategy. As shown in Fig. 6, without loss of generality and for better presentation, we fix the workload requirements of SS  $s_3$  as 0.6 Mbps and consider the utility of DCO  $o_1$  with different workload values of SSs  $s_1$  and  $s_2$ . When we consider the same workload value of SS  $s_1$  (or  $s_2$ ) and increase the workload value of SS  $s_2$  (or  $s_1$ ), DCO  $o_1$  is able to serve SS  $s_2$  (or  $s_1$ ) with more computing resources, so the utility of DCO  $o_1$  increases, even though there is the competition between the DCOs. In Fig. 7, we consider the performance of SS  $s_1$  with different service rate of both DCOs. When the service rates of both DCOs are small, even though the delay of the service is large for SS  $s_1$ , both DCOs set low prices for the service, and the utility of SS  $s_1$  reaches the maximum value. However, when both DCOs increase their service rate simultaneously, the DCOs set higher prices for the service of SS  $s_1$ . Accordingly, the utility of SS  $s_1$  decreases. Moreover, when DCO  $o_1$  (or DCO  $o_2$ ) has a low service rate, but DCO  $o_2$  (or DCO  $o_1$ ) improves its service rate, the utility of SS  $s_1$  first increases and then decreases. Because when the service rate of DCO  $o_1$  (or DCO  $o_2$ ) is low, its price is set at a low value. When DCO  $o_2$  (or DCO  $o_1$ ) improves its service rate, because of the competition of both DCOs, the utility of SS  $s_1$  first increases because of the low price and high quality of service. However, when DCO  $o_2$  (or DCO  $o_1$ ) continues to improve its service rate, DCO  $o_1$  (or DCO  $o_2$ ) also increases its price to receive high utility. Therefore, the utility of SS  $s_1$  decreases.

## V. RELATED WORKS

The resource allocation problem has been widely studied for data center networking systems. [8] performed a comprehensive analysis of energy efficiency in a general infrastructure supporting the cloud computing paradigm. The authors first defined a systematic approach for analysis, then utilized the approach to analyze data centers and finally extracted existing challenges and future works. In [9], the authors overviewed data center networks for cloud computing and evaluate con-

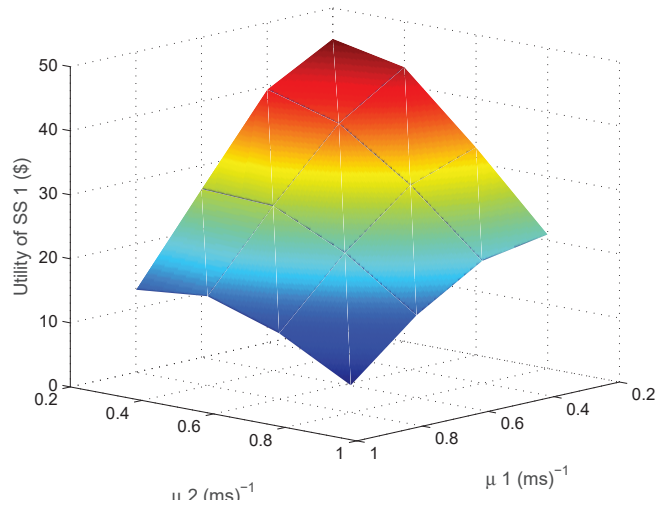


Fig. 7: Utility of SS  $s_1$  vs the service rate of both DCOs

struction prototypes for future works in order to improve the agility for multi-tenant demands in the cloud, responsiveness and scalability.

Specifically, in [10], the authors studied various problems for large-scale data centers including task assignment, data placement, and data movements. An optimization algorithm that could minimize the total cost during the big data computing services was proposed. To ensure all users to efficiently and securely share the network resources, authors in [11] evaluated the performance of several popular network sharing policies, such as SecondNet [12], Oktopus [13], Gatekeeper [14], Seawall [15], NetShare [16], and FairCloud [17], in a data center with multiple users. In [18], the authors showed a class of data center network structures based on hypergraph theory and combinatorial block design theory. Compared with the classic fat-tree model, the new structures of constructing large data center networks are more flexible and scalable. In [19], the authors made comprehensive comparison study for typical data center networks with regard to their Network Power Effectiveness (NPE). The results showed that Flattened Butterfly [20] network topology achieved high NPE in most of the cases, and the NPE of the server-centric architectures was usually higher than the NPEs of Fat-Tree [21] and VL2 [22] architectures.

In order to reduce the power consumption of network elements, the authors in [23] designed a two-level, pod-level, and core-level power optimization model, namely, Hierarchical Energy Optimization (HERO). The HERO optimized the way of shutting down network switches and links while guaranteeing full connectivity and maximizing link utilization of the network. A novel framework was proposed in [24], where high energy efficiency could be achieved by assigning virtual machines to servers and reducing the number of active switches and balance traffic flows. In order to achieve high energy efficiency, the authors in [25], took advantage of the application characteristics and topology features and proposed TE-VMA and TER algorithms to assign virtual machines and routing traffic flows, respectively. In [26], a novel architecture

of Cloud-integrated Cyber-Physical System was proposed and the enabling technologies for Complex Industrial Applications (CIA) was outlined. The paper provided solutions for the virtualized resource management techniques, the scheduling of cloud resources and life cycle management from the perspective of CIA. In [27], the authors focused on the performance-guaranteed energy-saving schemes. Based on the constraint that the transmission of every flow has to be accomplished before a rigorous deadline, the authors explored the most energy efficient way of scheduling and routing flows on the network, as well as determining the transmission speed for every flow. [28] explored the benefit of electricity price variations across time and locations for the data centers. The authors proposed a GreFar algorithm to optimize the energy cost and fairness among different organizations subject to queuing delay and maximum server inlet temperature constraints. In [29], the authors considered a QoS-constrained resource allocation problem with game theory. In the game, each participant first solved its optimization problem with binary integer programming. Then an evolutionary mechanism was designed to consider the relationships of different participants and minimize their efficiency losses. In [30], the authors considered the adaptive and stable application deployment in clouds with a multi-objective evolutionary game-theoretic framework. In the proposed framework, resource allocation strategy to applications was analyzed based on the location and the operational conditions in a cloud.

In [31], the authors considered the efficiency of water usage, and optimized a framework for the workload management of data centers. In order to design a new, upgraded and expanded data center network, the authors in [32] proposed a data center network design framework. By searching the space of all networks that are feasible under a user-specified data center model, the proposed framework maximized bisection bandwidth and minimized end-to-end latency of the designed network. In [33], the authors analyzed the management problem of data centers with multiple layers or heterogeneous devices. A new simulation tool, called Data Center Simulator with small scale operating system and storage, was proposed.

## VI. CONCLUSIONS AND FUTURE WORK

In this paper, we have studied the resource allocation problem when multiple data center operators (DCOs) and service subscribers (SSs) coexist in the networks. In order to analyze the relationships among multiple DCOs and SSs, we have modeled the scenario as a multi-leader multi-follower hierarchical game, where the DCOs act as leaders and the SSs act as followers. With the prediction of behaviors of all SSs and reactions of other DCOs, we have discussed the utilities of the DCOs in different situations and proposed the sub-gradient algorithm in the noncooperative game and coalition game with Kalai-Smorodinsky bargaining strategies to gain the benefits in terms of social welfare and the utilities of the DCOs, respectively. Simulation results have demonstrated that for the benefits of SSs, all DCOs compete with each other and all DCOs adopt the proposed sub-gradient strategies to achieve high social welfare of the game. On the other hand, for the

benefits of DCOs, DCOs form a coalition and perform Kalai-Smorodinsky bargaining behaviors so as to achieve high and fair revenues. The game analysis of the relationship between DCOs and SSs provides an outlook for the future work in the multi-DCO multi-SS scenarios. Future work is in progress to consider the resource allocation problem including both the massive and edge data center networks.

## APPENDIX A: PROOF OF LEMMA 1

*Proof:* As the cost of the SS is continuous, we take the second derivative of  $v_1$  with respect to  $m_{11}$ ,

$$\frac{\partial^2 v_1}{\partial m_{11}^2} = -\frac{2\lambda_1^2 \mu_1}{(\mu_1 m_{11} - \lambda_1)^3}. \quad (25)$$

Since the second derivative of  $v_1$  with respect to  $m_{11}$  is less than zero,  $v_1$  is a concave function of  $m_{11}$ . By setting the first derivative of  $v_1$  with respect to  $m_{11}$  as zero, i.e.,

$$\frac{\partial v_1}{\partial m_{11}} = \left( \frac{\lambda_1}{\mu_1 m_{11} - \lambda_1} \right)^2 - p_1 = 0, \quad (26)$$

we can obtain the optimal strategy for the SS, which is given by,

$$m_{11}^* = \frac{\lambda_1}{\mu_1 \sqrt{p_1}} + \frac{\lambda_1}{\mu_1}. \quad (27)$$

■

## APPENDIX B: PROOF OF LEMMA 2

*Proof:* The convergence of the sub-gradient algorithm has been proved in [45] and [46].

According to [45] and [46], the sub-gradient algorithm is able to achieve a sub-optimal solution with small ranges. Therefore, with a given moving step size, each DCO is unable to unilaterally adjust its price in order to receive higher utility when the sub-gradient algorithm converges to a sub-optimal solution.

Furthermore, when the starting price and step size  $\Delta_i, \forall i \in \{1, 2, \dots, M\}$  are fixed, the results in the second iteration are fixed. According to the mathematical induction, we suppose that at the  $Q^{th}$  iteration, the prices of both DCOs are fixed. Then in the  $(Q+1)^{th}$  iteration, according to the proposed sub-gradient algorithm, the step size is fixed and the direction from the current iteration to the next iteration is unique. Therefore, the prices of both DCOs in the  $(Q+1)^{th}$  iteration are also fixed. Therefore, based on the above, the game can converge to a unique outcome, when the starting price and step size  $\Delta_i, \forall i \in \{1, 2, \dots, M\}$  are fixed. ■

## APPENDIX C: PROOF OF LEMMA 3

*Proof:* According to the definition of Kalai-Smorodinsky bargaining, the optimal prices for DCOs  $\mathbf{p}^* = [p_1^*, p_2^*, \dots, p_M^*]^T$  satisfy

$$\begin{aligned} \mathbf{p}^* &= \arg \max \mathbf{u}_1^* \\ s.t. \quad \frac{u_1^*}{u_1^{\max}} &= \frac{u_2^*}{u_2^{\max}} = \dots = \frac{u_M^*}{u_M^{\max}}. \end{aligned} \quad (28)$$

When the prices offered by the DCOs satisfy

$$p_i = (p_i^{\max})^-, \quad (29)$$

the corresponding utility of each DCO  $o_i$ ,  $\forall i \in \{1, 2, \dots, M\}$ , is

$$u_i = \sum_{j=1}^N \frac{1}{N} u_{ij}^{\max}, \quad (30)$$

where

$$u_{ij}^{\max} = \frac{\lambda_j}{\mu_i} \sqrt{p_i^{\max}} + \frac{\lambda_j}{\mu_i} p_i^{\max} - c\beta_{ij} \frac{\lambda_j}{\mu_i \sqrt{p_i^{\max}}} - c\beta_{ij} \frac{\lambda_j}{\mu_i}. \quad (31)$$

Furthermore, as the maximum utility of each DCO  $o_i$ ,  $\forall i \in \{1, 2, \dots, M\}$ , equals

$$u_i^{\max} = \sum_{j=1}^N u_{ij}^{\max}, \quad (32)$$

where

$$u_{ij}^{\max} = \left( \frac{\lambda_j}{\mu_i} \sqrt{p_i^{\max}} + \frac{\lambda_j}{\mu_i} p_i^{\max} - c\beta_{ij} \frac{\lambda_j}{\mu_i \sqrt{p_i^{\max}}} - c\beta_{ij} \frac{\lambda_j}{\mu_i} \right). \quad (33)$$

We discover

$$\frac{u_1}{u_1^{\max}} = \frac{u_2}{u_2^{\max}} = \dots = \frac{u_M}{u_M^{\max}} = \frac{1}{N}. \quad (34)$$

According to the utility function of each DCO  $o_i$ ,  $\forall i \in \{1, 2, \dots, M\}$ ,  $u_i$  is monotonically increasing with the improvement of  $p_i$ . Therefore, when  $p_i$  approaches its maximum value, the corresponding utility of DCO  $o_i$  can also achieve the highest value. The highest utility of DCO  $o_i$  follows the constraint in (28). Accordingly, when the price of DCO  $o_i$  is

$$p_i^* = (p_i^{\max})^-, \quad (35)$$

the utility of each DCO  $o_i$ ,  $\forall i \in \{1, 2, \dots, M\}$  achieves

$$u_i^* = \sum_{j=1}^N \frac{1}{N} u_{ij}^{\max}. \quad (36)$$

## REFERENCES

- [1] A. Greenberg, J. Hamilton, D. A. Maltz, P. Patel, "The cost of a cloud: research problems in data center networks," *ACM SIGCOMM Computer Comm. Review*, vol. 39, no. 1, pp. 68-73, Jan. 2009.
- [2] Amazon Web Service, available at "http://aws.amazon.com".
- [3] Google app engine, available at "http://code.google.com/appengine/".
- [4] Google docs and spreadsheets, available at "http://docs.google.com".
- [5] Microsoft office live, available at "http://office.live.com".
- [6] Windows Azure, available at "http://www.microsoft.com/azure/".
- [7] Yahoo! Mail, available at "http://mail.yahoo.com".
- [8] T. Mastelic, A. Oleksiak, H. Claussen, I. Brandic, J.-M. Pierson and A. V. Vasilakos, "Cloud computing: Survey on energy efficiency," *ACM Computing Surveys (CSUR)*, vol. 47, no. 2, pp. 1-36, Jan. 2015.
- [9] B. Wang, Z. Qi, R. Ma, H. Guan and A. V. Vasilakos, "A survey on data center networking for cloud computing," *Computer Networks*, vol. 91, pp. 528-547, Nov. 2015.
- [10] L. Gu, D. Zeng, P. Li, and S. Guo, "Cost minimization for big data processing in geo-distributed data centers," *IEEE Trans. Emerging Topics in Computing*, vol. 2, no. 3, pp. 314-323, Sep. 2014.
- [11] T. Wang, Z. Su, Y. Xia and M. Hamdi, "Rethinking the data center networking: architecture, network protocols, and resource sharing," *IEEE Access*, vol. 2, pp. 1481-1496, Dec. 2014.
- [12] C. Guo, G. Lu, H. J. Wang, S. Yang, C. Kong, P. Sun, W. Wu, and Y. Zhang, "SecondNet: A data center network virtualization architecture with bandwidth guarantees," in *Proc. 6th Int'l Conference (Co-NEXT'10)*, Philadelphia, PA, Dec. 2010.
- [13] H. Ballani, P. Costa, T. Karagiannis, and A. Rowstron, "Towards predictable datacenter networks," in *Proc. ACM SIGCOMM'11*, vol. 41, no. 4, pp. 242-253, Toronto, Canada, Aug. 2011.
- [14] H. Rodrigues, J. R. Santos, Y. Turner, P. Soares, and D. Guedes, "Gatekeeper: Supporting bandwidth guarantees for multi-tenant datacenter networks," in *Proc. USENIX WIOV*, Portland, OR, Jun. 2011.
- [15] A. Shieh, S. Kandula, A. Greenberg, C. Kim, and B. Saha, "Sharing the data center network," in *Proc. 8th USENIX Conf. Netw. Syst. Design Implement (NSDI)*, pp. 309-322, Boston, MA, Mar. 2011.
- [16] T. Lam, and G. Varghese, "NetShare: Virtualizing bandwidth within the cloud," *UCSD Technical Report*, 2009.
- [17] L. Popa, A. Krishnamurthy, S. Ratnasamy, and I. Stoica, "FairCloud: Sharing the network in cloud computing," in *Proc. ACM SIGCOMM Conf. Applications, Technologies, Architectures, and Protocols for Computer Comm.*, pp. 187-198, Helsinki, Finland, Aug. 2012.
- [18] J. Zhang, Z. Fang, G. Qu, and S. Q. Zheng, "A new class of data center network structures," in *Proc. IEEE Globecom'13*, Atlanta, GA, Dec. 2013.
- [19] Y. Shang, D. Li, J. Zhu, and M. Xu, "On the network power effectiveness of data center architectures," *IEEE Trans. Computers*, vol. 64, no. 11, pp. 3237-3248, Nov. 2015.
- [20] J. Kim, W. J. Dally, and D. Abts "Flattened butterfly: a cost-efficient topology for high-radix networks," *ACM SIGARCH Computer Architecture News*, vol. 35, no. 2, pp. 126-137, Jun. 2007.
- [21] C. E. Leiserson, "Fat-trees: Universal Networks for Hardware-Efficient Supercomputing," *IEEE Trans. Computers*, vol. 34, no. 10, pp. 892-901, Oct. 1985.
- [22] A. Greenberg, J. R. Hamilton, N. Jain, S. Kandula, C. Kim, P. Lahiri, D. A. Maltz, P. Patel, and S. Sengupta, "VL2: A scalable and flexible data center network," *Comm. of the ACM*, vol. 54, no. 3, pp. 95-104, Mar. 2011.
- [23] Y. Zhang, and N. Ansari, "HERO: Hierarchical energy optimization for data center networks," *IEEE Systems Journal*, vol. 9, no. 2, pp. 406-415, Jun. 2015.
- [24] L. Wang, et al, "GreenDCN: A General Framework for Achieving Energy Efficiency in Data Center Networks," in *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 1, pp. 4-15, Jan. 2014.
- [25] L. Wang, F. Zhang, A. V. Vasilakos, C. Hou and Z. Liu, "Joint virtual machine assignment and traffic engineering for green data center networks," *SIGMETRICS Perform. Eval. Rev.*, vol. 41, no. 3, pp. 107-112, Jan. 2014.
- [26] Z. Shu, J. Wan, D. Zhang and D. Li, "Cloud-integrated cyber-physical systems for complex industrial applications," *Mobile Networks and Applications*, pp. 1-14, Nov. 2015.
- [27] L. Wang, F. Zhang, K. Zheng, A. V. Vasilakos, S. Ren and Z. Liu, "Energy-Efficient Flow Scheduling and Routing with Hard Deadlines in Data Center Networks," *Distributed Computing Systems (ICDCS), 2014 IEEE 34th International Conference on*, pp. 248-257, Madrid, Spain, Jun. 2014.
- [28] M. Polverini, A. Cianfrani, S. Ren and A. V. Vasilakos, "Thermal-Aware Scheduling of Batch Jobs in Geographically Distributed Data Centers," in *IEEE Transactions on Cloud Computing*, vol. 2, no. 1, pp. 71-84, Mar. 2014.
- [29] G. Wei, A. V. Vasilakos, Y. Zheng and N. Xiong, "A game-theoretic method of fair resource allocation for cloud computing services," *The journal of supercomputing*, vol. 54, no. 2, pp. 252-269, Nov. 2010.
- [30] Y. Ren, J. Suzuki, C. Lee, A. V. Vasilakos, S. Omura and K. Oba, "Balancing performance, resource efficiency and energy efficiency for virtual machine deployment in DVFS-enabled clouds: an evolutionary game theoretic approach," *Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation. ACM*, pp. 1205-1212, Vancouver, Canada, Jul. 2014.
- [31] L. Liu, S. Ren, and Z. Han, "Scalable workload management for water efficiency in data centers," in *Proc. IEEE Globecom'14*, Austin, TX, Dec. 2014.
- [32] A. R. Curtis, T. Carpenter, M. Elsheikh, A. L.-. Ortiz, and S. Keshav, "REWIRE: An optimization-based framework for unstructured data center network design," in *Proc. IEEE INFOCOM'12*, Orlando, FL, Mar. 2012.

- [33] C.-J. Chen, Y.-S. Liu, and R.-G. Chang, "DCSim: Design analysis on virtualization data center," in *Proc. 9th Int'l Conf. Ubiquitous Intelligence and Computing and Autonomic and Trusted Computing (UIC/ATC)*, Fukuoka, Japan, Sep. 2012.
- [34] E. Caron, F. Desprez, D. Loureiro, and A. Muresan, "Cloud computing resource management through a grid middleware: A case study with DIET and Eucalyptus," in *Proc. IEEE Int'l Conf. Cloud Computing 2009*, pp. 151-154, Bangalore, India, Sep. 2009.
- [35] A. Gandhi, M. Harchol-Balter, R. Das, and C. Lefurgy, "Optimal power allocation in server farms," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 37, no. 1, pp. 157-168, Jun. 2009.
- [36] Z. Liu, M. Lin, A. Wierman, S. H. Low, and L. L. Andrew, "Geographical load balancing with renewables," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 39, no. 3, pp. 62-66, Dec. 2011.
- [37] S. Ren and Y. He, "COCA: Online distributed resource management for cost minimization and carbon neutrality in data centers," in *Proc. ACM Int'l Conference on High Performance Computing, Networking, Storage and Analysis*, Denver, CO, Nov. 2013.
- [38] H. Zhang, M. Bennis, L. A. DaSilva and Z. Han, "Multi-leader Multifollower Stackelberg Game among Wi-Fi, Small Cell and Macrocell Networks," *IEEE Global Communications Conference*, pp. 4520-4524, Austin, TX, Dec. 2014.
- [39] Y. Chen, J. Zhang, and Q. Zhang, "Utility-aware refunding framework for hybrid access femtocell network," *IEEE Trans. Wireless Comm.*, vol. 11, no. 5, pp. 1688-1697, May 2012.
- [40] H. Zhang, M. Bennis, L. A. DaSilva and Z. Han, "Multi-leader multifollower stackelberg game among Wi-Fi, small cell and macrocell networks," *2014 IEEE Global Communications Conference*, pp. 4520-4524, Austin, TX, Dec. 2014.
- [41] H. Park and M. Schaar, "Fairness strategies for multi-user multimedia applications in competitive environments using the kalai-smorodinsky bargaining solution," in *Proc. IEEE Int'l Conf. Acoustics, Speech and Signal Processing (ICASSP)'07*, Honolulu, HI, Apr. 2007.
- [42] H. Zhang, Y. Xiao, L. X. Cai, D. Niyato, L. Song, and Z. Han, "A hierarchical game approach for multi-operator spectrum sharing in LTE unlicensed," in *Proc. IEEE Globecom'15*, San Diego, CA, Dec. 2015.
- [43] J. Chen and A. L. Swindlehurst, "Downlink resource allocation for multi-user MIMO-OFDMA systems: The Kalai-Smorodinsky bargaining approach," in *Proc. IEEE Int'l Workshop Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)'09*, Aruba, Dutch Antilles, Netherlands, Dec. 2009.
- [44] C. Zhang and H. Zhao, "A novel fair cooperation strategy based on Kalai-Smorodinsky bargaining solution for selfish cooperative relay networks," in *Proc. IEEE Int'l Conf. Intelligent Transportation Systems (ITSC)'14*, pp. 2107-2112, Qingdao, China, Oct. 2014.
- [45] S. P. Boyd and L. Vandenberghe, "*Convex Optimization*," Cambridge University Press, 2004.
- [46] Y. Xiao, G. Bi, and D. Niyato, "A simple distributed power control algorithm for cognitive radio networks," *IEEE Trans. Wireless Comm.*, vol. 10, no. 11, pp. 3594-3600, Nov. 2011.