Demand Response of Data Centers: A Real-time Pricing Game between Utilities in Smart Grid

Nguyen H. Tran*, Shaolei Ren‡, Zhu Han§, Sung Man Jang*, Seung Il Moon* and Choong Seon Hong*
*Department of Computer Engineering, Kyung Hee University, Korea; email: {nguyenth, smjang, moons85, cshong}@khu.ac.kr
‡School of Computing and Information Sciences, Florida International University, USA; email: sren@cis.fiu.edu
§Electrical and Computer Engineering Department, University of Houston, USA; email: zhan2@uh.edu

***Abstract*—We study the demand response (DR) of geo-distributed data centers (DCs) using a dynamic pricing scheme. Our proposed pricing scheme is constructed based on a formulated two-stage Stackelberg game where each utility sets a real-time price to maximize its own profit in Stage I; and based on these prices, the DCs' service provider minimizes its cost via workload shifting and dynamic server allocation in Stage II. First, we show that there exists a unique Stackelberg equilibrium. Then, we propose an iterative and distributed algorithm that converges to this equilibrium, where the "right prices" are set for the "right demand". Finally, we verify our proposal by traced-base simulation and results show that our pricing scheme outperforms other baseline schemes significantly.**

I. Introduction

A. Motivations, Challenges and Contributions

Data centers (DCs) are well-known as large-scale consumers of electricity and a study shows that many DC operators paid more than $10M [1] on their annual electricity bills, which continues to rise with the flourishing of cloud-computing services. Recent works have shown that DC operators can save more than $5\% - 45\%$ [2] operation cost by leveraging time and location diversities of electricity prices. However, most of the existing research is based on one important assumption: *the electricity price applying on DC does not change with demand*, which is not true since DCs have enormous energy consumption and have impacts on power prices. Many DC operators are considering how to run their geo-distributed DCs on smart grid, which is designed to coordinate the energy supply and demand more effectively through its advanced two-way communications. An important feature of smart grid is demand response (DR). DR programs seek to provide incentives to induce dynamic demand management of customers' electricity load in response to power supply conditions. Due to its huge and rapidly increasing energy consumption, DCs should be encouraged significantly to participate in the DR programs. One of the DR programs is using real-time pricing schemes to reduce the peak-to-average (PAR) load ratio by encouraging customers to shift their energy demand away from peak hours. The challenge of an effective pricing scheme is how to charge the customers with a *right price* not only at the *right time* but also on the *right amount* of customers' demand. A real-time pricing scheme is considered effective if it can mitigate the large fluctuation of energy consumption between peak and off-peak hours to increase power grid's reliability and robustness.

We consider the problem of using real-time pricing of utilities to enable the geo-distributed DCs' participation into the DR program. We show that there is an interaction between geo-distributed DCs and their local utilities; and it is the first challenge of this DR problem. Specifically, when participating in the DR program, DCs operator will distribute its energy demand geographically based on the electric prices adjusted intelligently by the local utilities. However, the utilities set their prices based on the total demand including the DCs demand, which is only known when the price is available. We clearly see that this dependency makes it difficult for both DCs and utilities to make their decisions. The second challenge is an interaction among local utilities feeding power to the geo-distributed DCs. Specifically, the DCs' decisions depend on the electric prices set by local utilities; therefore, if any local utility changes its prices, it can affect other DCs' pricing decision. Since in practice the utilities are non-cooperative, how to design a pricing mechanism that can enable an equilibrium price profile is the bottleneck of this DR program.

To tackle two above discussed challenges, our contributions can be summarized as follows: First, we transform the functional space of the geo-distributed DCs' DR program into a mathematical space of a formulated two-stage Stackelberg game. In this game, each utility will set a real-time price to maximize its own profit in Stage I; and given these prices, the DCs' operator will minimize its cost via workload shifting and dynamic server allocation in Stage II. Second, we use the backward induction method to find a unique Stackelberg equilibrium of this two-stage game. Based on this result, we propose an iterative and distributed algorithm to achieve the Stackelberg equilibrium. We also examine the algorithm's convergence where the "right prices" are set for the "right demand". Finally, we perform real-world trace-based simulation to solidify the analysis. The results show that our proposed pricing scheme can flatten the workload not only over time but also over space. Due to space limitations, all proofs can be found in the technical report available online [3].

B. Related Work

There are many existing research on DCs' cost minimization takes the electricity price for granted [2], [4], [5], which does not follow any DR programs. For those work considering DR of geo-distributed DCs, based on the interactions between DCs and utilities, we simply divide them into two categories.

1) One-way interaction: In reality one of the most popular DR programs of DCs is Coincident Peak Pricing (CPP), which is studied in [6]. However, current DCs do not respond actively to the warning signals due to the uncertainty of these warnings [6], which motivates researchers to devise more effective DR approaches. The authors in [7] use a "prediction-based" method where the customers (DCs) respond to the prices which are chosen based on a supply function. Hence, in this work only customers respond to a predicted price while there is no action from the power suppliers to set the prices corresponding to the demand.

2) Two-way interaction: Two recent papers [8], [9] in this category are highly related to our work. Both consider dynamic pricing mechanisms that make utilities and DCs coupled. However, the system model of [9] assumes that all utilities cooperate to solve a social optimization problem, which is not relevant to current practice since there is no information exchange between utilities in reality. On the other hand, the pricing scheme of [8] is based on a heuristic approach, which cannot maximize the utilities' profit as well as minimize their cost.

II. System Model and Problem Formulation

We consider discrete time model $t \in \mathcal{T} = \{1, \dots, T\}$ of a billing cycle (e.g., typically a month), where the length of a time slot t matches an interval at which the DCs' decisions and utilities' real-time prices can be updated (such as one hour). Let $\mathcal{I} = \{1, \dots, I\}$ denote the set of sites where DCs are located. Each DC i is assumed to be powered by a local utility company and have S_i homogeneous servers.

We observe that there exists a special mutual interaction between DCs and utilities that can be modeled as a leader-follower game, i.e. two-stage Stackelberg game. Specifically, the utilities are the leaders that simultaneously set the prices to maximize their profit in Stage I and DCs will make their decisions on workload shifting and dynamic server provisioning to minimize their cost in Stage II. We describe this two-stage game formulation in the reverse sequence, starting with Stage-II optimization problem.

A. DCs' Cost Minimization in Stage II

We first describe the workload model of a typical DCs. We then elaborate the DCs' cost focusing on the energy cost and delay cost model. Finally, we formulate the Stage-II DC's cost minimization.

1) Workload Model: Even though DCs can support a wide range of workloads, we generally divide them into two typical types of workload: interactive jobs and batch jobs. While the former is delay-sensitive and non-flexible (e.g. banking service, online game, etc.), the later is delay-tolerant and flexible to schedule (e.g. scientific application, map reduce workload, etc.). We assume that each DC processes its batch jobs locally (i.e. batch jobs cannot be re-directed to other DCs for load balancing) similarly to [5]. For interactive jobs, we denote the total arrival rate to the DCs' front-end server, (i.e. all DCs are managed by a DC service provider (DSP)) at time t by $\Lambda(t)$ and this front-end server is responsible for splitting the total incoming workload $\Lambda(t)$ into separate workloads of geo-dispersed DCs, denoted by $\{\lambda_i(t)\}_{i \in \mathcal{I}}$.

2) DC's Cost and QoS Model: The DSP tries to not only minimize its energy and migration cost but also guarantee the QoS requirements of the interactive jobs.

Energy Cost: Since batch jobs are flexible to schedule, we assume that the batch job processing consumes an amount of energy $e_i^b(t)$ of each DC i in timeslot t. On the other hand, the energy consumption[1] of delay-sensive jobs at DC i is [1]

$$e_i^d(t) = s_i(t)\big(P_{idle} + (P_{peak} - P_{idle})U_i(t) + (PUE(t) - 1)P_{peak}\big)$$

where $s_i(t)$ is the server count, μ_i is the service rate of a server, P_{peak} and P_{idle} are the server's peak and idle power, respectively, $U_i(t) = \frac{\lambda_i(t)}{s_i(t)\mu_i}$ is the average server utilization, and $PUE(t)$ is the power usage effectiveness measuring the energy (e.g. cooling) efficiency of the DC. We can rewrite $e_i^d(t)$ as follows

$$e_i^d(t) = a_i \lambda_i(t) + b_i(t) s_i(t), \quad \forall i \in \mathcal{I}, t \in \mathcal{T}, \tag{1}$$

where $a_i = (P_{peak} - P_{idle})/\mu_i$ and $b_i(t) = P_{idle} + (PUE(t) - 1)P_{peak}$. Therefore, denoting the total energy by

$$e_i(t) = e_i^d(t) + e_i^b(t), \tag{2}$$

and given a price $p_i(t)$ at time t, the energy cost of DC i is $e_i(t)p_i(t)$.

Migration Cost: Since migrating the workload from front-end server to geo-distributed DCs can be very costly (e.g. migrating virtual machines or video content requests over the Internet could be expensive due to reserving bandwidth from ISP), we model the migration cost to DC i as

$$\omega d_i c_i(\lambda_i), \tag{3}$$

where d_i is the transmission delay from front-end server to DC i, ω is a weight factor and $c_i(\lambda_i)$ is a function assumed to be strictly increasing and convex. Since d_i is proportional to the distance, it can be assumed to be a constant and we see that migrating more requests from the front-end server to a farther DC is more costly. For analysis tractability, we choose a quadratic function $c_i(\lambda_i(t)) = \lambda_i(t)^2$ since it is widely used to penalize the action in control theory.

QoS Constraint: We assume that each delay-sensitive request imposes a maximum delay D_i that the DSP has to guarantee when shifting this request to DC i. Therefore, the QoS constraint in terms of delay guarantee can be modeled as follows

$$\frac{1}{s_i(t)\mu_i - \lambda_i(t)} + d_i \le D_i, \forall i, \tag{4}$$

where $1/(s_i(t)\mu_i - \lambda_i(t))$ is the average delay time of a request processed in DC i with arrival rate $\lambda_i(t)$ and service rate $s_i(t)\mu_i$ by queueing theory.

[1] We alternatively use either power or energy since the time slot is the same.

3) Problem Formulation: Our model focuses on two key controlling "knobs" of DCs' cost minimization: the workload shifting to DC $\lambda_i(t)$ and the number of active servers provisioned $s_i(t)$ at site i, $\forall i$. Then, the Stage-II DC cost minimization is given by

$$\begin{array}{lll} \mathsf{DC}: & \text{minimize} & \sum_{t=1}^{T}\sum_{i=1}^{I} e_i(t)p_i(t) + \omega d_i \lambda_i(t)^2 \qquad (5)\\ & \text{subject to} & constraints\ (1),(2),(4),\\ & & \sum_{i=1}^{I} \lambda_i(t) = \Lambda(t),\ \forall t, \qquad (6)\\ & & 0 \le s_i(t) \le S_i,\ \forall i,t, \qquad (7)\\ & & 0 \le \lambda_i(t) \le s_i(t)\mu_i,\ \forall i,t, \qquad (8)\\ & \text{variables} & s_i(t), \lambda_i(t),\ \forall i,t. \qquad (9) \end{array}$$

While constraints (1), (2) and (4) are the definitions of the objective function and QoS contraint, the remaining constraints are straight-forward. In (6), all of the incoming workload must be served by some DCs. Moreover, (7) limits the number of active servers and (8) means that the total workload assigned to a DC must be less than its capacity.

B. Non-Cooperative Pricing Game in Stage I

In this stage, we first describe the utility's revenue and cost models to form the individual objective of each utility's profit maximization. We next formulate the non-cooperative pricing game between utilities.

1) Utility Revenue's Model: The optimal energy consumption of DCs at time t that can be obtained from solving DC depends on prices $p_i(t)$, $\forall i$, of all utilities. Denote the corresponding optimal power demand by $e_i(p(t))$, where $p(t) := \{p_i(t)\}_{i\in\mathcal{I}}$. We further assume that due to the grid regulations at each region, the lower and upper bound of the real-time price should be imposed and denoted by p_i^l and p_i^u, $\forall i, t$, respectively. Furthermore, besides the power demand of DCs, each utility has its own background load (e.g. residential demand). Since there are considerable works focusing on the residential DR programs, we assume that the background load of utility i, denoted by $B_i(p_i(t))$, also responds to the price and can be modeled by the following function

$$B_i(p(t)) = \begin{cases} B_i^l, & p_i(t) \le p_i^l;\\ \alpha - \beta p_i(t), & p_i^l \le p_i(t) \le p_i^u;\\ B_i^u, & p_i(t) \ge p_i^u, \end{cases} \qquad (10)$$

where B_i^l and B_i^u are the minimum of maximum background demands of site i due to the physical constraints of consumers (i.e. maximum and minimum power of electric devices or vehicles). Based on the total power requested by DCs and background's demands, the revenue of utility i at time t is given by

$$rev_i(p(t)) = \big(e_i(p(t)) + B_i(p_i(t))\big)p_i(t). \qquad (11)$$

2) Utility Cost's Model: On the other hand, every utility incurs a cost when it serves the customers' load. When load increases, the utility's cost also increases since normally blackouts happen due to the overload, which is a disaster to any utilities. Hence, we can model the utility's cost based on a widely-used electric load index (ELI) as follows

$$cost_i(p(t)) = \gamma ELI = \gamma\left(\frac{e_i(p(t)) + B_i(p_i(t))}{C_i(t)}\right)^2 C_i(t),$$

where $C_i(t)$ is utility i capacity at time t, and γ reflects the weight of the cost. ELI is an important economic indicator where a high value of ELI notifies the utility to spend more for stability investment [9].

3) Stage-I Pricing Game Formulation: In reality, the geo-distributed utilities usually have no communication exchange to optimize the social performance. Instead, each utility i at time slot t has its own goal to maximize its profit, which is defined as the difference between revenue and cost as follows

$$u_i(p_i(t), p_{-i}(t)) = rev_i(p(t)) - cost_i(p(t)), \qquad (12)$$

where $p_{-i}(t)$ denotes the price vector of other utilities except i. This notation comes from an observation that there is a game between utilities because the profit of each utility not only depends on its energy price but also on the others'. Hence, the Stage-I utility profit maximization game is defined as follows

- *Players*: the utilities in the set $\mathcal{I}$;
- *Strategy*: $p_i^l \le p_i(t) \le p_i^u$, $\forall i \in \mathcal{I}, t \in \mathcal{T}$;
- *Payoff function*: $\sum_{t=1}^{T} u_i(p_i(t), p_{-i}(t))$, $\forall i \in \mathcal{I}$.

III. Two-Stage Stackelberg Game: Equilibria and Algorithm

In this section, we first apply the backward induction method to solve the Stackelberg game. Then, we propose an iterative algorithm to reach an equilibrium of this game.

A. Backward Induction Method

1) Optimal Solutions at Stage II: We realize that the stage-II DCs' cost minimization can be decomposed into independent problems at each time slot t. Henceforth, we only consider a specific time period and drop the time dependence notation for ease of presentation. In this stage, DCs cooperate with each other to minimize the total cost by determining the workload allocation λ_i and the number of active servers s_i at each DC i. It is easy to see that the DCs' cost minimization is a convex optimization problem.

First, we observe that constraint (4) must be active because otherwise the DSP can decrease its energy cost by reducing $s_i(t)$. Hence, we have (4) is equivalent to

$$s_i(\lambda_i) = \left[\frac{1}{\mu_i}\left(\lambda_i + \tilde{D_i}^{-1}\right)\right]_0^{S_i}, \qquad (13)$$

where $[.]_x^y$ is the projection onto the interval $[x, y]$ and $\tilde{D}_i := D_i - d_i$. In practice most DCs can have enough number of servers to serve all requests at the same time due to the illusion of infinite capacity of DCs [4]. Therefore, we adopt $s_i(\lambda_i) =$

$\frac{1}{\mu_i}\left(\lambda_i + \tilde{D_i}^{-1}\right)$ in the sequel. By substituting this $s_i(\lambda_i)$ into the objective of DC, we have an equivalent problem DC′ as follows

$$\mathsf{DC'}: \quad \min_{\lambda}. \qquad \sum\nolimits_{i=1}^{I} f_i(\lambda_i) \tag{14}$$

$$\text{s.t.} \qquad \sum\nolimits_{i=1}^{I} \lambda_i = \Lambda, \tag{15}$$

$$\lambda_i \geq 0, \ \forall i, \tag{16}$$

where $f_i(\lambda_i) := \omega d_i \lambda_i^2 + p_i\left(a_i + \frac{b_i}{\mu_i}\right)\lambda_i + p_i\left(e_b + \frac{b_i \tilde{D_i}^{-1}}{\mu_i}\right)$. It can be seen that DC′ is a strictly convex problem, which has a unique solution. Since DSP likes to have $\lambda_i > 0$, $\forall i$, in order to utilize all DCs resources, we characterize the unique solution of DC and a necessary condition to achieve this solution with the optimal $\lambda_i^* > 0$, $\forall i$, as the following result.

Theorem 1. *Given a price vector p, we have the unique solutions of Stage-II* DC *problem as follows*

$$\lambda_i^* = \frac{\nu^* - p_i A_i}{2\omega d_i} > 0, \tag{17}$$

$$s_i^* = \frac{1}{\mu_i}\left(\lambda_i^* + \tilde{D_i}^{-1}\right), \forall i, \tag{18}$$

only if

$$\omega > \omega_{th}^1 := \left(\hat{d}\max_i\{p_i A_i\} - \sum\nolimits_{i=1}^{I} p_i A_i / d_i\right) / 2\Lambda, \tag{19}$$

where $\hat{d} := \sum_{i=1}^{I} 1/d_i$, $A_i := a_i + \frac{b_i}{\mu_i}$ *and* $\nu^* = \frac{1}{\hat{d}}\left(2\omega\Lambda + \sum_{i=1}^{I} p_i A_i / d_i\right)$.

We can consider condition (19) as a (lower bound) guideline for DSP to choose an appropriate weight factor ω to ensure all DCs have positive requests.

2) Nash Equilibrium at Stage I: We continue to characterize the Nash equilibrium of the Stage-I game based on the Stage-II solutions. In the non-cooperative game, one of the most important questions is whether there exists a unique Nash equilibrium. In this Stage-I game, given all other utilities' strategies p_{-i}, a natural strategy of utility i is the best response strategy as follows

$$\mathbf{BR}_i(p_{-i}) = \arg\max_{p_i^l \leq p_i \leq p_i^u} u_i(p_i, p_{-i}), \forall i. \tag{20}$$

where $u_i(p_i, p_{-i}) = \Big(e_i^*(p) + B_i(p_i)\Big)p_i - \gamma C_i\left(\frac{e_i^*(p)+B_i(p_i)}{C_i}\right)^2$ and $e_i^*(p) = (a_i\lambda_i^* + b_i s_i^*) + e_i^b$. With λ_i^* and s_i^* obtained from Theorem 1, $e_i^*(p)$ is equal to

$$\frac{A_i^2 p_i}{2\omega d_i}\left(\frac{1}{\hat{d}d_i} - 1\right) + \frac{A_i}{2\omega\hat{d}d_i}\sum_{j\neq i}\frac{A_j p_j}{d_j} + \frac{A_i\Lambda}{\hat{d}d_i} + \frac{b_i}{\mu_i\tilde{D}_i} + e_i^b.$$

When all utilities play best response strategies, a Nash equilibrium p^e is a profile that satisfy $p_i^e = \mathbf{BR}_i(p_{-i}^e)$, $\forall i$, i.e. every utility's strategy is its best response to others' strategies. Then we have this result.

Theorem 2. *(Existence and Uniqueness) There exist a Nash equilibrium of the Stage-I game. Furthermore, if*

$$\omega \geq \omega_{th}^2 := \max_i\left\{\frac{A_i\sum_{j\neq i}A_j/d_j - A_i^2\hat{d}(1 - 1/(d_i\hat{d}))}{2\beta\hat{d}d_i}\right\}, \tag{21}$$

then starting from any initial point, the best response strategies converge to a unique Nash equilibrium p^e of the Stage-I game.

B. Distributed Algorithm

We first describe the detailed operations of the proposed algorithm and provide its convergence performance. Next, we discuss about the practical implementation issue of the algorithm.

Algorithm 1 Demand Response of Data Center with Real-time Pricing

```
1: initialize: Set $k = 0$, $p_i^{(0)} = p_i^u$, $\forall i$, and $\omega$ satisfies (21);
2: repeat
3:     Utility $i$ broadcasts its $p_i^{(k)}$ to all customers, $\forall i$;
4:     The front-end server collects $p^{(k)}$ from all DCs, updates $e_i^*(p)^{(k)}$ and send it back to DC $i$, $\forall i$;
5:     Each DC $i$ reports its $e_i^*(p)^{(k)}$ to the local utility;
6:     Utility $i$ receives the demand responses from the local DC $e_i^*(p)^{(k)}$ and background users $B_i(p)^{(k)}$, then solves $p_i^{(k+1)} = \mathbf{BR}_i(p_{-i}(k))$, $\forall i$;
7: until $\left|p^{(k+1)} - p^{(k)}\right| < \epsilon$.
```

1) Proposed Algorithm's Operations and Convergence: We continue proposing a distributed algorithm, shown in Algorithm 1 (Alg. 1), that can achieve the Nash equilibrium. We assume that Alg. 1 operates at the beginning of each pricing update period (i.e. one hour) and the algorithm runs for many iterations (communication rounds with a parameter k) until it converges to a price setting equilibrium. Here, based on the total incoming workload, the front-end server of the DSP first collects all prices from its local DCs and calculate the optimal energy consumption (line 4). After that the front-end server will feedback these energy consumption data to its local DCs, which then forwards its own information to the local utility (line 5). Each utility solves its own profit maximization problem to find an optimal price, then broadcasts this price to its local DCs and background customers (line 6). The process repeats until the game reach the Nash equilibrium as the prices converge (line 7). At this state the price setting is finalized and applied to the whole time slot t. We can see that Alg. 1 converge to a unique Nash equilibrium of Stage-I game according to the best response strategies by Theorem 2.

2) Practical Issues and Implementation Discussion: First, we assume the DSP deploys a front-end server to distribute the incoming workload to DCs. This can be done by using various practical solutions such as incorporating the authoritative DNS servers (which is used by Akamai) or HTTP ingress proxies (which is used by Google and Yahoo) into the front-end servers. Furthermore, in reality there is only a sub-set of

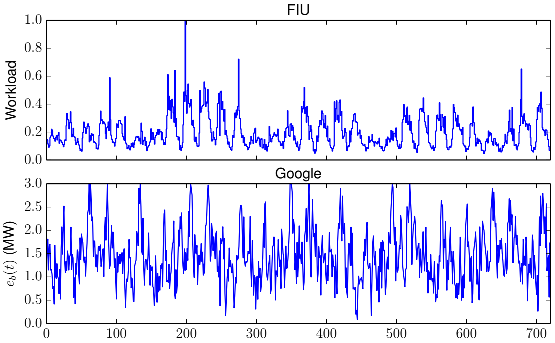


Fig. 1. The request rate from FIU (top) and batch job power from Google (bottom).

TABLE I
AVERAGE OPTIMAL PRICES COMPARISONS WITH γ EFFECT

Sites	FIU			
	Baseline 1	Alg. 1 $\gamma = 1$	Alg. 1 $\gamma = 4$	Alg. 1 $\gamma = 8$
1	42.08	19.22	19.71	20.33
2	87.82	35.76	36.77	38.03
3	60.69	27.56	28.32	29.28
4	33.17	15.29	15.62	16.04
5	29.75	14.10	14.37	14.72
6	35.43	16.32	16.70	17.18

DCs to which a workload type can be routed to due to the availability resource constraint of each DC. This issue can be easily addressed by incorporating more constraint into our model such as [10], and in practice we can implement it by classifying the workload types at front-end server before routing. Second, we assume that DCs communicate with its front-end server by choosing one of the egress links of its Internet Service Provider (ISP). Specifically, the total time of one iteration consists of the transmission time and computational time. While the transmission time from utilities to DCs (and vice versa) is from 1 to 10 ms over a broadband speed of 100 Mbps, it is from 50 to to 100 ms for a one-way communication between DCs and the front-end servers over a current ISP's path. The computational time depends on the processing power of the front-end server and smart meters on calculating (20), which is low-complexity problem and can be in the time-scale of microsecond [11]. Based on our simulation results, the equilibrium can be reached in less than 5 iterations, which means that the total time of Alg. 1 can be approximately one second for each one-hour time slot.

IV. Trace-Based Simulations

In this section, we conduct trace-based simulations to validate our analysis and evaluate the performance of Alg. 1. First, we present the simulation setups. Next, we describe the baseline pricing methods for comparison. Finally, we show the results and analyze the performance comparison.

A. Setups

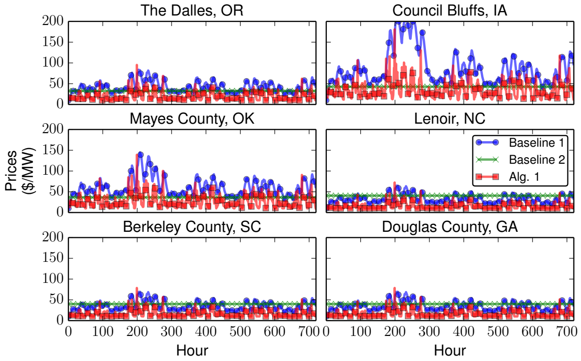


Fig. 2. Optimal prices at six locations.

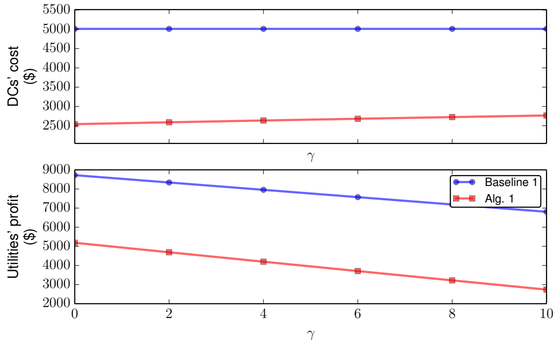
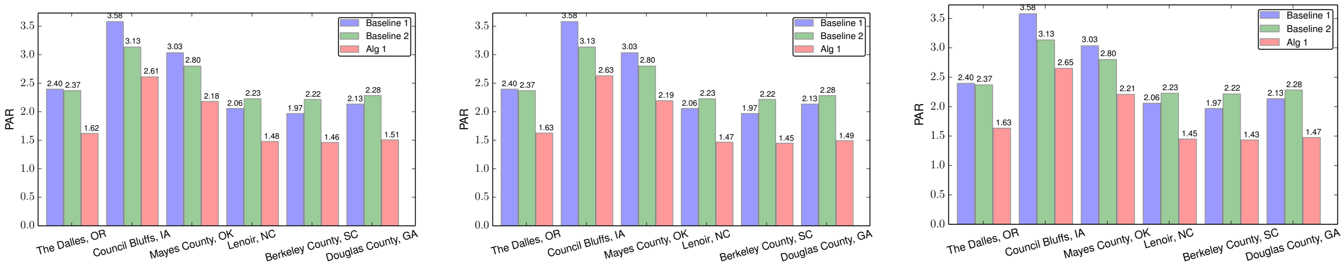


Fig. 3. Effect of γ to average DCs' cost and utilities' profit.

1) DCs: We consider six geo-distributed DCs where their PUEs are set to 1.5. The homogeneous servers of six DCs have peak power of 200 W and idle power of 100 W, and the service rate of each server is chosen uniformly between 1.1 and 1.2. We set ω to 1 to satisfy (19) with d_i is proportional to the distance from front-end server and D_i is chosen uniformly between 100 and 300 ms, $\forall i$. There are two realistic traces that we use for the simulation. The first trace is the incoming workload at the front-end server, which is scaled respectively to service rates and shown in Fig. 1. This data is profiled from January 1 to June 30, 2012, at the Florida International University (FIU) [5]. The second trace is the power demand of delay-tolerant batch jobs $e_b(t)$ of Google by recent study [12]. The workload series and batch job power demand spans over 30 days and each point of series is a one-hour period.

2) Utilities: Since lacking the public information of local utilities, we assume that at each time slot all utilities have the capacities $C_i(t)$ uniformly distributed in the range of 25 and 30 MW, which is a standard measure for a medium-size utility. The lower and upper bounds of the real-time price, p_i^l and p_i^u, are set to 1 and 300 ($/MWh), respectively. The utility cost parameter γ is set to 1 unless otherwise stated.

Regarding to the residential power demand $B(p)$, α and β parameters are chosen uniformly in the range of $[25, 30]$ and $[0.25, 0.30]$, respectively.



(a) (b) (c)

Fig. 4. PAR at six locations with: (a) $\gamma = 1$, (b) $\gamma = 4$, c) $\gamma = 8$.

B. Baseline Pricing Schemes for Comparison

We consider two baseline pricing schemes for the simulation comparison as follows.

1) Baseline 1: The first baseline is based on the proposed dynamic pricing scheme of [8]. At each utility i, this pricing scheme can be briefly described as follows

$$p_i(t+1) = \delta(PD_i(t) - PS_i(t)) + p_i(t), \qquad (22)$$

where PD_i and PS_i are the power demand and supply of utility i at time t. We set δ to 0.5 in all simulation scenarios.

2) Baseline 2: The second baseline is based on the Google's contract with their local utilities. According to the empirical study in [13], there are six Google's DCs powered by their local utilites at the following locations: The Dalles, OR; Council Bluffs, IA; Mayes County, OK; Lenoir, NC; Berkeley County, SC and Douglas County, GA. In these locations, Google's DCs are infered to have long-term contracts with their local utilities as the following fixed rates $[32.57, 42.73, 36.41, 40.68, 44.44, 39.97]$ \$/MWh, respectively. We use this baseline mainly for PAR comparisons since it is not fair to compare static prices versus dymamic prices in terms of cost or profit.

C. Results

We first show the optimal prices by Alg. 1 to compare with other baseline schemes. Then we compare the total DCs' cost and utilities' profit. Finally, we compare the PAR performance of three schemes.

1) Optimal solutions: We first provide a sample-path optimal prices of three schemes at six locations in Fig. 2 corresponding to two workload traces. Since Baseline 1 and Alg. 1 employ dynamic pricing mechanisms, we can observe that the utilities' prices of these two schemes vary according to the workload pattern. We also observe the effect of migration cost to the optimal prices in this figure. Since the nearest DCs to the front-end server are sites 2 and 3, Fig. 2 shows that all dynamic pricing schemes set high prices at these sites compared with the other sites. Furthermore, we also investigate the effect γ in Stage I since our simulation shows that the average prices of Alg. 1 are not affected by ω. Table I shows that if we increase γ, then the Alg 1's optimal prices also increase since the higher the weight utilities' ELI cost factor is, the more conservative utilities are in terms of reliability by raising the prices. We also see that Baseline 1 always overprices Alg. 1.

2) Total DCs' cost and utilities' profit: We also evaluate the effect of parameter γ to average DCs' cost and utilities' profit in Fig. 3. First, we can see that Baseline 1 with higher prices has higher DCs' cost and utilities' profit than those of Alg. 1. Therefore, Alg. 1 can give more incentives to encourage the DCs to join the DR program. Second, we can see that when γ increases, the utilities' profit of both schemes decrease due the cost in (12). With Alg. 1, we see that small γ is favorable because it can provide low DCs' cost and high utilities profit.

3) PAR: The final factor that we examine is PAR, which is one of the most important metrics to measure the effectiveness of designs for smart grid since the fluctuation of energy consumption between peak and off-peak hours indicate power grid's reliability and robustness. Reducing PAR is the ultimate goal of any DR program designs, so is our proposed Alg. 1. Fig 4 compares the PAR of three schemes with different γ. The most important observation is that the PAR's performance of Alg. 1 outperforms those of other schemes over time and space significantly.

V. Conclusion and Future Work

We study the DR of geo-distributed DCs using smart grid. We first formulate this DR program into a two-stage Stackelberg game to model the interactions between utilities and DCs. Specifically, in this game the role of each utility is setting a price to maximize its profit, while the DCs minimize its cost. We then characterize the existence of a Stackelberg equilibrium of this game where all utilities agree on a stable price setting without deviation intention. We next develop an iterative and distributed algorithm to reach one equilibrium point. We validate and our proposal's effectiveness with the simulation results based on realistic traces.

Acknowledgment

This research was funded by the MSIP (Ministry of Science, ICT and Future Planning), Korea in the ICT R&D Program 2014.

We sincerely thank Cheng Wang for providing the useful traced data.

References

[1] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, and B. Maggs, "Cutting the electric bill for internet-scale systems," in *Proc. ACM SIGCOMM 2009*, Barcelona, Spain, 2009, pp. 123–134.

[2] L. Rao, X. Liu, L. Xie, and W. Liu, "Minimizing Electricity Cost: Optimization of Distributed Internet Data Centers in a Multi-Electricity-Market Environment," in *Proc. IEEE INFOCOM*, San Diego, CA, USA, Mar. 2010, pp. 1–9.

[3] N. H. Tran, S. Ren, Z. Han, S. man Jang, S. I. Moon, and C. S. Hong, "Demand Response of Data Centers:A Real-time Pricing Game between Utilities in Smart Grid," Tech. Rep., 2014. [Online]. Available: http://networking.khu.ac.kr/nguyenth/MultiUtilityGame2.pdf

[4] Z. Liu, M. Lin, A. Wierman, S. H. Low, and L. L. Andrew, "Greening geographical load balancing," in *Proc. ACM SIGMETRICS*, San Jose, California, USA, Jun. 2011, pp. 233–244.

[5] S. Ren and Y. He, "COCA: online distributed resource management for cost minimization and carbon neutrality in data centers," in *Proc. SC13 Int. Conf. High Perform. Comput. Networking, Storage Anal.*, Denver, Colorado, USA, Nov. 2013, pp. 39:1–39:12.

[6] Z. Liu, A. Wierman, Y. Chen, B. Razon, and N. Chen, "Data center demand response: Avoiding the coincident peak via workload shifting and local generation," *Perform. Eval.*, vol. 70, no. 10, pp. 770–791, Oct. 2013.

[7] P. Wang, L. Rao, X. Liu, and Y. Qi, "D-Pro: Dynamic Data Center Operations With Demand-Responsive Electricity Prices in Smart Grid," *IEEE Trans. Smart Grid*, vol. 3, no. 4, pp. 1743–1754, Dec. 2012.

[8] Y. Li, D. Chiu, C. Liu, and L. Phan, "Towards dynamic pricing-based collaborative optimizations for green data centers," in *IEEE 29th Int. Conf. Data Eng. Work.*, Brisbane, Australia, 2013, pp. 272–278.

[9] H. Wang, J. Huang, X. Lin, and H. Mohsenian-Rad, "Exploring smart grid and data center interactions for electric power load balancing," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 41, no. 3, pp. 89–94, Jan. 2014.

[10] P. X. Gao, A. R. Curtis, B. Wong, and S. Keshav, "It's not easy being green," in *Proc. ACM SIGCOMM*, Helsinki, Finland, Aug. 2012, pp. 211–222.

[11] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, Mar. 2004.

[12] C. Wang, B. Urgaonkar, and Q. Wang, "Data center cost optimization via workload modulation under real-world electricity pricing," *Arxiv Prepr. arXiv1308.0585*, pp. 1–14, 2013. [Online]. Available: http://arxiv.org/abs/1308.0585

[13] H. Xu and B. Li, "Reducing Electricity Demand Charge for Data Centers with Partial Execution," *arXiv Prepr. arXiv1307.5442*, 2013. [Online]. Available: http://arxiv.org/abs/1307.5442