Proactive Demand Response for Data Centers:
A Win-Win Solution

Hao Wang, *Student Member, IEEE,* Jianwei Huang, *Senior Member, IEEE,*
Xiaojun Lin, *Senior Member, IEEE,* and Hamed Mohsenian-Rad, *Senior Member, IEEE*

Abstract—In order to reduce the energy cost of data centers, recent studies suggest distributing computation workload among multiple geographically dispersed data centers, by exploiting the electricity price difference. However, the impact of data center load redistribution on the power grid is not well understood yet. This paper takes the first step towards tackling this important issue, by studying how the power grid can take advantage of the data centers' load distribution *proactively* for the purpose of power load balancing. We model the interactions between power grid and data centers as a two-stage problem, where the utility company chooses proper pricing mechanisms to balance the electric power load in the first stage, and the data centers seek to minimize their total energy cost by responding to the prices in the second stage. We show that the two-stage problem is a bilevel quadratic program, which is NP-hard and cannot be solved using standard convex optimization techniques. We introduce benchmark problems to derive upper and lower bounds for the solution of the two-stage problem. We further propose a branch and bound algorithm to attain the *globally optimal* solution, and propose a heuristic algorithm with low computational complexity to obtain an alternative *close-to-optimal* solution. We also study the impact of background load prediction error using the theoretical framework of robust optimization. The simulation results demonstrate that our proposed scheme can not only improve the power grid reliability but also reduce the energy cost of data centers.

Index Terms—Smart grid, data center, demand response, dynamic electricity pricing, load balancing, proactive design.

Nomenclature

Acronyms

PS1	Stage-1 problem
PS2	Stage-2 problem
PI	Integrated problem
RS1	Stage-1 of the restricted problem
RS2	Stage-2 of the restricted problem
PE1	Equivalent problem of the Stage-1 problem
PE2	Equivalent problem of the Stage-2 problem
PR1	Relaxed Stage-1 problem
WCP	Worst-case performance optimization problem

Sets

$\mathcal{T}$	Set of time slots
$\mathcal{N}$	Set of data centers

Indices

t	Index of time slots
i	Index of data centers

Parameters

T	Number of time slots
N	Number of data centers
L^t	Total incoming workload within time slot t
M_i	Total number of servers in data center i
μ_i	Service rate of servers in data center i
d_i^t	Transmission delay to data center i in time slot t
D	Delay bound
P_{idle}	Average idle power of server
P_{peak}	Average peak power of server
R_i	Power usage effectiveness of data center i
ξ	Empirical parameter of power consumption
α_i^t	Base price for data center i in time slot t
β_i	Sensitivity parameter of price for data center i
Q_i^t	Available supply to data center i in time slot t
B_i^t	Background load in location i and time slot t
C_i	Power capacity in location i
$\underline{\pi}_i^t$	Price lower bound for data center i in time slot t
$\overline{\pi}_i^t$	Price upper bound for data center i in time slot t
$\pi_{\max}^t$	Maximum average price in time slot t
θ_i	Coefficient for energy consumption of data center i
E^t	Total energy required in time slot t
$\underline{E}_i^t$	Energy lower bound for data center i in time slot t
$\overline{E}_i^t$	Energy upper bound for data center i in time slot t
$\Delta_{i,\min}^t$	Error lower bound in location i and time slot t
$\Delta_{i,\max}^t$	Error upper bound in location i and time slot t

Variables

λ_i^t	Workload assigned to data center i in time slot t
x_i^t	Number of active servers in data center i and time t
e_i^t	Energy consumption of data center i in time slot t
s_i^t	Billing reference for data center i in time slot t
π_i^t	Unit energy price for data center i in time slot t
r_i^t	Electric load ratio in location i and time slot t
$\underline{z}_i^t$	Binary variables
$\overline{z}_i^t$	Binary variables
δ_i^t	Load prediction error in location i and time slot t

I. INTRODUCTION

Energy management of large and distributed data centers has become an increasingly important problem. With the fast development of cloud computing services, it is now common for a cloud provider (*e.g.*, Google, Microsoft, and Amazon) to build multiple, large, and geographically dispersed data centers across the continent. Each data center may include hundreds of thousands of servers, massive storage equipment, cooling facilities, and power transformers. The energy consumption and cost of data centers hence can be significant [2]. For example, Google reported in 2011 that its data centers continuously draw almost 260 MW of power, which is more than what Salt Lake City consumes [3]. Microsoft's data center in Washington US consumes 48 MW of power, which is equivalent to the power consumption of about 40,000 households. This has motivated growing research activities toward optimizing the data center operations to reduce the total energy cost. For example, Qureshi *et al.* in [2] proposed an energy cost minimization method for distributed data centers to exploit electricity price difference. The idea is later extended in [4]–[10].

However, most existing studies of energy management of distributed data centers have focused on the energy cost minimization from the viewpoint of data centers, but fail to consider the impact of such energy management practice on the power grid. Note that, due to their enormous energy consumption, data centers are expected to have a great influence on the operation of the power grid [11]. Without taking such impact into account, these energy management schemes may adversely affect power-grid stability and load balancing.

In this paper, we aim to study the energy cost minimization of distributed data centers based on their impact to the power grid. We seek to benefit from the recent advances in two-way communications that are available in smart grid [12] to allow interactions and coordinations between energy suppliers and consumers in real time to improve demand side management. In our proposed framework, the utility company can set dynamic prices to the *demand-responsive data centers*, and the data centers can dynamically change energy consumption in response to the price changes. This can effectively coordinate demand with supply, and hence avoid unintended power overloading.

The overall framework of our proposed system setup is shown in Fig. 1. Cloud service users send computing requests via Internet to the cloud provider. Exploiting various electricity prices at different locations, the cloud provider minimizes the total energy cost by assigning users' requests to different data centers. The utility company utilizes the demand response of data centers, and tries to achieve power load balancing by altering the electricity consumption of data centers through dynamic pricing.

The main contributions of this paper are as follows:

- *Data center and smart grid interaction*: To the best of our knowledge, this is the first paper that studies the interactions between smart grid and data centers by considering the active decisions on both sides. In particular, how does the utility company properly incentivize data
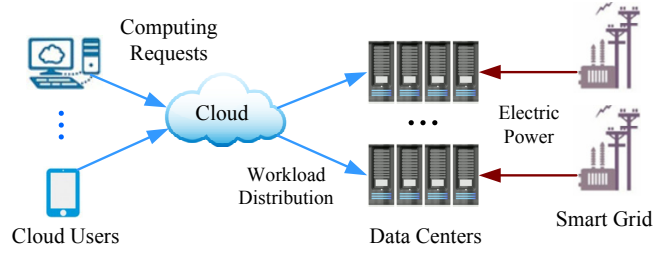

Fig. 1: Smart grid and data center interaction.

centers to provide demand response services toward a reliable power grid?

- *Modeling and solution methods*: We formulate the interactions between smart grid and data centers as a two-stage price optimization problem. In its original form, this problem cannot be solved by standard convex programming techniques. Therefore, we reformulate the problem as a mixed integer quadratic program, and design a customized branch-and-bound algorithm to attain the globally optimal solution. We also design a low-complexity descent algorithm to attain a close-to-optimal solution.
- *Performance benchmarks*: To help characterizing the optimal solution of the two-stage price optimization problem, we construct two single-level optimization problems, namely an *Integrated Problem* and a *Restricted Problem*, which correspond to the performance upper and lower bounds of the two-stage price optimization problem.
- *Case studies and implications*: Our proposed method can not only balance the power load for smart grid but also reduce total energy cost for data centers, hence achieving a *win-win* result.

The remainder of the this paper is organized as follows. We review the related work in Section II. After that, we formulate the system model as a two-stage price optimization problem in Section III. In Section IV, we study two benchmark problems to provide performance bounds for the formulated two-stage price optimization problem. In Section V, we analyze the solution of the two-stage price optimization problem, design a branch-and-bound algorithm to yield the global optimum, and propose an alternative heuristic algorithm to solve the sub-optimal solution. In Section VI, we analyze the worst-case performance by considering the prediction error in background power load. Performance of the proposed scheme is evaluated in Section VII. This paper is concluded in Section VIII.

II. RELATED WORK AND MOTIVATION

A. Literature Review

There are many existing research results on managing data center's workload to reduce energy cost, such as those studying the energy cost minimization problem with multi-electricity-market environment [4], green renewable generators [5], online optimization [6], service level agreements [7], and deregulated electricity price [8]. Zhang *et al.* [9] designed a Vickrey-Clarke-Groves auction mechanism, in which tenants of data

centers voluntarily bid for emergency demand response. However, these results did not consider the active response by the utility companies, nor did they consider how the data centers' demand response may bring large load fluctuations across different locations over time. This motivates us to study the interactions between smart grid and geographically dispersed data centers, and examine how smart grid can properly incentivize data centers through dynamic pricing to improve the grid reliability.

There has been a large body of research on demand response of strategic energy consumers [13]–[16]. For example, in [13], Mohsenian-Rad and Leon-Garcia suggested scheduling household devices based on the predicted prices to minimize the electricity cost. In [14], Nguyen *et al.* proposed a game theoretic model, in which an electricity provider dynamically updates the energy prices to reduce the peak load, by considering the load profiles of users. In [15], Li *et al.* studied demand response based on utility maximization, and proposed a distributed algorithm to compute optimal prices and power schedules. In [16], Wong *et al.* designed a time-dependent price to incentivize users to shift power load so as to relieve stress during peak hours.

B. Motivation

Different from traditional residential or industrial consumers, data centers are special electricity consumers. This is not only because of their enormous energy consumption, but also because of flexibility of energy consumptions over multiple locations. The previous studies in [4]–[9] mainly focused on the workload distribution from the perspective of data centers. As reported in [10], such workload distribution of data centers has great impact on power load balancing in the smart grid.

In the power system, the utility company is responsible for supplying power to meet the demand, and for maintaining the safe operation of the smart grid system. The utility company can utilize the demand response of data centers to manage their energy consumption. However, most of the existing demand response programs focused on the time flexibility of residential demands, without considering the demand side management over multiple locations. The latter is difficult to do for residential demands, but is very suitable in the case of geographically dispersed data centers.[1] This motivates us to design the dynamic pricing incentive mechanism from the grid operator's point of view, in order to incentivize the proper demand response from multiple geographically dispersed data centers. Tran *et al.* [17] studied demand response of data centers in a multi-utilities environment, and modeled the interactions between utilities and data centers as a Stackelberg game. Different from [17], we study the interaction between one utility company and one cloud provider (with multiple data centers) as a bi-level optimization problem, propose two
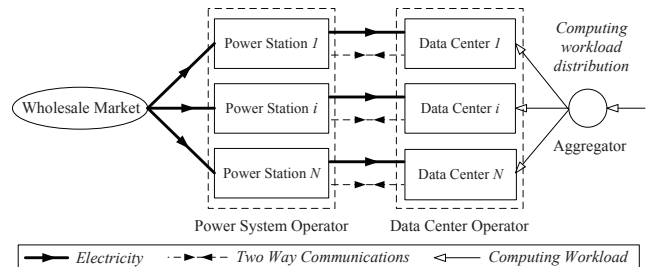

Fig. 2: The architecture for data center demand response.

benchmark problems to estimate the performance bounds, and propose two algorithms to solve the optimal prices and close-to-optimal prices, respectively.

III. SYSTEM MODEL

We consider a discrete time model $t \in \mathcal{T} = \{1, ..., T\}$, where the length of a time slot matches the time-scale at which the workload allocation decisions and dynamic pricing decisions are updated, *e.g.* once an hour [4]. Let $\mathcal{N} = \{1, ..., N\}$ denote the set of geographically dispersed data centers, where each data center $i \in \mathcal{N}$ has M_i homogeneous servers, and has the same function in terms of supporting various kinds of applications (e.g., Internet services, image processing). As we will explain later, not all servers are turned on during each time slot.

Fig. 2 illustrates the system architecture of data centers and smart grid. We assume that a group of geographically dispersed data centers are operated by a single cloud provider, and there is a traffic aggregator (*e.g.*, a front-end portal server) responsible for distributing the total incoming computing workload L^t within time slot t to data centers in different regions [4]. Each data center is powered by a dedicated power substation in the power grid, and all the substations are operated by the same utility company.[2] In each time slot t, we model the interactions between utility company and data centers in two stages. In Stage 1, the utility company sets a billing reference s_i^t, which determines the electricity tariff as we will explain later for each data center i to balance the load on the power grid. In Stage 2, we assume that the data centers can predict the workload accurately at the beginning of each time slot. Then data centers cooperate with each other (as they belong to the same cloud operator) so as to minimize the total energy cost by determining the computing workload allocation λ_i^t and the number of active servers x_i^t in each data center i. Next, we discuss these decisions in details.

A. Stage 2: Data Center's Energy Cost Minimization

First, we consider the Stage-2 problem, where a cloud provider (such as Google) wants to minimize the total energy cost of multiple data centers. In practice, data centers directly

---

[1]The cloud provider owns multiple data centers located in different geographical locations, and thus gains flexibility of power loads over locations via workload assignment over different data centers. As an example, when Google responds to a user's web search query, the corresponding computation can be done in any of the Google's data centers (as long as certain service quality agreement is satisfied).

[2]Many practical examples motivates our assumption of one utility company. For example, Alibaba cloud, a Chinese cloud provider, runs five data centers at different locations in China, and three of which are served by the State Grid Corporation of China. Such scenarios also exist in deregulated electricity markets, such as in California US [18].

negotiate with the utility company regarding the electricity rates [19]. In time slot t, the utility company charges data center i with the following regional electricity price π_i^t per unit of energy:

$$\pi_i^t = \alpha_i^t + \beta_i(e_i^t - s_i^t), \tag{1}$$

where e_i^t is the data center's the electricity consumption, s_i^t is called the billing reference, $\beta_i > 0$ is a sensitivity parameter, and $\alpha_i^t > 0$ denotes the base price, all at location i in time slot t. The dynamic pricing scheme in (1) is motivated by the tiered electricity pricing, which has been widely implemented in various power markets such as the United States, Japan, and China. The key idea of tiered pricing is to set several pricing tiers for the energy consumption, and the unit price per unit of energy increases with the tiers progressively [20]. In (1), the term $\beta_i(e_i^t - s_i^t)$ reflects the difference between electricity consumption e_i^t and the billing reference s_i^t. The unit price π_i^t will be higher than the base price if $e_i^t > s_i^t$.

Next, we discuss the data centers' optimization constraints.

1) Workload constraint: In each time slot t, users' computing requests (workload to the cloud provider) are received by a front-end portal server. Then a total of N data centers should work together to complete the total workload of L^t, with the allocation to data center i as λ_i^t:

$$\sum_{i=1}^{N} \lambda_i^t = L^t, \ \lambda_i^t \geq 0, \ \forall i \in \mathcal{N}, t \in \mathcal{T}. \tag{2}$$

2) QoS (delay) constraint: It is important for data centers to provide QoS guarantees to the users, and QoS can be specified by the service level agreement (SLA) [21]. SLA usually measures the average performance for the operation of a data center during a period of time. We consider both the *transmission delay* (incurred before the request arrives at a data center) and the *queuing delay* (experienced after the request arrives at a data center). We define d_i^t as the transmission delay experienced by a computing request from the aggregator to data center i during time slot t. Notice that d_i^t is usually much less than the length of a time slot. To model the queuing delay, we use queuing theory to estimate the average processing time in data center i when there are x_i^t active servers processing workload λ_i^t with a service rate μ_i per server.[3] Applying the results from M/M/1 queuing theory [5], the average waiting time is approximately $\frac{1}{\mu_i x_i^t - \lambda_i^t}$. To meet the QoS requirement, the total time delay experienced by a computing request should satisfy some delay bound D, which is the maximum waiting time that a request can tolerate. For simplicity, in this paper, we will assume homogeneous requests that have the same delay bound D. Therefore, we have the following QoS constraint

$$d_i^t + \frac{1}{\mu_i x_i^t - \lambda_i^t} \leq D, \ \forall i \in \mathcal{N}, t \in \mathcal{T}, \tag{3}$$

where $\mu_i x_i^t > \lambda_i^t$.

[3]We assume that the servers in the same data center i are homogeneous and have the same service rate μ_i.

3) Server constraint: At each data center i, there are tens of thousands of servers providing cloud computing services to meet users' requests. Let M_i denote the maximum number of available servers. The cloud provider can switch on and off servers to adjust the service time. Since the number of servers is usually large, we can relax the integer constraint on the number of active servers without significantly affecting the optimal result. Therefore, we have the following server constraint[4]

$$0 \leq x_i^t \leq M_i, \ \forall i \in \mathcal{N}, t \in \mathcal{T}. \tag{4}$$

4) Energy consumption constraint: The energy consumption of data centers consists of IT energy consumption (*e.g.*, CPU, memory, and storage) and ancillary energy consumption (*e.g.*, cooling, lighting, and power facility). The quantitative relation between IT energy consumption and ancillary energy consumption is measured by the *power usage efficiency* (PUE) [22], which is defined as the ratio of total energy consumption to IT energy consumption. The energy used by computing equipments is considered to be productive. On the contrary, the energy for ancillary infrastructure (*e.g.*, cooling, lighting, and power facility) is auxiliary. PUE helps us understand the total energy consumption based on the IT energy consumption. Therefore, we can calculate the total energy consumption of a data center using PUE, amount of computing workload, and number of active servers. Precisely, based on the data center power model in [11], we formulate the energy consumption of data center i in time slot t as

$$e_i^t = x_i^t \left(P_{idle} + (R_i - 1)P_{peak}\right) + x_i^t(P_{peak} - P_{idle})\gamma_i^t + \xi_i,$$

where P_{idle} and P_{peak} represent the average idle power and average peak power of a single server, respectively. The power efficiency parameter $R_i > 1$ denotes PUE of data center i. The parameter ξ_i is an empirical constant indicating the base energy consumption of data center i, and γ_i^t denotes the average server utilization of data center i in time slot t.

We substitute the average server utilization $\gamma_i^t = \lambda_i^t/(\mu_i x_i^t)$, and rewrite e_i^t in the following equivalent form:

$$e_i^t = \left(P_{idle} + (R_i - 1)P_{peak}\right) x_i^t + \frac{P_{peak} - P_{idle}}{\mu_i}\lambda_i^t + \xi_i,$$
$$\forall i \in \mathcal{N}, t \in \mathcal{T}, \tag{5}$$

which is an affine function with respect to the number of active servers x_i^t and the computing workload λ_i^t.

Given the operational requirements of the power substation, we limit the maximum power that can be consumed by data center i in time slot t as

$$0 \leq e_i^t \leq Q_i^t, \ \forall i \in \mathcal{N}, t \in \mathcal{T}. \tag{6}$$

where Q_i^t denotes the available power supply to data center i in time slot t.

With the above constraints, we can formulate the cloud provider's energy cost minimization problem in Stage 2.

[4]We set the minimum required number of active servers in each data center as zero. It can also be set as a non positive to reflect operational requirements for the data center, without changing the engineering insights from the analysis.

The objective is to minimize the data centers' total energy cost over all locations and all time slots by choosing the workload allocation λ_i^t and the number of active servers x_i^t for each data center $i \in \mathcal{N}$ and each time $t \in \mathcal{T}$. As the operational constraints (2)-(6) are decoupled across time slots, we formulate the energy cost minimization problem in time slot t as follows:

Stage-2 Problem (PS2): Total Energy Cost Minimization

$$\min_{\boldsymbol{\lambda}^t, \, \boldsymbol{x}^t} \qquad \sum_{i \in \mathcal{N}} \Big(\alpha_i^t + \beta_i(e_i^t - s_i^t) \Big) e_i^t$$

$$\text{subject to} \quad \text{Constraints (2)–(6),}$$

where $\boldsymbol{\lambda}^t = \{\lambda_i^t, \ \forall i \in \mathcal{N}\}$ and $\boldsymbol{x}^t = \{x_i^t, \ \forall i \in \mathcal{N}\}$ denote the workload allocation vector and active server number vector for each time slot $t \in \mathcal{T}$, respectively. The energy cost of data center i is calculated as the product of its energy consumption e_i^t and the corresponding unit price $\alpha_i^t + \beta_i(e_i^t - s_i^t)$.

Note that, the optimal value of workload allocation λ_i^t, number of active servers x_i^t and energy consumption e_i^t in (5) are functions of the billing references $\boldsymbol{s}^t = \{s_i^t, \ \forall i \in \mathcal{N}\}$ in time slot t. Given $\boldsymbol{s}^t$, we can solve Problem **PS2**, and will present the optimal solutions of λ_i^t, x_i^t and e_i^t in Section V.

B. Stage 1: Smart Grid's Power Load Balancing Problem

We are now ready to consider the Stage-1 power load balancing problem for the smart grid. We classify the load into two groups: data centers and others. We focus on the data centers' loads as they have geographical flexibility, and let the latter group as background loads. With the emergence of smart grid communications technologies, it is possible for the utility company to incentivize the data centers to shift loads from heavily loaded regions to lightly loaded regions. In our proposed framework, the smart grid optimizes dynamic tiered prices by setting the billing references $\boldsymbol{s}^t$ in each time slot t to balance power load across geographical locations. To measure the power load levels in different locations, we define the *electric load ratio* in location i and time slot t as

$$r_i^t(\boldsymbol{s}^t) = \frac{e_i^t(\boldsymbol{s}^t) + B_i^t}{C_i}, \tag{7}$$

where B_i^t is the background power load, and C_i is the capacity of power substation i. Note that the load ratio r_i^t is a function of the energy consumption e_i^t, and thus also depends on the billing reference $\boldsymbol{s}^t$ for all locations in time slot t. The utility company aims at balancing the load ratio $r_i^t(\boldsymbol{s}^t)$ at all locations in each time slot.

Let $Q_i^t = C_i - B_i^t$ be the maximum available power supply to data center i in time slot t. Since our study focuses on the demand response of data centers, we denote the aggregate energy usage of all the users other than data centers as the background energy load. We assume that the utility company is able to accurately forecast[5] the background energy load ahead of each time slot [23].

---

[5]We first solve the two-stage problem assuming perfect background load prediction. In section VI, we will further study the impact of prediction error.

Based on the load ratio r_i^t, we define the *electric load index (ELI)* across all locations in time slot t as

$$ELI \triangleq \sum_{i \in \mathcal{N}} \Big(r_i^t(\boldsymbol{s}^t) \Big)^2 C_i, \tag{8}$$

where *ELI* measures the overall load ratio across all locations. Note that electric load ratio r_i^t is a normalized indicator, which does not reflect the importance of those locations with large capacities. Therefore, we introduce the capacities C_i as the weighted coefficients in *ELI*. We can show that minimizing *ELI* with respect to e_i^t yields an equal load ratio across all locations in the ideal case (without considering any constraints):

$$\frac{e_1^t + B_1^t}{C_1} = \cdots = \frac{e_N^t + B_N^t}{C_N},$$

which indicates no overloading problem occurs in any of the locations. Therefore, the system reliability is improved at these locations.

However, such even load distribution may not be achievable in practice, because the energy consumption e_i^t should also satisfy the operational constraints for workload allocation and number of active servers in (2)–(6). Moreover, the cloud provider and the utility company are independent entities. Data centers are operated by the cloud provider, which implies that the energy consumption of data centers cannot be directly controlled by the utility company.

In order to balance the electricity load, in this paper we focus on the scenario where the utility company charges dynamic prices to incentivize users to shift their electricity usage to less loaded locations. To encourage the participation of data centers into the demand response program and prevent the utility company from abusing its market power, constraints should be set to regulate the dynamic prices. In practice, the utility company and data centers usually negotiate with each other and enter into a contract [19] to specify the pricing structure. Based on related studies [24], we set the following constraints for the energy price π_i^t:

$$\underline{\pi}_i^t \leq \alpha_i^t + \beta_i(e_i^t - s_i^t) \leq \overline{\pi}_i^t, \ \forall i \in \mathcal{N}, t \in \mathcal{T}, \tag{9}$$

$$\frac{1}{N} \sum_{i \in \mathcal{N}} \big[\alpha_i^t + \beta_i(e_i^t - s_i^t) \big] \leq \pi_{\max}^t, \ t \in \mathcal{T}, \tag{10}$$

where (9) ensures that the price charged to the data centers is always contained within the range $[\underline{\pi}_i^t, \overline{\pi}_i^t]$. Constraint (10) enforces that the dynamic prices across all locations have an average price ceiling $\pi_{\max}^t$, which is specified by the contract between the utility company and data centers [24]. Constraint (10) can prevent the utility company from charging the maximum possible price in all locations. More precisely, the utility company has to provide lower prices to other locations if it charges a higher prices at some locations, so that (10) can be satisfied. This will give a guarantee to the cloud provider, such that the dynamic price will not arbitrarily increase the energy cost of the data centers.

After the contract terms (*e.g.*, constraints (9) and (10)) are settled, the utility company is responsible of enforcing the

price constraints (9) and (10).[6] We formulate the smart grid's load balancing problem in time slot t as follows:

Stage-1 Problem (PS1): Electric Power Load Balancing

$$\min_{\boldsymbol{s}^t} \quad \sum_{i \in \mathcal{N}} \left(r_i^t(\boldsymbol{s}^t) \right)^2 C_i$$

subject to   Constraints (9) and (10),

where the electric load ratio r_i^t depends on the energy consumption e_i^t, which is the optimal solution of Stage-2 Problem **PS2**.

C. Two-stage Price Optimization Problem

For Problem **PS2**, we can show that constraints (2)–(6) can be equivalently rewritten as constraints of data centers' energy consumption:

$$\sum_{i \in \mathcal{N}} \theta_i e_i^t = E^t, \tag{11}$$

$$\underline{E}_i^t \le e_i^t \le \overline{E}_i^t, \ \forall i \in \mathcal{N}, \tag{12}$$

where θ_i, E^t, $\underline{E}_i^t$ and $\overline{E}_i^t$ are system parameters. Constraint (11) is derived from the workload constraint in (2), which specifies that the summation of θ_i-weighed energy consumption of all the data centers should reach E^t in order to process the total workload L^t. The box constraint (12) sets the energy consumption upper bound $\underline{E}_i^t$ and lower bound $\overline{E}_i^t$ for each data center, to meet all the inequality constrains in (3)–(6). For the proof and detailed representation of the parameters, please see [33].

Using constraints (11) and (12), we can simplify Problem **PS2** into an equivalent energy consumption distribution problem, in which the cloud provider directly decides the energy consumption of data center e_i^t to minimize the energy cost. The equivalent energy consumption distribution problem is presented as follows:

PE2: Equivalent Problem of PS2

$$\min_{\boldsymbol{e}^t} \quad \sum_{i \in \mathcal{N}} \left(\alpha_i^t + \beta_i(e_i^t - s_i^t) \right) e_i^t$$

subject to   Constraints (11) and (12),

where $\boldsymbol{e}^t = \{e_i^t, \ \forall i \in \mathcal{N}\}$. Once the energy consumption e_i^t is determined, we can find the corresponding workload allocation λ_i^t and number of active servers x_i^t.

Fig. 3 shows the relation between the two-stage problems **PS1** and **PE2**, each of which is executed once in each time slot. In Stage 1, at the beginning of each time slot, the utility company sets billing references for data centers to optimize the *ELI* performance. This leads to the tiered price $\pi_i^t = \alpha_i^t + \beta_i(e_i^t - s_i^t)$ for each data center i. In Stage 2, the cloud provider optimizes the energy consumption e_i^t of each data center in order to minimize the total energy consumption $\sum_{i \in \mathcal{N}} \pi_i^t e_i^t$ in time slot t.

---

[6]To enforce constraints (9) and (10), the utility company should carefully determine the dynamic prices and consider the corresponding responses from the data centers, as the price constraints (9) and (10) involve both dynamic prices and energy consumption responses of data centers.
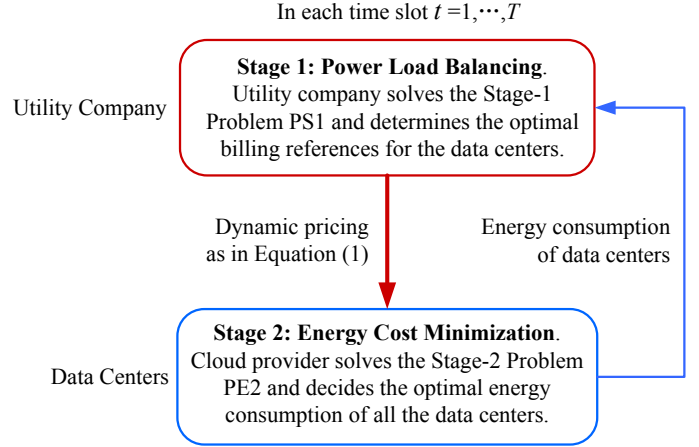
In each time slot $t = 1, \cdots, T$

Fig. 3: Two-stage optimization problem.

The two-stage problem is a challenging optimization problem to solve, due to the coupled variables and constraints. As the utility company aims to balance the electric load across locations, it will consider the response of the cloud provider in Stage 2, when computing the optimal billing references $\boldsymbol{s}^t$ in Stage 1. Before solving the two-stage problem, we will introduce two benchmark problems to bound the optimal solution.

IV. PERFORMANCE BENCHMARKS

The two-stage problem is a quadratic bilevel program with coupled constraints, which is NP-hard in general and cannot be solved effectively by standard convex optimization algorithms. Before proposing solution methods to solve the two-stage problem, we construct two benchmarks, the integrated problem and the restricted problem, to provide lower bound and upper bound of the *ELI* performance, which are helpful in terms of solving the two-stage problem in Section V.

A. The Integrated Problem

We consider the following integrated problem as a benchmark, where the utility company directly decides the optimal workload assignments and the number of active servers for each data center (without the need of dynamic pricing). This will reveal the minimum *ELI* that the system can achieve if the utility company and the data centers fully cooperate with each other.

The integrated problem is formulated as follows.

PI: Integrated Problem

$$\min_{\boldsymbol{\lambda}^t, \ \boldsymbol{x}^t} \quad \sum_{i \in \mathcal{N}} \left(r_i^t \right)^2 C_i$$

subject to   Constraints (2)–(6).

We can see that the objective is consistent with the utility company's objective of load balancing in Problem **PS1**. The constraints are the same in Problem **PS2** for data centers' operation. Problem **PI** is a convex quadratic program, which can be solved by standard convex optimization techniques [25].

Intuitively, compared with the scenario where the utility company incentivizes data centers through dynamic pricing, direct control of data centers' operation would be more efficient in terms of load balancing. This can lead to a lower bound of the *ELI* performance stated in the following proposition.

Proposition 1: The optimal solution of the integrated problem **PI** provides a lower bound of the optimal *ELI* performance of **PS1**.

To prove *Proposition 1*, we need to the show that the feasible set of the integrated problem **PI** is larger than that of the original two-stage optimization problem. In the integrated problem **PI**, the utility company directly controls the workload allocation and the number of servers in the data centers, subject to the data center operation constraints (2)-(6). Whereas in the two-stage problems **PS1** and **PS2**, the utility company aims at indirectly managing data centers' operation in **PS2** through price incentives in **PS1**, subject to both data center operation constraints (2)-(6) and pricing constraints (9)-(10). Intuitively, when the utility company directly controls data centers' operation in **PI**, the decision is more flexible than incentive-based in through the two-stage problems **PS1** and **PS2**. Hence, the performance of **PI** should be better, which means a lower *ELI*. For the detailed proof, see [33].

Note that the *ELI* performance gap between the estimated lower bound and the optimal solution is affected by constraints (9) and (10) in the two-stage problem. For example, enlarging the price range $[\underline{\pi}_i^t, \overline{\pi}_i^t]$ in price constraint (9) can improve the optimal *ELI* performance to be close to the *ELI* lower bound, as the dynamic pricing scheme of the utility company has a larger feasible set.

B. The Restricted Problem

After we provide a lower bound for *ELI* by solving **PI**, we present another benchmark problem namely restricted problem **RS**.

In order to construct the restricted problem, first, we note that in the two-stage problem, different stages have different constraints that cannot be moved across stages. The Stage-1 problem is the upper-level problem, while the Stage-2 problem is the lower-level problem. The constraints of the Stage-1 problem **PS1** also apply to the Stage-2 problem **PE2**, but the operational constraints of data centers in the Stage-2 problem **PE2** only need to be satisfied by the data centers. Intuitively, moving constraints from the Stage-2 problem to the Stage-1 problem shrinks the utility company's action set. Thus, the way we formulate the restricted problem is to move the bounding constraint on energy consumption (12) in **PE2** to the Stage-1 problem **PS1**. Thus we formulate the restricted problem in time slot t with the modified Stage-1 and Stage-2 problems as follows.

RS1: Stage 1 of the Restricted Problem

$$\min_{\boldsymbol{s}^t} \quad \sum_{i \in \mathcal{N}} \left(r_i^t(\boldsymbol{s})\right)^2 C_i$$

subject to   Constraints (9), (10) and (12).

RS2: Stage 2 of the Restricted Problem

$$\min_{\boldsymbol{e}^t} \quad \sum_{i \in \mathcal{N}} \left(\alpha_i^t + \beta_i(e_i^t - s_i^t)\right)e_i^t$$

subject to   Constraints (11).

We use backward induction to solve **RS1** and **RS2**. We first solve Problem **RS2**. Since **RS2** is a convex quadratic program with equality constraints, we obtain the optimal solution in the closed form as

$$e_i^t = \frac{s_i^t}{2} - \frac{\alpha_i^t + \theta_i \sigma^t}{2\beta_i}, \ \forall i \in \mathcal{N}, \tag{13}$$

where σ^t is the Lagrangian multiplier corresponding to the energy equality constraint (11).

Substituting the optimal solution of Problem **RS2** (13) into Problem **RS1**, we have the restricted problem as a single-level optimization problem:

RS: Restricted Problem

$$\min_{\{\boldsymbol{s}^t, \boldsymbol{e}^t, \sigma^t\}} \quad \sum_{i \in \mathcal{N}} \left(r_i^t\right)^2 C_i$$

subject to   Constraints (9)–(13),

which is a convex quadratic program, and can be solved by standard convex programming algorithms [25].

Intuitively, moving constraints (12) from the Stage-2 problem to the Stage-1 problem shrinks the utility company's action set. This can lead to a performance degradation in term of a higher *ELI*, which serves as a upper bound stated in the following proposition.

Proposition 2: The optimal solution of the restricted problem **RS** provides an upper bound of the optimal *ELI* performance of **PS1**.

To prove *Proposition 2*, we need to the show that the feasible set of the restricted problem **RS** is smaller than that of the original two-stage optimization problem. Notice that the constraints (12) were in **PS2** of the original two-stage problem formulation, but here we move them to the Stage-1 problem **RS1** of the restricted two-stage formulation. Compared with **PS1**, in **RS1** the utility company's pricing decision in the restricted problem is more conservative, because the utility company has to satisfy the additional data center operation constraints (12). Intuitively, the restricted two-stage problem has a smaller feasible set than that of the original two-stage problem. Therefore, the solution that is obtained from the restricted problem provides an upper bound for the original two-stage problem. For the detailed proof, see [33].

Note that the estimation of the upper bound is affected by the parameter configurations in constraints (9), (10) and (12).

V. SOLVING THE ORIGINAL TWO-STAGE PROBLEM

After presenting *ELI* performance upper and lower bounds from the benchmark problems, we next solve the original two-stage problem through backward induction. We first solve the Stage-2 problem **PS2**, where data centers minimize the total energy cost. Then, we design a branch-and-bound algorithm for the Stage-1 problem **PS1** to attain the globally optimal solution.

A. Solving the Stage-2 Problem

In the Stage-2 problem **PS2**, data centers decide the work-load allocation λ_i^t and number of active servers x_i^t at all locations to minimize the total energy cost in each time slot, given the charging reference $\boldsymbol{s}^t$ announced by the utility company ahead of each time slot.

We have reformulated **PS2** as en equivalent problem **PE2** in Section III. As Problem **PE2** is strictly convex, we can compute the optimal solution e_i^{t*} through the Lagrangian dual method. This leads to the following result.

Theorem 1: The unique optimal solution of Problem **PE2** is

$$e_i^{t*}(\boldsymbol{s}^t) = \min \left\{ \max \left\{ \underline{E}_i^t, \frac{s_i^t}{2} - \frac{\alpha_i^t + \theta_i \sigma^t}{2\beta_i} \right\}, \overline{E}_i^t \right\}, \ \forall i \in \mathcal{N}. \tag{14}$$

where $e_i^{t*}(\boldsymbol{s}^t)$ is called the best response of data center i to the billing reference $\boldsymbol{s}^t$, and σ^t is the Lagrangian multiplier corresponding to the equality constraint (11).

Problem **PE2** can be solved by the standard subgradient method with a constant stepsize [25]. For the detailed proof, see [33].

B. Solving the Stage-1 Problem

After solving the Stage-2 problem **PE2**, we obtain the optimal energy consumption of data centers as functions of the given charging references $\boldsymbol{s}^t$. We next solve the Stage-1 problem **PS1**. Under the assumption of complete information, the utility company knows how the data centers will respond to the dynamic prices, and can predict the energy consumptions of data centers given the dynamic prices. Therefore, we can replace Problem **PE2** with its Karush-Kuhn-Tucker (KKT) conditions and transform the two-stage problem to a single-level optimization problem [26] by incorporating the KKT conditions of Problem **PE2** into Problem **PS1**.

Theorem 2: (Reformulation) The Stage-1 problem **PS1** can be written in the following equivalent problem with quadratic objectives, linear constraints, and complementarity constraints, denoted as **PE1**.

PE1: Equivalent Problem of the Two-stage Problem

$$\min_{\{s_i^t, e_i^t, \sigma^t, \underline{\omega}_i^t, \overline{\omega}_i^t\}, i \in \mathcal{N}} \sum_{i \in \mathcal{N}} (r_i^t)^2 C_i$$

subject to

$$\underline{\pi}_i^t \le \alpha_i^t + \beta_i(e_i^t - s_i^t) \le \overline{\pi}_i^t, \ \forall i \in \mathcal{N}, \tag{15}$$

$$\frac{1}{N} \sum_{i \in \mathcal{N}} \left[\alpha_i^t + \beta_i(e_i^t - s_i^t) \right] \le \pi_{\max}^t, \tag{16}$$

$$\alpha_i^t + 2\beta_i e_i^t - \beta_i s_i^t + \theta_i \sigma^t - \underline{\omega}_i^t + \overline{\omega}_i^t = 0, \ \forall i \in \mathcal{N}, \tag{17}$$

$$\underline{\omega}_i^t(\underline{E}_i^t - e_i^t) = 0, \ \forall i \in \mathcal{N}, \tag{18}$$

$$\overline{\omega}_i^t(e_i^t - \overline{E}_i^t) = 0, \ \forall i \in \mathcal{N}, \tag{19}$$

$$\sum_{i \in \mathcal{N}} \theta_i e_i^t = E^t, \tag{20}$$

$$\underline{E}_i^t \le e_i^t \le \overline{E}_i^t, \ \forall i \in \mathcal{N}, \tag{21}$$

$$\underline{\omega}_i^t \ge 0, \ \overline{\omega}_i^t \ge 0, \ \ \forall i \in \mathcal{N}, \tag{22}$$

where (17)-(22) are the KKT conditions of Problem **PE2**, and σ^t, $\underline{\omega}_i^t$, and $\overline{\omega}_i^t$ are the Lagrange multipliers associated with the equality and box constraints of **PE2**. Since Problem **PE2** is strictly convex, the KKT conditions (17)-(22) are necessary and sufficient for the optimal solution of Problem **PE2**.

Problem **PE1** is a quadratic program with nonconvex constraints, which cannot be solved efficiently by standard convex optimization techniques. However, we find that the nonconvexity only comes from the complementarity slackness conditions (18) and (19). We can linearize the complementarity slackness conditions (18) and (19) by introducing binary variables $\underline{z}_i^t \in \{0, 1\}$ and $\overline{z}_i^t \in \{0, 1\}$, and replace (18) and (19) by the following constraints:

$$e_i^t - \underline{E}_i^t \le \underline{z}_i^t K, \ \forall i \in \mathcal{N}, \tag{23}$$

$$\underline{\omega}_i^t \le (1 - \underline{z}_i^t)K, \ \forall i \in \mathcal{N}, \tag{24}$$

$$\overline{E}_i^t - e_i^t \le \overline{z}_i^t K, \ \forall i \in \mathcal{N}, \tag{25}$$

$$\overline{\omega}_i^t \le (1 - \overline{z}_i^t)K, \ \forall i \in \mathcal{N}, \tag{26}$$

where K is a sufficiently large constant. We can show that (18) is equivalent to (23) and (24).

- We first show that if (18) is satisfied, then (23) and (24) are also satisfied. There are three combinations to make (18) be satisfied. 1) When $e_i^t = \underline{E}_i^t$ and $\underline{\omega}_i^t > 0$, we have $\underline{z}_i^t \in [0, 1 - \frac{\underline{\omega}_i^t}{K}]$ from (23) and (24). As $\underline{z}_i^t$ is a binary variable, we obtain that $\underline{z}_i^t = 0$. 2) When $e_i^t > \underline{E}_i^t$ and $\underline{\omega}_i^t = 0$, we obtain that $\underline{z}_i^t = 1$. 3) When $e_i^t = \underline{E}_i^t$ and $\underline{\omega}_i^t = 0$, we obtain that $\underline{z}_i^t \in [0, 1]$, and thus either $\underline{z}_i^t = 0$ or $\underline{z}_i^t = 1$.

- We then show that if (23) and (24) are satisfied, then (18) is also satisfied. We discuss the following two cases by exhausting the choices of the binary variable $\underline{z}_i^t$. 1) When $\underline{z}_i^t = 0$, we have $e_i^t \le \underline{E}_i^t$ from (23). Together with the constraint $e_i^t \ge \underline{E}_i^t$ as in (21), we obtain $e_i^t = \underline{E}_i^t$, and thus (18) is satisfied. 2) When $\underline{z}_i^t = 1$, we have $\underline{\omega}_i^t \le 0$ from (24). Together with the constraint $\underline{\omega}_i^t \ge 0$ as in (22), we have $\underline{\omega}_i^t = 0$, and thus (18) is also satisfied.

Following a similar reasoning, we can show that (25) and (26) can replace (19).

To solve **PE1**, we design a branch-and-bound algorithm [27] to attain the optimal solution. We first relax the binary variables $\{0, 1\}$ to continuous variables within the range $[0, 1]$, and define the following relaxed quadratic problem **PR1**.

PR1: Relaxed Problem of PE1

$$\min \quad \sum_{i \in \mathcal{N}} \left(r_i^t \right)^2 C_i$$

subject to   Constraints $(15) - (17), \ (20) - (26),$
$$0 \le \underline{z}_i^t \le 1, \ \forall i \in \mathcal{N},$$
$$0 \le \overline{z}_i^t \le 1, \ \forall i \in \mathcal{N},$$
Variables: $\{s_i^t, e_i^t, \sigma^t, \underline{\omega}_i^t, \overline{\omega}_i^t, \underline{z}_i^t, \overline{z}_i^t\}, i \in \mathcal{N}.$

Initially, the algorithm takes the optimum of the integrated problem **PI** as the lower bound $\underline{F}$, and the optimum of the restricted problem **RS** as the upper bound $\overline{F}$. Then the algorithm start to solve the relaxed problem **PR1** and builds
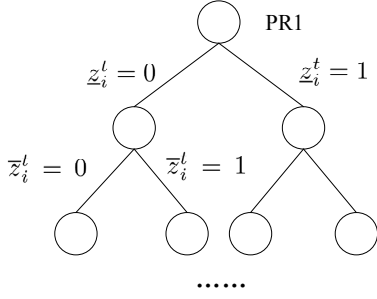
Fig. 4: Branch and bound tree.

the branch and bound tree by splitting the binary variables to enforce the binary variable constraints. Specifically, the algorithm adds the following constraints, $\underline{z}_i^t = 0$ or $\underline{z}_i^t = 1$, to the relaxed problem **PR1**, and derives two new convex quadratic problems (*e.g.*, two first-level children nodes in the branch-and-bound tree shown in Fig. 4). The algorithm continues to expand the tree by adding other constraints $\overline{z}_i^t = 0$ or $\overline{z}_i^t = 1$ until all the binary variables constraints are completely enforced. Meanwhile, the algorithm updates the lower bound $\underline{F}$ after solving each relaxed problem in the children node, and updates the upper bound $\overline{F}$, when a feasible solution with lower optimum is found. The branch-and-bound algorithm terminates at a globally optimal solutions when the lower bound meets the upper bound or all the nodes in the branch and bound tree have been evaluated [27]. In the worst-case, the branch-and-bound algorithm will traverse 2^{2N} nodes.

C. Heuristic Algorithm

The branch-and-bound algorithm in general has a very high worst-case computational complexity, and hence may not be suitable for solving a large-scale load balancing problem. Therefore, we propose a heuristic algorithm to solve the two-stage problem **PS1** and **PS2** for suboptimal solutions. Our heuristic algorithm is designed based on the descent approach, and iteratively reduces the value of *ELI* in Problem **PS1**.

We view the solution of Problem **PE2** as a function of the variables of Problem **PS1**. Observing the best response (14) in Problem **PE2**, we find the following monotonic relation between the charging reference s_i^t, the optimal energy consumption e_i^{t*}, and the unit price $\pi_i^t = \alpha_i^t + \beta_i(e_i^{t*} - s_i^t)$. Specifically, increasing s_i^t leads to increase in e_i^{t*} and σ^t, and decrease in e_j^{t*}, $\forall j \in \mathcal{N} \backslash i$, and all the unit prices π_i^t also decrease. On the contrary, decreasing s_i^t causes decrease in e_i^{t*} and σ^t, and increase in e_j^{t*}, $\forall j \in \mathcal{N} \backslash i$ and all the unit prices π_i^t. Note that minimizing the *ELI* performance (8) yields even load ratio r_i^t across all locations. Thus, we design a descent algorithm to redistribute the total energy consumptions, by decreasing s_i^t in high energy-consumption locations, and increasing s_i^t in low energy-consumption locations. The detailed algorithm is described in Algorithm 1. The utility company and data center iteratively compute the prices and energy consumption. In each iteration, the utility company provides a set of prices, and data centers respond to the prices and report the corresponding schedule of energy consumption (and do not reveal private information such as parameters and constraints). Algorithm 1

reduces *ELI* and its convergence to a feasible and possibly sub-optimal solution is guaranteed since the *ELI* performance is lower bounded by Problem **PI**. For the detailed proof, see [33].

Algorithm 1 Descent algorithm to solve the two-stage problem

1: **Initialization**: In each time slot $t \in \{1, ..., T\}$, set the iteration count $k = 1$, convergence tolerance $\epsilon > 0$, and step-size $\eta(k)$. Initialize the starting point $\boldsymbol{s}^t(k) \triangleq \{s_i^t(k), i \in \mathcal{N}\}$ by solving the restricted problem **RS**, and compute the average load ratio $r_{avg}^t(k) = \frac{\sum_{i \in \mathcal{N}} r_i^t(k)}{N}$.

2: **repeat**

3:     **Step1:** Compute the descent direction $\boldsymbol{g}^t(k)$ for $\boldsymbol{s}^t(k)$: if $r_i^t(k) > r_{avg}^t(k)$, then set $g_i^t(k) = -\frac{\theta_i}{\beta_i}$, $i \in \mathcal{N}$; otherwise, set $g_j^t(k) = \frac{\theta_j}{\beta_j}$, $j \in \mathcal{N} \backslash i$.

4:     **Step2:** Perform the search by using the iterations

5:     $\boldsymbol{s}^t(k+1) = \boldsymbol{s}^t(k) + \eta(k)\boldsymbol{g}^t(k)$;

6:     **Step3:** Given $\boldsymbol{s}^t(k+1)$, solve the optimal energy consumption $e_i^t(k+1)$ according to (14).

7:     **Step4:** Check the feasibility based on (9) and (10). If yes, update $r_{avg}^t(k+1) = \frac{\sum_{i \in \mathcal{N}} r_i^t(k+1)}{N}$. If not,

$$e_i^t(k+1) = e_i^t(k), \ s_i^t(k+1) = s_i^t(k),$$
$$r_{avg}^t(k+1) = r_{avg}^t(k), \ \eta(k+1) = \frac{1}{2}\eta(k).$$

8:     $k \leftarrow k + 1$;

9: **until** the convergence criteria $\|ELI(k) - ELI(k-1)\| \leq \epsilon$ is satisfied;

10: Return the sub-optimal solutions $\hat{\boldsymbol{s}}^t$, $\hat{\boldsymbol{e}}^t$.

11: **end**

VI. IMPACT OF BACKGROUND LOAD PREDICTION ERROR

In Section V, we solved the two-stage problem based on the assumption that the utility company can forecast the background power load B_i^t accurately. In practice, the prediction may have errors and the actual background load may deviate from the predicted values. We define the prediction errors for the background load in location i and time slot t as δ_i^t. Then, we can represent the actual background load $\hat{B}_i^t$ as the summation of predicted value and the prediction error:

$$\hat{B}_i^t = B_i^t + \delta_i^t.$$

Next we use the robust optimization approach [28] to analyze the impact of prediction errors. We assume that the prediction errors are bounded in known uncertainty sets as follows:

$$\Delta_{i,\min}^t \leq \delta_i^t \leq \Delta_{i,\max}^t, \ \forall i \in \mathcal{N}, \tag{27}$$

where $\Delta_{i,\min}^t$ and $\Delta_{i,\max}^t$ denote the lower bound and upper bound of the background load prediction error in location i and time slot t, respectively.

We let $\boldsymbol{\delta}^t = \{\delta_i^t, \ i \in \mathcal{N}\}$ denote the prediction-error vector. Our aim is to maximize the worst-case performance of power

load balancing. We formulate the worst-case performance optimization problem as:

WCP: Worst-case Performance Optimization Problem

$$\min_{\boldsymbol{s}^t} \max_{\boldsymbol{\delta}^t} \quad \sum_{i \in \mathcal{N}} \frac{\left(e_i^t(\boldsymbol{s}^t) + B_i^t + \delta_i^t\right)^2}{C_i}$$

subject to   Constraints (9), (10), (27),

which is a min-max optimization problem.

To solve Problem **WCP**, we first solve the inner *ELI* maximization problem of **WCP** (namely **IWCP**):

$$\max_{\boldsymbol{\delta}^t} \quad \sum_{i \in \mathcal{N}} \frac{\left(e_i^t(\boldsymbol{s}^t) + B_i^t + \delta_i^t\right)^2}{C_i}$$

subject to   Constraints (27),

which corresponds to the worst-case *ELI* performance. We can show that the objective function of Problem **IWCP** is convex in the prediction errors $\boldsymbol{\delta}^t$. Hence, the optimal solution of the **IWCP** problem must reach the boundary of the uncertainty set in (27). Moreover, as the total actual energy consumption $e_i^t(\boldsymbol{s}^t) + B_i^t + \delta_i^t$ is always non-negative, hence the objective function of Problem **IWCP** is a monotonically increasing function in $\boldsymbol{\delta}^t$. Thus we have the following result:

Proposition 3: The optimal solution of Problem **IWCP**, *i.e.*, the worst-case prediction error, lies at the upper bounds of the uncertainty set, *i.e.* $\delta_i^{t,*} = \Delta_{i,\max}^t, \ \forall i \in \mathcal{N}$.

Hence, we substitute the worst-case prediction error $\boldsymbol{\delta}^{t,*} = \{\delta_i^{t,*}, \ \forall i \in \mathcal{N}\}$ into Problem **WCP**, and obtain the following worst-case optimization problem:

$$\min_{\boldsymbol{s}^t} \quad \sum_{i \in \mathcal{N}} \frac{\left(e_i^t(\boldsymbol{s}^t) + B_i^t + \delta_i^{t,*}\right)^2}{C_i}$$

subject to   Constraints (9) and (10),

which solves the optimal billing references $\boldsymbol{s}^t$ to optimize the worst-case performance of *ELI*. Note that the above problem shares the same structure as Problem **PS1**, and thus can be solved by the same methodology presented in Section V.

VII. Simulation Results

In this section, we evaluate our proposed algorithms based on realistic system parameters, and compare the corresponding electric load index and energy cost between the solutions with that of the benchmark problems.

We consider four data centers that are geographically located in four different regions in the United States: New York, Maine, Rhode Island, and Boston. In each location, there is one data center powered by a power station. The numbers of servers in the four locations are 80000, 60000, 60000, and 80000, respectively. The service rates are 4, 3, 4 and 3 requests per server, and each server consumes 200watts electricity in the peak mode and 100watts when it is idle. We set power usage effectiveness as 1.5, 1.2, 1.2 and 1.5 for four data centers, respectively. We took hourly locational marginal prices and demands of the four locations on 4th March 2013 as

the base prices and background power load, according to [29], [30]. The dynamic computing requests are simulated based on the normalized workload trace of Google data centers on 20th December 2013 [31], [32].

A. Performance of the proposed algorithms

We first evaluate the optimal solutions of the two-stage problem and benchmark problems. The upper bound and lower bound for *ELI* over 24 hours are shown in Fig. 5. The optimal solution to the integrated problem provides a lower bound for *ELI*, and the optimal solution to the restricted problem provides an upper bound for *ELI*. Input the upper bound and lower bound into the branch-and-bound algorithm, we can solve the optimal solution to the two-stage problem. The corresponding optimal *ELI* lies in between the upper bound and lower bound, and is very close to the lower bound. Specifically, the optimal *ELI* is on average 1.5% higher than the lower bound across 24 hours.

In Fig. 6, the solid blue curve represents the optimal *ELI* performance of the branch-and-bound algorithm. The dash red curve represents the sub-optimal *ELI* performance obtained by the heuristic algorithm, which is close to the solid blue curve. This suggests that the heuristic algorithm achieves a performance close to the optimal result.
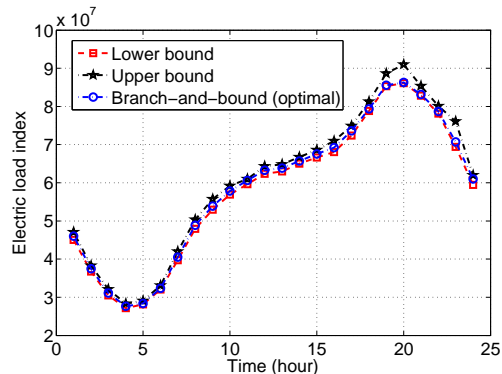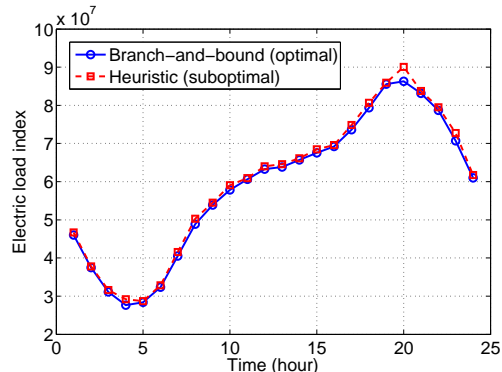

Fig. 5: Upper and lower bounds for ELI.

Fig. 6: Optimal and suboptimal ELI.

B. Effectiveness of optimized dynamic pricing

We demonstrate the effectiveness of our proposed dynamic pricing by comparing to the base pricing benchmark. Fig. 7
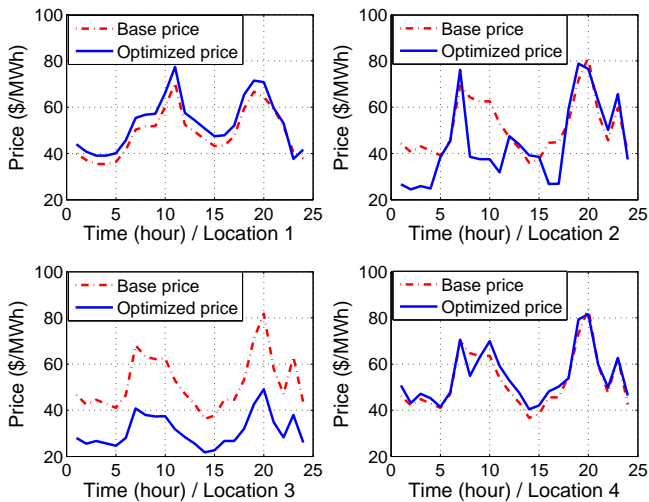
Fig. 7: Base pricing and optimized dynamic pricing.

shows the base prices (dash red curves) and the optimized dynamic prices (solid blue curves) for four data centers, respectively. We can see that the optimized prices may significantly deviate from the base prices. Take hour 10 for example, the optimized prices in location 1 and 4 are higher than the base prices, and in location 2 and 3 are lower than the base prices. This implies that the loads in location 1 and 4 are heavier than those in location 2 and 3. The utility company optimizes the prices for the data centers to re-distribute their energy consumption for load balancing.

In Fig. 8, the dash red curve represents the *ELI* of the base price benchmark, where the data centers are charged based on the fixed base prices. The solid blue curve represents the *ELI* with dynamic pricing, which shows that our proposed dynamic pricing scheme reduces *ELI* by an average of 4% across 24 hours comparing with the base pricing benchmark.

We also evaluate data centers' total energy cost over 24 hours, shown in Fig. 9. The energy cost with dynamic pricing is less than the base pricing benchmark. Specifically, the data centers reduce the total energy cost by an average of 28% across 24 hours, by taking advantage of dynamic prices and reallocating the workload. Fig. 8 and Fig. 9 show that the dynamic interactions between smart grid and data centers bring benefits to both sides and achieve a win-win situation.
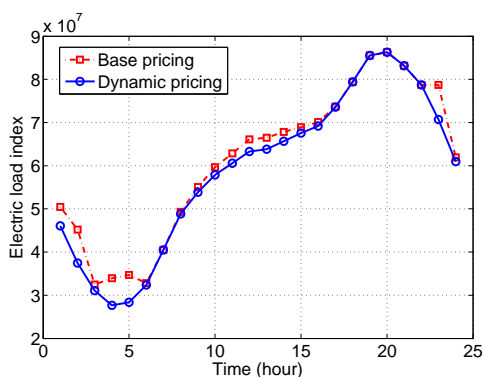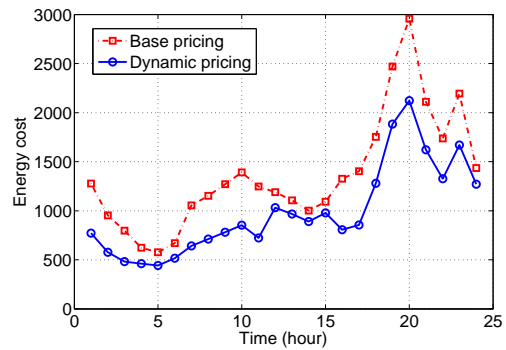

Fig. 8: Comparison of ELI.

Fig. 9: Comparison of energy cost.

We examine the power load distribution within one particular hour (*e.g.*, hour 24), and plot the background power load, power load of data centers, and the total load across four locations. Fig. 10 shows the load distribution with base pricing. The data centers' load (the white bar) is not balanced, since data centers assign workload to the location with the lowest base price as much as possible to minimize the energy cost. The consequence is that the power load is extremely high in the lowest-price location 2, bringing a risk of overloading. Fig. 11 shows the load distribution in the two-stage model with dynamic pricing. We can see the utility company tries to drive the load more evenly across different locations. Therefore, our proposed scheme can effectively improve the reliability of smart grid through re-balancing power load across different locations.
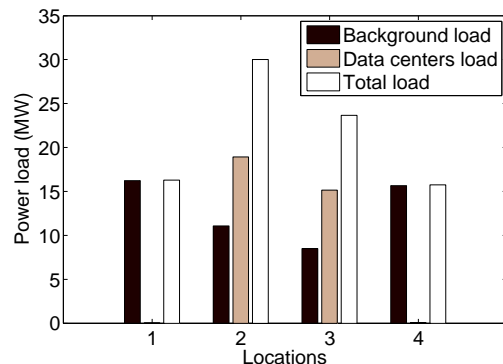

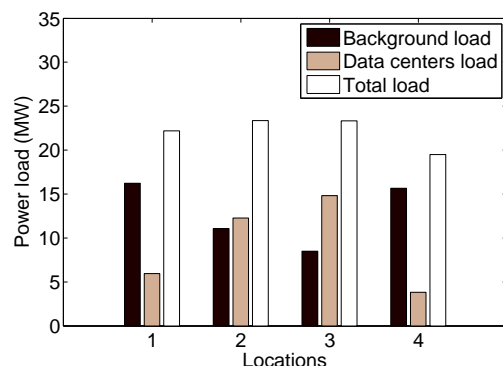Fig. 10: Power load (base pricing).

Fig. 11: Power load (dynamic pricing).

C. Impact of prediction errors

We conduct a case study to show the impact of prediction errors on the *ELI* performance. We set the bounds ($\Delta_{i,\min}^t$ and $\Delta_{i,\max}^t$) of the prediction errors as $\pm 10\%$ of the predicted values $\tilde{B}_i^t$ in location i and time slot t. Solving problem **WCP** in Section VI, we obtain the optimized worst-case *ELI* performance as dash red curve in Fig. 12. We also randomly generate a realization of prediction errors, and compare the *ELI* performance under the scenario with and without considering the prediction errors. If prediction errors are considered when optimizing the Stage-1 problem, the realized *ELI* performance (solid blue curve) can be guaranteed to be better than the worst-case *ELI*. However, if the prediction is assumed to be accurate with zero error (while in reality it is not), then the *ELI* performance (dash black curve) can be even worse than the worst-case benchmark (*e.g.* in the 20th time slot). Therefore, the results demonstrate the effectiveness of our proposed worst-case performance optimization problem, which provides a performance guarantee for *ELI* under prediction errors.
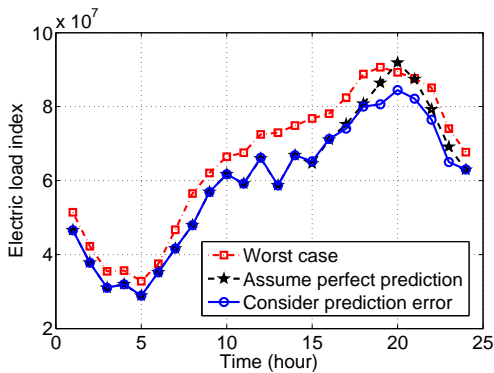

Fig. 12: ELI performance with prediction error.

VIII. CONCLUSIONS

In this paper, we studied the dynamic interactions between smart grid and data centers as a two-stage price optimization problem. To solve the two-stage optimization problem, we reformulated it as a mixed integer quadratic programming problem, and proposed a branch-and-bound algorithm to attain the globally optimal solution, and a low complexity heuristic descent algorithm to yield a close-to-optimal solution. The simulation results showed a win-win solution for both the utility company and data centers.

For future work, we would like to study the interaction between the utility company and data centers with high penetration of renewable energy and under incomplete information. Some cloud provides installed renewable energy facilities to power data centers. How to manage the renewable-powered data centers and what is the impact on the power system are worth of study. The utility company may not able to acquire private information of data-center operation, so how to incentivize data centers with asymmetric information is an interesting and practical problem for future study.

REFERENCES

[1] H. Wang, J. Huang, X. Lin and H. Mohsenian-Rad, "Exploring smart grid and data center interactions for electrical power load balancing," in *ACM Greenmetrics*, Pittsburgh, PA, June 2013.

[2] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, and B. Maggs, "Cutting the electric bill for Internet-scale systems," in *ACM SIGCOMM*, 2009.

[3] M. Pedram, "Energy-efficient datacenters," *IEEE Trans. on Computer Aided Design*, 31(10) 2012.

[4] 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. of IEEE INFOCOM*, 2010.

[5] Z. Liu, M. Lin, A. Wierman, S.H. Low, and L.L.H. Andrew, "Greening geographical load balancing," in *Proc. of ACM SIGMETRICS*, 2011.

[6] M. Lin, Z. Liu, A. Wierman, L.L.H. Andrew, "Online algorithms for geographical load balancing," in *Proc. of IEEE IGCC*, 2012.

[7] M. Ghamkhari and H. Mohsenian-Rad, "Energy and performance management of green data centers: a profit maximization approach," *IEEE Trans. on Smart Grid*, 4(2): 1017-1025, 2013.

[8] 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. on Smart Grid*, 3(4): 1743-1754, 2012.

[9] L. Zhang, S. Ren, C. Wu, and Z. Li, "A truthful incentive mechanism for emergency demand response in colocation data centers," in *Proc. of IEEE INFOCOM*, 2015.

[10] H. Mohsenian-Rad and A. Leon-Garcia, "Energy-information transmission tradeoff in green cloud computing," in *Proc. of IEEE Globecom*, 2010.

[11] H. Mohsenian-Rad, and A. Leon-Garcia, "Coordination of cloud computing and smart power grids," in *Proc. of IEEE SmartGridComm*, 2010.

[12] X. Fang, S. Misra, G. Xue, and D. Yang, "Smart grid-the new and improved power grid: a survey," *IEEE Communications Surveys and Tutorials*, 14: 944-980, 2012.

[13] H. Mohsenian-Rad and A. Leon-Garcia, "Optimal residential load control with price prediction in real-time electricity pricing environments," *IEEE Trans. on Smart Grid*, 1(2): 120-133, 2010.

[14] H.K. Nguyen, J.B. Song, and Z. Han, "Demand side management to reduce peak-to-average ratio using game theory in smart grid," in *Proc. of IEEE INFOCOM Workshops*, 2012.

[15] N. Li, L. Chen, and S.H. Low, "Optimal demand response based on utility maximization in power networks," in *Proc. of IEEE PES General Meeting*, 2011.

[16] C.J. Wong, S. Ha, S. Sen, and M. Chiang, "Optimized day-ahead pricing for the smart grid with device-specific scheduling flexibility," *IEEE Journal on Selected Areas in Communications*, 1075-1085, 2012.

[17] N.H. Tran, D.H. Tran, S. Ren, Z. Han, and C.S. Hong, "How geo-distributed data centers do demand response: a game-theoretic approach," *IEEE Trans. on Smart Grid*, to appear.

[18] http://www.datacentermap.com/usa/california/

[19] H. Xu and B. Li, "Reducing electricity demand charge for data centers with partial execution," in *Proc. of ACM e-Energy*, 2014.

[20] J. Lai, et al, "Evaluation of evolving residential electricity tariffs," Ernest Orlando Lawrence Berkeley National Laboratory, 2011.

[21] L.A. Barroso and U. Holzle, "The case for energy-proportional computing," *Computer*, 40: 3337, 2007.

[22] A. Qureshi, "Power-demand routing in massive geo-distributed systems," Ph.D. dissertation, Massachusetts Institute of Technology, 2010.

[23] A. Motamedi, H. Zareipour and W.D. Rosehart, "Electricity price and demand forecasting in smart grids," *IEEE Trans. on Smart Grid*, 3(2): 664-674, 2012.

[24] M. Zugno, J. Morales, P. Pinson, and H. Madsen, "A bilevel model for electricity retailers participation in a demand response market environment," *Energy Economics*, 36: 182-197, 2013.

[25] S. Boyd and L. Vandenberghe, "Convex optimization," New York, NY, USA: Cambridge University Press, 2004.

[26] Y. Yuan, Z. Li, and K. Ren, "Quantitative analysis of load redistribution attacks in power systems," *IEEE Trans. on Parallel and Distributed Systems*, 23(9): 1731-1738, 2012.

[27] J.F. Bard, "Practical bilevel optimization: algorithms and applications," Kluwer Academic Publishers, 1998.

[28] A. Ben-tal and A. Nemirovski, Selected topics in robust convex optimization, *Mathematical Programming*, 112(1): 125-158, 2008.

[29] http://www.iso-ne.com

[30] http://www.nyiso.com

[31] http://www.google.com/transparencyreport/traffic/data/

[32] Z. Sun, F. Kong, X. Liu, et. al., "Intelligent joint spatio-temporal management of electric vehicle charging and data center power consumption," in *Proc. of IGCC*, 2014.

[33] H. Wang, J. Huang, X. Lin and H. Mohsenian-Rad, "Proactive Demand Response for Data Centers: A Win-Win Solution," [Online] http://arxiv.org/abs/1511.00575

Hamed Mohsenian-Rad (S'04-M'09-SM'14) received his Ph.D. degree in Electrical and Computer Engineering from the University of British Columbia in Vancouver, Canada in 2008. Currently, he is an Assistant Professor of Electrical Engineering at the University of California at Riverside. Dr. Mohsenian-Rad is the recipient of the National Science Foundation (NSF) CAREER Award 2012, the Best Paper Award from the IEEE Power and Energy Society General Meeting 2013, and the Best Paper Award from the IEEE International Conference on Smart Grid Communications 2012. He serves as Editor for the IEEE Trans. on Smart Grid, IEEE Power Engineering Letters, and the IEEE Communications Letters. His research interests include modeling, analysis, and optimization of power systems and smart grids with focus on energy storage, renewable power generation, demand response, cyber-physical security, and large-scale power data analysis.

Hao Wang (S'10) is currently pursuing the Ph.D. degree with the Department of Information Engineering, The Chinese University of Hong Kong. His research interests are in the control and optimization of network systems, with a recent focus on the big data analytics of renewable energy, operations and economics of smart grid.

Jianwei Huang (S'01-M'06-SM'11) is an Associate Professor and Director of the Network Communications and Economics Lab (ncel.ie.cuhk.edu.hk), in the Department of Information Engineering at the Chinese University of Hong Kong. He received the Ph.D. degree from Northwestern University in 2005, and worked as a Postdoc Research Associate in Princeton during 2005-2007. He is the co-recipient of 8 international Best Paper Awards, including IEEE Marconi Prize Paper Award in Wireless Communications in 2011. He has co-authored four books: "Wireless Network Pricing," "Monotonic Optimization in Communication and Networking Systems," "Cognitive Mobile Virtual Network Operator Games, and "Social Cognitive Radio Networks". He has served as an Editor of several top IEEE Communications journals, including JSAC, TWC, and TCCN, and a TPC Chair of many international conferences. He is the Vice Chair of IEEE ComSoc Cognitive Network Technical Committee and the Past Chair of IEEE ComSoc Multimedia Communications Technical Committee. He is a Senior Member and a Distinguished Lecturer of IEEE Communications Society.
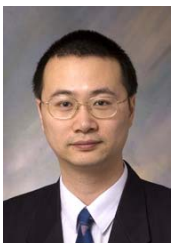
Xiaojun Lin (S'02-M'05-SM'12) received his B.S. from Zhongshan University, Guangzhou, China, in 1994, and his M.S. and Ph.D. degrees from Purdue University, West Lafayette, Indiana, in 2000 and 2005, respectively. He is currently an Associate Professor of Electrical and Computer Engineering at Purdue University.

Dr. Lin's research interests are in the analysis, control and optimization of large communication networks and cyber-physical systems. He received the IEEE INFOCOM 2008 best paper award and 2005 best paper of the year award from *Journal of Communications and Networks*. His paper was also one of two runner-up papers for the best-paper award at IEEE INFOCOM 2005. He received the NSF CAREER award in 2007. He was the Workshop co-chair for IEEE GLOBECOM 2007, the Panel co-chair for WICON 2008, the TPC co-chair for ACM MobiHoc 2009, and the Mini-Conference co-chair for IEEE INFOCOM 2012. He is currently serving as an Associate Editor for IEEE/ACM Transactions on Networking, an Area Editor for (Elsevier) Computer Networks journal, and has served as a Guest Editor for (Elsevier) Ad Hoc Networks journal.