Dynamic Multi-Tenant Coordination for Sustainable Colocation Data Centers

Yuanxiong Guo, *Member, IEEE*, Miao Pan, *Member, IEEE*, Yanmin Gong, *Member, IEEE*, and Yuguang Fang, *Fellow, IEEE*

Abstract—Colocation data centers are an important type of data centers that have some unique challenges in managing their energy consumption. Tenants in a colocation data center usually manage their servers independently without coordination, leading to inefficiency. To address this issue, we propose a formulation of coordinated energy management for colocation data centers. Considering the randomness of workload arrival and electricity cost function, we formulate it as a stochastic optimization problem, and then develop an online algorithm to solve it efficiently. Our algorithm is based on Lyapunov optimization, which only needs to track the instantaneous values of the underlying random factors without requiring any knowledge of the statistics or future information. Moreover, alternating direction method of multipliers (ADMM) is utilized to implement our algorithm in a decentralized way, making it easy to be implemented in practice. We analyze the performance of our online algorithm, proving that it is asymptotically optimal and robust to the statistics of the involved random factors. Moreover, extensive trace-based simulations are conducted to illustrate the effectiveness of our approach.

Index Terms—Colocation data centers, energy-efficiency, green computing, Lyapunov optimization, distributed algorithm

1 Introduction

As the backbone of our modern economy, data centers have been widely deployed, ranging from small server rooms that power small- to medium-sized organizations, to the enterprise data centers that support US corporations, and to the server farms that run cloud computing services. However, as the explosion of digital content, e-commerce, and Internet traffic (also referred as "Big Data"), data centers are also one of the largest and fastest-growing consumers of electricity in US. According to a report from the Natural Resource Defense Council (NRDC) [1], in 2013, U.S. data centers consumed an estimated 91 billion kilowatt-hours of electricity, accounting for more than 2 percent of all U.S. electricity usage. Moreover, their consumptions are expected to grow to 140 billion kilowatt-hours annually by 2,020, resulting in 13 billion dollars in electricity bills and 100 million metric tons of carbon pollution per year.

Given the tremendous amounts of electricity usage and associated carbon emissions, a lot of research [2], [3], [4], [5], [6], [7], [8] have been done to improve the energy-efficiency and sustainability of owner-operated data centers such as Google or Facebook data centers. In this type of data center, data center operators have full control of both IT equipment and facilities, and can directly adjust the CPU speed of servers, turn off/on servers, or schedule workloads to change their energy consumption profiles. Therefore, owner-operated data centers can easily optimize their energy utilization through various power management techniques such as dynamic frequency scaling, dynamic capacity provisioning, workload migration, and advanced cooling (see [9] for a survey of these techniques). Although data centers of this type are commonly known to the public, their consumption is actually very small compared with other types of data center.

On the other hand, another important type of data center, multi-tenant colocation data centers as exemplified by Equinix and TelecityGroup, is largely unexplored. A colocation data center (simply called "colo") rents out spaces for multiple tenants to host their own servers, and the colo operator is mainly responsible for facility support such as power supply, cooling, and security. The colo business mode has become increasingly popular (annual growth rate of 18-20 percent) over the last decade due to its lower operation cost, and is adopted by many companies such as Salesforce, Wikipedia, Akamai, and Amazon. In fact, it is projected that multi-tenant data centers will make up more than one-fourth of all data center capacity by 2016 [1]. Therefore, it is crucial to improve the energy efficiency and sustainability of colocation data centers.

However, besides the challenges faced by nearly all data centers, colocation data centers are subject to unique challenges that can not be solved by existing solutions. In a colo, the colo operator desires reducing its electricity cost but has little control over tenants' servers, while tenants manage their servers independently based on their workload conditions without any coordination with others. Furthermore, tenants in a colocation data center are usually billed for their electricity usage based on their subscribed/reserved peak power at fixed rates no matter how much energy they

- *Y. Guo and Y. Gong are with the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74078. E-mail: {richard.guo, yanmin.gong}@okstate.edu.*
- *M. Pan is with the Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77204. E-mail: mpan2@uh.edu.*
- *Y. Fang is with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611. E-mail: fang@ece.ufl.edu.*

Manuscript received 27 June 2016; revised 12 Jan. 2017; accepted 3 Apr. 2017. Date of publication 25 Apr. 2017; date of current version 4 Sept. 2019. Recommended for acceptance by C. Rong.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TCC.2017.2698033

consume. Therefore, electricity costs in these collocation data centers are usually high.

In this paper, we investigate the coordinated energy management problem for the colo operator and tenants in a colocation data center. Our goal is to improve the energy sustainability of colocation data centers through coordinated management of tenants. By coordinating and jointly optimizing the workload scheduling and server capacity provisioning of tenants in a colocation data center, the total electricity cost of the colocation data center can be minimized. Major challenges include the time-varying and random operation environment such as electricity cost and workload condition, coupling of control decisions across time slots, and the inability of the colo operator to control tenants' servers directly. To resolve them, we first formulate the problem as a stochastic program by explicitly accounting for the uncertainties. Then an online and centralized control algorithm without requiring any a priori information is developed to solve the formulated stochastic program approximately based on the Lyapunov optimization framework. Next, we implement the proposed control algorithm in a distributed way through the alternating direction method of multipliers (ADMM) decomposition so that it can be easily deployed in practice.

In summary, the major contributions of this work are as follows:

- We propose a stochastic formulation for the coordinated energy management problem of colocation data centers. The quality of service for delay-sensitive workloads at each tenant is guaranteed when participating into such coordination. The uncertainties are explicitly modeled, and the coupling of tenants' control actions across time slots because of delay-tolerant workloads is considered.
- We develop a centralized online control algorithm based on Lyapunov optimization to solve the proposed stochastic optimization problem efficiently without requiring any a priori information about the underlying uncertainties. The control algorithm is proved to be asymptotically optimal and offers an explicit trade-off between cost saving and workload delay.
- We implement the proposed control algorithm in a distributed way based on ADMM so that it can be easily deployed in practice. We show the colo operator can indirectly control the tenants to maximize the total benefits obtained through coordination.
- We conduct extensive performance evaluations based on real-world traces. We show that the proposed approach can achieve substantial cost savings for both tenants and data center operator compared with the current practice.

The remainder of this paper is organized as follows. We first present the related work in Section 2. Then, in Section 3 we describe the models for both tenants and the data center operator in a colocation data center, and formulate an optimization problem to maximize their joint benefits. We next present the online control algorithm and describe the decentralized implementation to solve the formulated optimization problem in Sections 4 and 5, respectively. After that, we analyze the performance of our algorithm in Section 6 and present the simulation results of our approach based in Section 7. Finally, we conclude our paper in Section 8.

2 Related Work

Data center power management has attracted a lot of attention in the past decade. Several power management techniques in data centers have been developed so far. Dynamic capacity provisioning [6] has been well investigated to dynamically adjust the number of active servers to match the current workload and deployed in practice (e.g., Facebook's Autoscale [10]). Considering the geographic diversity of distributed data centers, geographical load balancing has been proposed to minimize the energy costs or environmental impacts [2], [3], [5], [11], [12]. Workload scheduling [4], [6], [13] is also useful for solving the energy problem in data centers. Moreover, energy storage has been used in data centers to reduce peak demand, minimize energy cost, or facilitate the integration of renewable energy [7], [8], [14], [15]. Instead of reducing the impacts of huge energy consumption, some recent research activities [16], [17] have focused on utilizing the flexibility of data center energy usage to provide demand response resources to power grids. However, the techniques proposed in the above studies are mostly designed for owner-operated data centers and cannot be directly applied to colocation data centers, where the colo operator lacks the control over tenants' servers.

Colocation data centers have recently attracted increasing attention in literature. One stream of research focuses on incentive mechanisms design for tenants to reduce load when receiving a demand response request. The problem is first considered in [18], which proposes a heuristic mechanism to incentivize tenants' load reduction. No strategic behaviors of tenants are considered. Zhang et al. [19] propose a VCG-type reverse auction mechanism for mandatory emergency demand response (EDR) that is approximately truthful and enforces tenants to reveal their private information. Chen et al. [20] design an incentive mechanism based on parameterized supply function bidding, which is simple and applicable to both mandatory and voluntary EDR. Game theory is used in [21] to minimize the social cost of a colocation data center during mandatory and emergency DER events. Another stream of research focuses on the optimal coordination of tenants to minimize the operating costs (monetary or environmental) of the colo. Islam et al. [22] proposes a bidding scheme for tenants to minimize the carbon footprint in a colocation date center. In [23], an online heuristic algorithm is proposed to optimize the reward rates offered to tenants for cost savings. In the preliminary work [24], we propose a static distributed algorithm to coordinate tenants in a single time slot without considering any uncertainties. This paper also focuses on this stream. However, different from previous studies, we consider the dynamic control of tenants with delay-tolerant workloads in multiple time slots. Uncertainties in workload and electricity cost are considered. Moreover, by proposing an online and distributed approach to coordinating tenants' behaviors, our approach is easy to be implemented in practice.

Another related direction is demand side management in smart grids. Li et al. [25] propose a distributed algorithm for the utility company and the customers to joint compute the optimal day-ahead electricity prices under the deterministic setting. Considering the uncertainty of the operation

environment, an online and distributed algorithm is developed in [26] to coordinate the energy utilization of residential households in smart grids. However, since the system models are different, their solution cannot be directly used in solving our problem. Moreover, we adopt ADMM decomposition instead of dual decomposition used in these papers and achieve much higher efficiency in the resulting distributed algorithm.

3 System Modeling and Problem Formulation

We consider a colocation data center with N tenants, operated by a data center operator (DCO). Each tenant $i \in \mathcal{N} = \{1, 2, \ldots, N\}$ manages its own servers and subscribes a certain peak power supply from the DCO based on a long-term contract. The DCO is responsible for managing the data center facility to provide power supply, cooling, and physical security for tenants. We consider a discrete-time system with time denoted by $t = 0, 1, 2, \ldots$, and the duration of each time slot ranges from 15 minutes to one hour.

3.1 Tenants

Without loss of generality, we assume that each tenant $i \in \mathcal{N}$ owns M_i homogeneous servers. In general, two types of IT workloads are supported in data centers: Delay-sensitive interactive applications such as Internet services and online gaming, and delay-tolerant batch applications such as scientific applications and financial analysis. Delay-sensitive workloads have strict requirements on the response time (usually in the order of ms), while delay-tolerant workloads can be scheduled to run any time as long as they are finished before some deadlines (e.g., several hours to multiple days). The flexibility of delay-tolerant workloads makes it possible to actively manage data center power consumption. Although IT workloads may require multiple IT resources, we assume that workloads are computation-intensive and the CPU resource is the bottleneck resource.

For delay-sensitive workload of tenant i, the arrival rate at time t is $\lambda_i(t)$, the mean service rate per server is μ_i, and the maximum response time indicated by the SLA is rt_i. To characterize the delay performance, we adopt the M/G/1/PS queueing model to analyze the workload serving process. Denote by $a_i(t)$ the number of servers allocated to serve the delay-sensitive workload. In order to meet the response time requirement, we have

$$\frac{1}{\mu_i - \lambda_i(t)/a_i(t)} \leq rt_i. \tag{1}$$

For delay-tolerant workload of tenant i, we denote its IT resource demand at time t as $w_i(t)$, which is random and has a maximum value $\overline{w}_i$, i.e., $0 \leq w_i(t) \leq \overline{w}_i, \forall t$. Due to its delay-tolerant nature, we assume that the delay-tolerant workload is buffered first in a queue before being served. Let $Q_i(t)$ be the amount of the unfinished delay-tolerant workload at the beginning of time slot t and $b_i(t)$ be the number of servers allocated to serve the delay-tolerant workload at time t. The delay-tolerant workload queue evolves over time as follows:

$$Q_i(t+1) = [Q_i(t) - b_i(t) f_i + w_i(t)]^+, \tag{2}$$

where f_i denotes the IT resource provided by a single server during one time slot, and the operator $[x]^+ := \max\{x, 0\}$. To ensure that delay-tolerant workload is not delayed for an arbitrarily long time, we control the system so that the queues in the system are stabilized according to the following definition

$$\overline{Q} := \limsup_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{i=1}^{N} \mathbb{E}\{Q_i(t)\} < \infty. \tag{3}$$

The average power consumption of a server p_i associated with tenant i is often described by a linear function [4]

$$p_i(u) = \alpha_i^0 + (\alpha_i^1 - \alpha_i^0) u, \tag{4}$$

where α_i^0 is its power consumption at idle status, α_i^1 is its power consumption at fully utilized status, and u denotes its average CPU utilization level. According to the M/G/1/PS model, the average CPU utilization level of servers allocated for delay-sensitive workload is $\lambda_i(t)/(a_i(t)\mu_i)$. For the delay-tolerant workload, the allocated servers are fully utilized due to its batch processing nature. Using the above models, the power consumption $d_i(t)$ for tenant i in serving workloads at time t can be calculated as

$$\begin{aligned} d_i(t) &= a_i(t)\left(\alpha_i^0 + \left(\alpha_i^1 - \alpha_i^0\right)\frac{\lambda_i(t)}{a_i(t)\mu_i}\right) + \alpha_i^1 b_i(t) \\ &= \alpha_i^0 a_i(t) + \alpha_i^1 b_i(t) + (\alpha_i^1 - \alpha_i^0)\lambda_i(t)/\mu_i. \end{aligned} \tag{5}$$

Various power management techniques exist for reducing tenants' server energy consumption, such as DVFS and geographical load balancing. Here, we assume that tenant i manages its energy consumption by turning off all unused servers after choosing $a_i(t)$ and $b_i(t)$. Hence the number of servers $m_i(t)$ that are switched off at time t for tenant i is stated as

$$m_i(t) = M_i - a_i(t) - b_i(t), \forall i, t. \tag{6}$$

Note that in current practice, tenant i will turn on all M_i servers irrespective of the workload condition because it does not have any incentive to turn off any of them.

However, switching off servers may result in performance degradation or inconvenience [23]. Let $U_i(m_i(t))$ denote the additional cost incurred when turning off m_i servers at time t for tenant i compared to the case of keeping all servers active. The load curtailment cost function $U_i(\cdot)$ can take various forms depending on different goals of tenant i. For instance, it may include switching cost and delay cost [18]. Here, we only assume that the cost function $U_i(\cdot)$ is non-negative, non-decreasing, convex, and has $U_i(0) = 0$ similar to previous work [16], [20].

3.2 Colocation Data Center Operator

The DCO is responsible for providing reliable power supply and cooling to tenants' servers hosted in the colocation data center. The electricity may be generated by on-site renewable or conventional generation, purchased from electricity market, or both. We capture the time-varying and random electricity cost of the DCO through a generic function $C_t(x) := C(x; \omega(t))$, where x denotes the data center power demand and $\omega(t)$ denotes the time dependent randomness factors affecting the electricity cost such as on-site renewable generation or time-varying electricity price. We assume that this electricity cost function $C_t(x)$ at each time slot t for

any realization of $\omega(t)$ is convex, non-negative, and non-decreasing with respect to the total power demand x drawn from the electricity market. Typical examples include quadratic or piece-wise linear forms as proposed in [27].

In a colocation data center, the overall power consumption consists of two parts: IT power and non-IT power. The IT power is the total power consumed by the servers of all tenants. Given the model (5), the IT power consumption is $\sum_{i=1}^{N} d_i(t)$. On the other hand, data centers have a large portion of power consumption from non-IT purposes such as cooling and power distribution. To capture this aspect, we use the power usage effectivenss (PUE) factor β defined as the ratio of the overall power to the IT power consumed by the data center facility. In practice, β ranges from 1.1 to 2.0, depending on factors such as outside temperature and cooling technology in use. Therefore, the overall power consumption for the colocation data center is $\beta \sum_{i=1}^{N} d_i(t)$.

3.3 Social Cost Minimization

In this paper, we consider the setting that the DCO wishes to offer some incentives (e.g., economic rewards) to its tenants so that tenants can reduce their energy consumption to help DCO reducing its electricity cost. Considering the uncertainty in the system, the long-term average performance is of interest in this paper. In particular, the objective of DCO here is to induce tenants' load curtailment in a way that minimizes the average social cost defined as the sum of the total tenant costs due to load curtailment and the DCO's electricity cost over a large time horizon. Note that the term of incentives issued by the DCO to tenants for load curtailment gets cancelled in the definition of the social cost. Hence the problem can be stated as follows: For the dynamic system defined by (2), design a control strategy which, given the past and the present random electricity cost function $C_t(\cdot)$ and delay-sensitive and delay-tolerant workload arrivals, chooses the IT resource allocation decisions $\mathbf{a} := \{a_i(t), \forall i, t\}$, $\mathbf{b} := \{b_i(t), \forall i, t\}$, $\mathbf{m} := \{m_i(t), \forall i, t\}$, and $\mathbf{d} := \{d_i(t), \forall i, t\}$ such that the time-average social cost is minimized while keeping the system stable. This can be formulated as the following stochastic program

$$\min_{\mathbf{a},\mathbf{b},\mathbf{m},\mathbf{d}} \limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\left\{\sum_{i=1}^{N} U_i(m_i(t)) + C_t\left(\beta \sum_{i=1}^{N} d_i(t)\right)\right\} \quad (7a)$$

$$\text{s.t.} \quad m_i(t) = M_i - a_i(t) - b_i(t), \forall i, t \quad (7b)$$

$$a_i(t) \geq \frac{\lambda_i(t)}{\mu_i - 1/rt_i}, \forall i, t \quad (7c)$$

$$\overline{Q} < \infty. \quad (7d)$$

$$a_i(t), m_i(t), b_i(t) \geq 0, \forall i, t \quad (7e)$$

$$d_i(t) = \alpha_i^0 a_i(t) + \alpha_i^1 b_i(t) + \frac{\lambda_i(t)}{\mu_i}(\alpha_i^1 - \alpha_i^0), \forall i, t. \quad (7f)$$

Here the expectation is taken with respect to the random workload arrivals $\lambda_i(t)$, $w_i(t)$ and electricity cost function C_t. Also in the above problem formulation, we make a simplification that $a_i(t)$ and $b_i(t)$ do not need to be integer-valued, which is acceptable and common in literature since the number of servers in a typical data center is quite large [3], [4], [28].

4 Lyapunov-Based Online Algorithm

The challenging of solving the above problem lies in the uncertainty of future workload arrivals and electricity cost function. Moreover, the detailed statistics of these random processes may be unknown. In the following, we design an online algorithm based on Lyapunov optimization [29], which is asymptotic optimal and requires minimum information on the random dynamics in the system.

First, we define the following Lyapunov function

$$L(t) := \frac{1}{2}\sum_{i=1}^{N} Q_i(t)^2. \quad (8)$$

Define the queuing state of the system at time t as $\mathbf{Q}(t) := (Q_i(t), \forall i)$. Then the one-slot conditional Lyapunov drift is stated as

$$\Delta(t) := \mathbb{E}\{L(t+1) - L(t)|\mathbf{Q}(t)\}, \quad (9)$$

where the expectation is taken with respect to the randomness of workload arrivals and electricity cost function, as well as the randomness in choosing control decisions. In order to minimize the cost while stabilizing the system, we add a scaled form of the expected social cost over one time slot to the above drift function to obtain the following *drift-plus-penalty* term

$$\Delta_V(t) := \Delta(t) + V\mathbb{E}\left\{\sum_{i=1}^{N} U_i(m_i(t)) + C_t\left(\beta\sum_{i=1}^{N} d_i(t)\right)|\mathbf{Q}(t)\right\}, \quad (10)$$

where V is a positive control parameter to adjust the trade-off between minimizing social cost and reducing queue length (i.e, reducing workload delay) as explained in detail later. We have the following lemma regarding the *drift-plus-penalty* term:

Lemma 1. *For any feasible control action under constraints (7b), (7c), (7e), and (7f) that can be implemented at time slot t, we have the following inequality*

$$\begin{aligned}\Delta_V(t) \leq B_1 &+ \sum_{i=1}^{N}\mathbb{E}\{Q_i(t)(w_i(t) - b_i(t)f_i)|\mathbf{Q}(t)\} \\ &+ V\mathbb{E}\left\{\sum_{i=1}^{N} U_i(m_i(t)) + C_t\left(\beta\sum_{i=1}^{N} d_i(t)\right)|\mathbf{Q}(t)\right\},\end{aligned} \quad (11)$$

where B_1 is a constant given by the following

$$B_1 := \sum_{i=1}^{N} \frac{\overline{w}_i^2 + (M_i f_i)^2}{2}. \quad (12)$$

Proof. Taking square on both sides of the queuing models (2) and using the fact that $([x]^+)^2 \leq x^2$, we have

$$\begin{aligned}Q_i^2(t+1) &= \left([Q_i(t) - b_i(t)f_i + w_i(t)]^+\right)^2 \\ &\leq Q_i(t)^2 + 2Q_i(t)(w_i(t) - b_i(t)f_i) \\ &\quad + (w_i(t) - b_i(t)f_i)^2.\end{aligned}$$

Rearranging the above inequality and using the facts that $0 \leq w_i(t) \leq \overline{w}_i, \forall i$ and $0 \leq b_i(t) \leq M_i$, we obtain

$$\frac{Q_i^2(t+1) - Q_i^2(t)}{2} \leq Q_i(t)(w_i(t) - b_i(t)f_i) + \frac{\overline{w}_i^2 + (M_i f_i)^2}{2}. \quad (13)$$

By summing over all tenants i, taking the expectation w. r.t. $\mathbf{Q}(t)$ on both sides, and adding penalty term

$$V\mathbb{E}\left\{\sum_{i=1}^{N} U_i(m_i(t)) + C_t\left(\beta\sum_{i=1}^{N} d_i(t)\right)|\mathbf{Q}(t)\right\},$$

we arrive at the upper bound of the *drift-plus-penalty* term as shown in the Lemma. □

Now we present the proposed algorithm as shown in Algorithm 1. The design principle behind our control algorithm is to greedily minimize the R.H.S. of (11) at each time slot.

Algorithm 1. Online Algorithm to Solve Problem (7)

1: The DCO initializes the control parameter $V > 0$, $Q_i(0) \leftarrow 0$, $\forall i$, and $t \leftarrow 0$, and then collects server information $(M_i, f_i, \mu_i, \alpha_i^0, \alpha_i^1, \forall i)$ and service requirement $(rt_i, \forall i)$ from all tenants.
2: **loop**
3: The DCO collects the current state information $\mathbf{Q}(t)$ and $(\lambda_i(t), \forall i)$ from all tenants and observes $C_t(\cdot)$ at time slot t
4: The DCO solves the following optimization problem:

$$\min_{\mathbf{a},\mathbf{b},\mathbf{m},\mathbf{d}} \quad \sum_{i=1}^{N}[VU_i(m_i(t)) - Q_i(t)b_i(t)f_i] + VC_t\left(\beta\sum_{i=1}^{N} d_i(t)\right) \tag{14a}$$

$$\text{s.t.} \quad m_i(t) = M_i - a_i(t) - b_i(t), \forall i \tag{14b}$$

$$a_i(t) \geq \frac{\lambda_i(t)}{\mu_i - 1/rt_i}, \forall i, t \tag{14c}$$

$$a_i(t), b_i(t), m_i(t) \geq 0, \forall i \tag{14d}$$

$$d_i(t) = \alpha_i^0 a_i(t) + \alpha_i^1 b_i(t) + \frac{\lambda_i(t)}{\mu_i}(\alpha_i^1 - \alpha_i^0), \forall i, \tag{14e}$$

and sends the optimal solution $(m_i^*(t), a_i^*(t), b_i^*(t))$ to each tenant i.
5: Each tenant i follows the schedule received from the DCO to do resource allocation and load curtailment, and updates its queue status as $Q_i(t+1) \leftarrow [Q_i(t) - b_i^*(t)f_i + w_i(t)]^+$.
6: $t \leftarrow t + 1$
7: **end loop**

Note that the above algorithm only requires the knowledge of the instantaneous values of system state information and does not require any knowledge of the statistics of the underlying random processes. Moreover, by assuming that both the objective function and the feasible set in Problem (14) are convex, the optimal solution can be efficiently computed by the DCO using the interior-point method [30].

However, in order to solve Problem (14), it requires the DCO to know all the tenant cost functions, queuing status, and all the constraints at each time slot, which may be impractical since they may contain some private information. Moreover, the DCO cannot control the servers of tenants directly. This motivates us to develop a decentralized algorithm that does not need these private information from tenants and is easy to implement in practice. The key idea is to decompose Problem (14) into multiple subproblems that can be solved by each tenant itself independently, possibly under the coordination of DCO.

5 Decentralized Algorithm to Solve Problem (14)

A common approach to develop decentralized algorithms is through dual decomposition with the subgradient dual update. However, this approach requires the functions $U_i(\cdot)$ and $C_t(\cdot)$ to be strictly convex and is often slow. In our setting, it is common for these functions to be in affine forms. Moreover, it is not easy to choose a right step size in subgradient methods. In the following, we develop a decentralized algorithm based on the alternating direction method of multipliers (ADMM) [31], which does not suffer from the aforementioned drawbacks.

5.1 Background on ADMM

The ADMM is a simple but powerful algorithm that is well suited to distributed convex optimization and has been widely used in applied statistics and machine learning [31]. The algorithm solves problems in the following form

$$\begin{aligned} \min \quad & f(x) + g(z) \\ \text{s.t.} \quad & Ax + Bz = c, \end{aligned} \tag{15}$$

with variables $x \in \mathbf{R}^n$ and $z \in \mathbf{R}^m$, where $A \in \mathbf{R}^{p\times n}$, $B \in \mathbf{R}^{p\times m}$, $c \in \mathbf{R}^p$, and $f : \mathbf{R}^n \rightarrow \mathbf{R}$ and $g : \mathbf{R}^m \rightarrow \mathbf{R}$ are convex. Here, the objective function is separable over two sets of variables, x and y.

As with the method of multipliers, we can form the augmented Lagrangian

$$\begin{aligned} \mathcal{L}_\rho(x, z, y) = f(x) + g(z) + y^T(Ax + Bz - c) \\ + (\rho/2)\|Ax + Bz - c\|_2^2, \end{aligned} \tag{16}$$

where $\rho > 0$ is the penalty parameter and y is the dual variable corresponding to the constraint $Ax + Bz = c$. This augmented Lagrangian can be viewed as the unaugmented Lagrangian associated with the problem

$$\begin{aligned} \min \quad & f(x) + g(z) + (\rho/2)\|Ax + Bz - c\|_2^2 \\ \text{s.t.} \quad & Ax + Bz = c. \end{aligned} \tag{17}$$

Note that the above problem is equivalent to Problem (16) since the quadratic penalty term added to the objective function is zero for any feasible solution x and z. The key benefit of including the penalty term is that the dual Problem of (17) is differentiable under mild conditions on f and g. This can greatly improve the convergence property when solving the problem using iterative methods.

ADMM consists of the following iterations

$$x^{k+1} := \underset{x}{\text{argmin}}\ \mathcal{L}_\rho(x, z^k, y^k), \tag{18}$$

$$z^{k+1} := \underset{z}{\text{argmin}}\ \mathcal{L}_\rho(x^{k+1}, z, y^k), \tag{19}$$

$$y^{k+1} := y^k + \rho(Ax^{k+1} + Bz^{k+1} - c), \tag{20}$$

where the step size ρ is simply the penalty parameter. Similar to dual ascent algorithm, it consists of x-minimization step (18), z-minimization step (19), and a dual variable

update (20). However, in ADMM, x and z are updated in an alternating or sequential fashion, which allows for decomposition when f or g are separable.

The convergence of ADMM can be proved under very mild assumptions, which generally hold in practice [32]. Moreover, ADMM converges to modest accuracy, which is sufficient for many applications, within a few tens of iterations in many cases.

5.2 ADMM-Based Algorithm

To solve Problem (14) using ADMM, we observe that the objective function in the new problem is separable over two sets of variables $x := \{\mathbf{a}, \mathbf{b}, \mathbf{m}\}$ and $y := \{\mathbf{d}\}$, and the equality constraints (14e) are the only coupling constraint, which matches the ADMM form. For the sake of simplicity, we omit time index t which will be clear from the context in the rest of this paragraph.

By relaxing the coupling constraints (14e), we formulate the augmented Lagrangian of (14) as

$$\begin{aligned}\mathcal{L}_\rho(x,y,u) = &\sum_{i=1}^{N}[VU_i(m_i) - Q_i b_i f_i] + VC\left(\beta\sum_{i=1}^{N} d_i\right)\\ &+\sum_{i=1}^{N} u_i(d_i - \alpha_i^0 a_i - \alpha_i^1 b_i - P_i)\\ &+\sum_{i=1}^{N}(\rho/2)(d_i - \alpha_i^0 a_i - \alpha_i^1 b_i - P_i)^2,\end{aligned} \tag{21}$$

where $P_i := \lambda_i(\alpha_i^1 - \alpha_i^0)/\mu_i$ does not depend on decision variables and thus can be viewed as a constant for each tenant i, $\rho > 0$ is the augmented Lagrangian parameter, and $(u_i, \forall i)$ are the dual variables corresponding to constraints (14e).

The problem is then solved by updating x, y, and u sequentially. Specifically, at the $(k+1)$th iteration, the x-minimization step involves solving the following problem

$$\begin{aligned}\min_{\mathbf{a},\mathbf{b},\mathbf{m}} \quad &\sum_{i=1}^{N}\Big(VU_i(m_i) - Q_i b_i f_i - u_i^k\big(\alpha_i^0 a_i + \alpha_i^1 b_i\big)\\ &+ (\rho/2)\big(\alpha_i^0 a_i + \alpha_i^1 b_i\big)\big(\alpha_i^0 a_i + \alpha_i^1 b_i - 2d_i^k + 2P_i\big)\Big),\end{aligned} \tag{22}$$

subject to constraints (14b), (14c), and (14d). Note that this problem can be decomposed over tenants since both the objective function and the constraints are separable over i. Moreover, the above problem is convex and can be solved by the interior point method [30].

After obtaining x^{k+1} from the x-minimization step, the y-minimization step involves solving the following problem

$$\begin{aligned}\min_{\mathbf{d}} \; &VC\left(\beta\sum_{i=1}^{N} d_i\right) + (\rho/2)\sum_{i=1}^{N} d_i^2\\ &+\sum_{i=1}^{N} d_i\big(u_i^k - \rho\big(\alpha_i^0 a_i^{k+1} + \alpha_i^1 b_i^{k+1} + P_i\big)\big)\end{aligned} \tag{23}$$

Then, with the optimal x^{k+1} and y^{k+1}, the final step is to update the dual variables

$$u_i^{k+1} := u_i^k + \rho\big(d_i^{k+1} - \alpha_i^0 a_i^{k+1} - \alpha_i^1 b_i^{k+1} - P_i\big). \tag{24}$$

Note that both the x-minimization step and the dual update step can be carried out independently in parallel for each $i \in \mathcal{N}$. The y-minimization step needs to solve an optimization problem with N variables. In the following, we show that we can simplify this step by solving an optimization problem with a single variable.

First, let $\overline{d}$ denote the average of d_i across all $i \in \mathcal{N}$. Problem (23) can be rewritten as

$$\begin{aligned}\min_{\mathbf{d},\overline{d}} \quad &VC\big(\beta N\overline{d}\big) + (\rho/2)\sum_{i=1}^{N} d_i^2\\ &+\sum_{i=1}^{N} d_i\big(u_i^k - \rho\big(\alpha_i^0 a_i^{k+1} + \alpha_i^1 b_i^{k+1} + P_i\big)\big)\end{aligned} \tag{25a}$$

$$\text{s.t.} \quad \overline{d} = (1/N)\sum_{i=1}^{N} d_i. \tag{25b}$$

Note that minimizing over $d_i, \forall i$ with $\overline{d}$ fixed has the solution

$$\begin{aligned}d_i = &\overline{d} - u_i^k/\rho + \alpha_i^0 a_i^{k+1} + \alpha_i^1 b_i^{k+1} + P_i\\ &+ (1/N)\sum_{i=1}^{N}\big(u_i^k/\rho - \alpha_i^0 a_i^{k+1} - \alpha_i^1 b_i^{k+1} - P_i\big).\end{aligned} \tag{26}$$

Therefore, the above problem can be computed by solving the unconstrained optimization problem

$$\begin{aligned}\min_{\overline{d}} \; &VC\big(\beta N\overline{d}\big) + (\rho N/2)\overline{d}^2\\ &+\rho\overline{d}\sum_{i=1}^{N}\big(u_i^k/\rho - \alpha_i^0 a_i^{k+1} - \alpha_i^1 b_i^{k+1} - P_i\big),\end{aligned} \tag{27}$$

and then applying (26). Note that the Problem (27) only contains a single variable and is easy to solve.

Moreover, substituting (26) for d_i^{k+1} in the dual update Equation (24) gives

$$u_i^{k+1} := \rho\left(\overline{d}^{k+1} + \frac{1}{N}\sum_{i=1}^{N}\big((u_i^k/\rho) - \alpha_i^0 a_i^{k+1} - \alpha_i^1 b_i^{k+1} - P_i\big)\right), \tag{28}$$

which does not depend on i. Therefore, the dual variables $u_i^{k+1}, i \in \mathcal{N}$ are all equal and can be replaced by a single dual variable u^{k+1}.

In summary, by substituting u and (26) in the expressions for x-minimization (22), $\overline{d}$-minimization (27), and dual variable update (28), our final algorithm consists of the following iterations

$$\begin{aligned}&(m_i^{k+1}, a_i^{k+1}, b_i^{k+1})\\ &:= \arg\min_{x_i \in \mathcal{X}_i}\Big(VU_i(m_i) - Q_i b_i f_i + \frac{\rho}{2}(\alpha_i^0 a_i + \alpha_i^1 b_i)^2\\ &\quad - \rho\big(\alpha_i^0 a_i + \alpha_i^1 b_i\big)\Big(u^k/\rho + \alpha_i^0 a_i^k + \alpha_i^1 b_i^k + \overline{d}^k\Big)\\ &\quad + (1/N)\rho\big(\alpha_i^0 a_i + \alpha_i^1 b_i\big)\sum_{i=1}^{N}\big(\alpha_i^0 a_i^k + \alpha_i^1 b_i^k + P_i\big)\Big)\end{aligned} \tag{29}$$

$$\begin{aligned}\overline{d}^{k+1} := \operatorname*{argmin}_{\overline{d}}\Big(&VC\big(\beta N\overline{d}\big) + u^k N\overline{d} + (\rho N/2)\overline{d}^2\\ &- \rho\overline{d}\sum_{i=1}^{N}\big(\alpha_i^0 a_i^{k+1} + \alpha_i^1 b_i^{k+1} + P_i\big)\Big)\end{aligned} \tag{30}$$

$$u^{k+1} := u^k + \rho\left(\overline{d}^{k+1} - \frac{1}{N}\sum_{i=1}^{N}\left(\alpha_i^0 a_i^{k+1} + \alpha_i^1 b_i^{k+1} + P_i\right)\right) \quad (31)$$

where $\mathcal{X}_i$ is the feasible region defined by constraints (14b), (14c), and (14d) for tenant i.

Algorithm 2 describes the entire procedures of solving our problem using the ADMM method.

Algorithm 2. Decentralized Algorithm to Solve Problem (14)

1: The DCO initializes $(1/N)\sum_{i=1}^{N}\left(\alpha_i^0 a_i^k + \alpha_i^1 b_i^k + P_i\right) \leftarrow 0$, $\overline{d}^0 \leftarrow 0$, $u^0 \leftarrow 0$, $k \leftarrow 0$, and broadcasts them to all tenants.
2: **repeat**
3: After receiving $(1/N)\sum_{i=1}^{N}\left(\alpha_i^0 a_i^k + \alpha_i^1 b_i^k + P_i\right)$, $\overline{d}^k$, and u^k, each tenant i solves the Problem (29), and sends the optimal solution $\alpha_i^0 m_i^{k+1}$ back to the DCO.
4: After collecting $\alpha_i^0 a_i^{k+1} + \alpha_i^1 b_i^{k+1} + P_i$ from all tenants $i \in \mathcal{N}$ and summing them together to get $\sum_{i=1}^{N}(\alpha_i^0 a_i^{k+1} + \alpha_i^1 b_i^{k+1} + P_i)$, the DCO solves the Problem (30) to obtain $\overline{d}^{k+1}$. Then it updates the dual variable u^{k+1} according to (31). It then broadcasts $(1/N)\sum_{i=1}^{N}(\alpha_i^0 a_i^{k+1} + \alpha_i^1 b_i^{k+1} + P_i)$, $\overline{d}^{k+1}$, u^{k+1} to all tenants.
5: $k \leftarrow k+1$
6: **until** Convergence criteria is met

Intuitively, our algorithm works in the following way. The dual variable u^k acts as the control price [30] the DCO offers to tenants for coordination. Our algorithm first optimizes workload schedules for tenants given the control price u^k. It then optimizes the average power consumption from all tenants given the previously computed schedules. The dual update chooses the reward price u^{k+1} to ensure that these two sets of variables converge to the same optimal workload schedules.

5.3 Case Study

In this section, we provide a case study of our decentralized algorithm with some cost functions proposed in the literature.

A widely-used electricity cost function for data centers is in the form of demand-responsive electricity price [15], [27], i.e., the electricity price charged to a data center is given as

$$\pi(e_d) = \begin{cases} p_1(e_d + e_r) + q_1, & \text{if } e_d + e_r \le e_0 \\ p_2(e_d + e_r) + q_2, & \text{if } e_d + e_r > e_0, \end{cases} \quad (32)$$

where $p_2 > p_1 \ge 0$, q_1, q_2, e_0 are parameters for demand-responsive pricing, e_d denotes the energy consumed by our colocation data center, and e_r denotes the energy usage of all other consumers in the local electricity market. Also, this piecewise function is smooth, i.e., $p_1 e_0 + q_1 = p_2 e_0 + q_2$. Note that when the total demand in this local market exceeds a threshold e_0, the electricity price would increase much faster with respect to the total demand. Furthermore, the DCO may install some on-site renewable generators. Assume that the marginal cost of renewable generators is zero and no excess power can be sold back to the electricity market. With total power demand d and renewable power output r, the electricity cost paid by DCO is calculated as $\pi\left([d-r]^+\right) \times [d-r]^+$.

With the above models, the $\overline{d}$-minimization Problem (30) can be transformed into the following form

$$\begin{aligned} \min_{\theta_1,\theta_2,\overline{d}} \quad & V\theta_1 + u^k N\overline{d} + \frac{\rho N}{2}\overline{d}^2 \\ & - \rho\overline{d}\sum_{i=1}^{N}\left(\alpha_i^0 a_i^{k+1} + \alpha_i^1 b_i^{k+1} + P_i\right) \\ \text{s.t.} \quad & \theta_1 \ge (p_1(\theta_2 + e_r) + q_1)\theta_2 \\ & \theta_1 \ge (p_2(\theta_2 + e_r) + q_2)\theta_2 \\ & \theta_2 \ge \beta N\overline{d} - r, \theta_2 \ge 0, \end{aligned}$$

where θ_1 and θ_2 are auxiliary variables. Note that the above problem formulation can be readily solved by softwares such as CVX package [33] in MATLAB.

When only considering the inconvenience cost, the load curtailment cost function $U_i(\cdot)$ takes the following linear form [22]

$$U_i(m_i) = \gamma_i m_i, \quad (33)$$

where $\gamma_i > 0$ is a cost parameter ($/server) to model the possible wear-and-tear cost caused by server power switching as well as the reduced reserved processing capacity to handle sudden workload surge. With this linear cost function, the x-minimization step (29) becomes a quadratic program which can be readily solved by CVX package as well.

6 Performance Analysis

In this section, we present the analytical performance results of our online algorithm proposed in Section 4. For sake of simplicity, We focus on the case where the sequence of vectors $(C_t, \lambda_i(t), w_i(t), \forall i), t = 0, 1, \ldots$ is i.i.d. with an arbitrary distribution function. Note that our results could also be extended to the general case where the sequence of vectors $(C_t, \lambda_i(t), w_i(t), \forall i), t = 0, 1, \ldots$ is a finite-state irreducible and aperiodic Markov chain according to the results from the framework of Lyapunov optimization [29]. Moreover, we conduct our simulations based on real traces in the next section.

Theorem 1. *Suppose that $V > 0$ and $Q_i(0) = 0, \forall i$. When the sequence of vectors $(C_t, \lambda_i(t), w_i(t), \forall i), t = 0, 1, \ldots$ is i.i.d., and the mean workload arrival rates are strictly within the capacity region, i.e., $\exists\ \delta > 0 : (\mathbb{E}\{\lambda_i(t)\} + \delta, \mathbb{E}\{w_i(t)\} + \delta, \forall i) \in \mathbf{\Omega}$, we have the following results under our control algorithm:*

1) *The average queue length of delay-tolerant workloads satisfies*

$$\overline{Q} \le \frac{B_1 + VB_2}{\delta}, \quad (34)$$

where B_1 is given by (12) and $B_2 = \sum_{i=1}^{N} U_i(M_i) + \max_t C_t(\beta\sum_{i=1}^{N}\alpha_i^1 M_i)$.

2) *The time-average expected social cost of our algorithm $\overline{g}$ is within a bound B_1/V of the optimal offline value $\overline{g}^*$, i.e.,*

$$\overline{g} \le \overline{g}^* + B_1/V, \quad (35)$$

where $\overline{g}^*$ is the minimum average social cost achieved by the optimal offline algorithm with all future information.

Proof. To prove Theorem 1, we need the following lemma given by the framework of Lyapunov optimization. Its proof is similar to that in [14] and omitted here for brevity.

Lemma 2. *For any mean workload arrival rates $\mathbb{E}\{\lambda_i(t)\} = \overline{\lambda}_i, \forall i$ and $\mathbb{E}\{w_i(t)\} = \overline{w}_i, \forall i$ within a capacity region $\mathbf{\Omega}$, there exists a stationary and randomized control policy that selects feasible control decisions $\hat{a}_i(t)$, $\hat{b}_i(t)$, $\hat{m}_i(t)$, and $\hat{d}_i(t)$ every time slot t purely as a function of current system state $(C_t, \lambda_i(t), w_i(t), \forall i)$ while satisfying the following:*

$$\mathbb{E}\{\hat{b}_i(t) f_i\} = \overline{w}_i, \forall i \tag{36}$$

$$\mathbb{E}\left\{\sum_{i=1}^{N} U_i(\hat{m}_i(t)) + C_t\left(\beta \sum_{i=1}^{N} \hat{d}_i(t)\right)\right\} = \overline{g}^*(\overline{\lambda}_i, \overline{w}_i, \forall i), \tag{37}$$

where $\overline{g}^*(\overline{\lambda}_i, \overline{w}_i, \forall i)$ is the minimum value of social cost that can be achieved with workload arrival rates $(\overline{\lambda}_i, \overline{w}_i, \forall i)$.

Although the above control policy is optimal, deriving it requires the detailed probability distribution functions of all combinations of $(C_t, \lambda_i(t), w_i(t), \forall i)$ and suffers the curse of dimensionality if solved by dynamic programming. In the following, we use the existence of such a policy to obtain some bounds on the performance of our algorithm.

First, we prove the average queue bound (34). Since $(\lambda_i + \delta, \overline{w}_i + \delta, \forall i) \in \mathbf{\Omega}$, according to the above lemma there exists a stationary and randomized control policy that can achieve

$$\mathbb{E}\{\hat{b}_i(t) f_i\} = \overline{w}_i + \delta, \forall i \tag{38}$$

$$\mathbb{E}\left\{\sum_{i=1}^{N} U_i(\hat{m}_i(t)) + C_t\left(\beta \sum_{i=1}^{N} \hat{d}_i(t)\right)\right\} = \overline{g}^*(\overline{\lambda}_i + \delta, \overline{w}_i + \delta, \forall i). \tag{39}$$

Recall that our algorithm greedily minimizes the R.H.S. of the inequality (11) at each time slot t over all feasible control policies including the above stationary and randomized policy. Substituting the results (38) and (39) into the R.H.S. of (11) and using the fact that this policy is independent of queue state $\mathbf{Q}(t)$, we obtain

$$\Delta_V(t) \leq B_1 + V\overline{g}^*(\overline{\lambda}_i + \delta, \overline{w}_i + \delta, \forall i) - \delta \sum_{i=1}^{N} Q_i(t). \tag{40}$$

Using the law of iterative expectation and taking the expectation of both sides, we get

$$\begin{aligned}\mathbb{E}\{L(t+1) - L(t)\} + V\mathbb{E}\left\{\sum_{i=1}^{N} U_i(m_i(t)) + C_t\left(\beta \sum_{i=1}^{N} m_i(t)\right)\right\} \\ \leq B_1 + V\overline{g}^*(\overline{\lambda}_i + \delta, \overline{w}_i + \delta, \forall i) - \delta \sum_{i=1}^{N} \mathbb{E}\{Q_i(t)\}.\end{aligned} \tag{41}$$

By ignoring the penalty term and summing over $t = \{0, 1, \ldots, T-1\}$ of the above inequality, we have

$$\begin{aligned}\mathbb{E}\{L(T) - L(0)\} \leq B_1 T + VT\overline{g}^*(\overline{\lambda}_i + \delta, \overline{w}_i + \delta, \forall i) \\ - \delta \sum_{t=0}^{T-1} \sum_{i=1}^{N} \mathbb{E}\{Q_i(t)\}.\end{aligned} \tag{42}$$

Then, after dividing both sides by T, using the facts that $L(t) \geq 0$ and $L(0)$ is finite, and taking $T \to \infty$, we arrive at the following:

$$\limsup_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{i=1}^{N} \mathbb{E}\{Q_i(t)\} \leq \frac{B_1 + V\overline{g}^*(\overline{\lambda}_i + \delta, \overline{w}_i + \delta, \forall i)}{\delta}. \tag{43}$$

Since $\overline{g}$ must be lower than the maximum per-slot social cost $\sum_{i=1}^{N} U_i(M_i) + \max_t C_t(\beta \sum_{i=1}^{N} \alpha_i^1 M_i)$, we have proved (34).

Second, we prove the bound (35) on the average social cost. From (41), we have

$$\begin{aligned}\mathbb{E}\{L(t+1) - L(t)\} + V\mathbb{E}\left\{\sum_{i=1}^{N} U_i(m_i(t)) + C_t\left(\beta \sum_{i=1}^{N} m_i(t)\right)\right\} \\ \leq B_1 + V\overline{g}^*(\overline{\lambda}_i + \delta, \overline{w}_i + \delta, \forall i).\end{aligned} \tag{44}$$

By summing over $t = \{0, 1, \ldots, T-1\}$, dividing both sides by T, and using the facts that $L(t) \geq 0$ and $L(0)$ is finite, we obtain

$$\begin{aligned}\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\left\{\sum_{i=1}^{N} U_i(m_i(t)) + C_t\left(\beta \sum_{i=1}^{N} m_i(t)\right)\right\} \\ \leq \overline{g}^*(\overline{\lambda}_i + \delta, \overline{w}_i + \delta, \forall i) + B_1/V.\end{aligned} \tag{45}$$

Letting $T \to \infty$ and $\delta \to 0$, we arrive at the following performance guarantee based on the Lebesgue's dominated convergence theorem

$$\begin{aligned}\limsup_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\left\{\sum_{i=1}^{N} U_i(m_i(t)) + C_t\left(\beta \sum_{i=1}^{N} m_i(t)\right)\right\} \\ \leq \overline{g}^* + B_1/V.\end{aligned} \tag{46}$$

□

7 Numerical Evaluation

In this section, we conduct trace-based simulations to evaluate the performance of our algorithm in a realistic scenario.

7.1 Simulation Setup

Colocation Data Center Setup. We consider a colocation data center located in Mountain View, California, which consists of ten tenants. Each tenants has 2,000 servers, and each server has an idle and peak power of 150 W and 250 W, respectively. The average PUE of the colo is set to 1.5, i.e., whenever a tenant consumes 1 kWh energy, the corresponding energy consumption at the colo level is 1.5 kWh. Therefore, the peak power consumption of the colo is 7.5 MW.

Renewable Generation. We assume that the DCO is equipped with 2 MW PV panel array. The solar power data are collected from the National Renewable Energy Laboratory [34]. A snapshot of the solar power data over one week is shown in Fig. 1a.

Electricity Cost Function. As shown in [27], by applying mean square error data fitting to the hourly energy demand and electricity price data from January to June, 2012 at this location, the following parameters for demand-responsive electricity price model (32) are obtained: $a_1 = 0.15$ \$/MWh, $b_1 = -15.6$ \$/MWh, $a_2 = 0.98$ \$/MWh, $b_2 = -364.2$ \$/MWh, $e_0 = 420$ MWh. A snapshot of the resulting demand responsive electricity price charged to the colo when other consumers in the electricity market use 415 MWh energy is depicted in Fig. 1b.

Tenant Workload Description. We assume that the average arrival rates for the two types of workloads are equal. The delay-sensitive workload data are collected from MSR

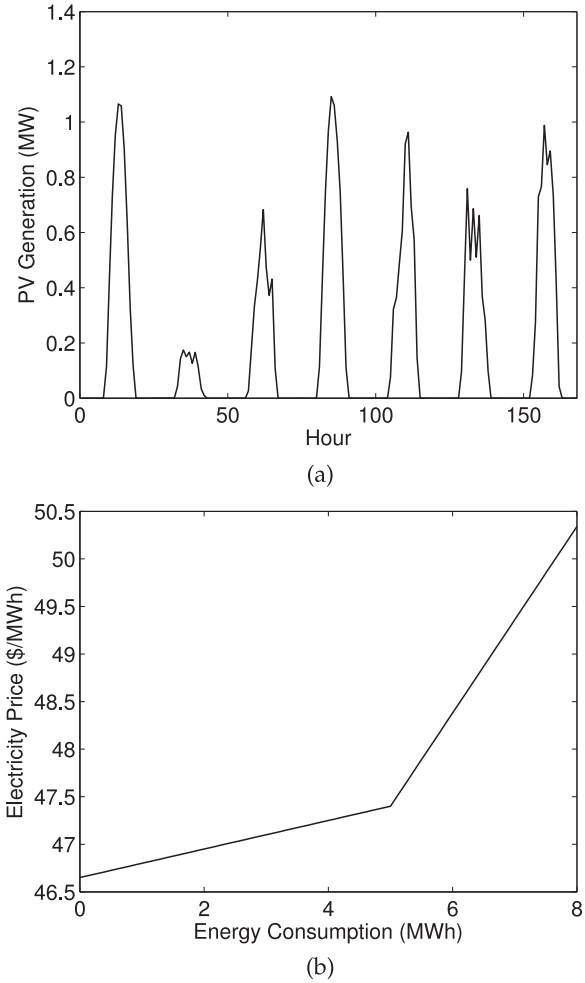


Fig. 1. Simulation data. (a) Solar trace. (b) Demand responsive electricity price when $e_r = 415$ MW.

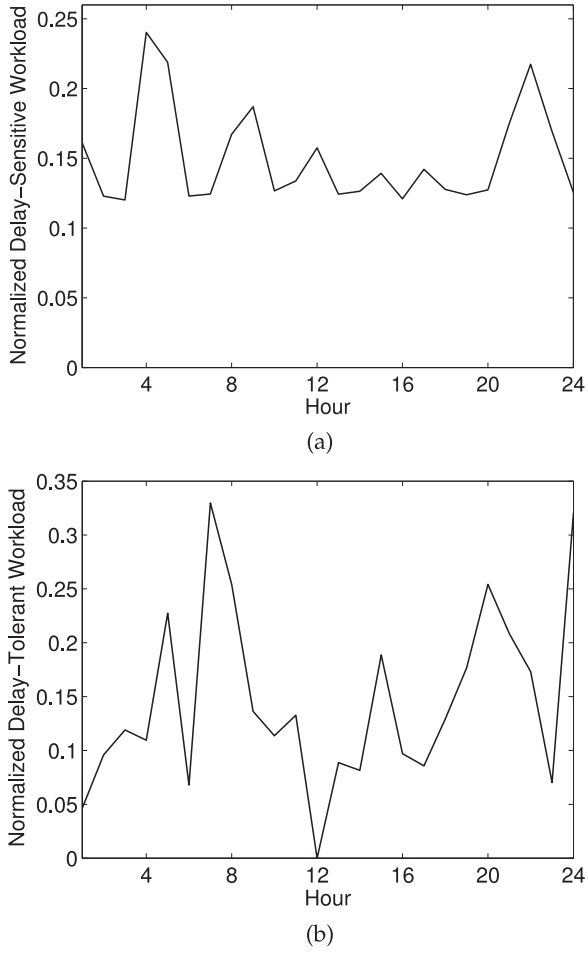


Fig. 2. Workload traces. (a) Delay-sensitive workload. (b) Delay-tolerant workload.

Cambridge [35]. A snapshot of the data over one day is depicted in Fig. 22a, where the workload is normalized with respect to a tenant's service capacity. As with [8], we choose MapReduce [36] which is a popular type of computation-intensive workloads in data centers as the example of delay-tolerant workloads for tenants. The historical Hadoop (an open source implementation of MapReduce) trace on a 600-machine cluster at Facebook [37] is used to calculate the hourly average delay-tolerant workload arrivals. A snapshot of the normalized data over one day is depicted in Fig. 2b.

The service rate of a server is set to 400 requests per second. The average delay requirements for all tenants are set to be no longer than 6 ms. We consider the inconvenience cost resulting from turning off servers as shown in (33). The cost parameter γ_i is set to be uniformly distributed between $0.69 \sim 0.75$ cent per server (i.e., $4.6 \sim 5$ cents per kWh). Note that the values of these parameters enable the tenant to cover the power management cost if housing servers in its own data center as explained in [19]. We set the simulation horizon to ten days with each time slot equal to one hour.

7.2 Simulation Results

Our evaluation results are shown below.

Social Cost. We first compare the total social costs incurred by our algorithm and the current practice without any coordination from tenants, as illustrated in Fig. 3a. We observe that our algorithm can provide cost savings (around 27 percent in average) compared with the current practice when $V = 10^5$. Therefore, it is important for DCO and tenants to collaborate in reducing the social cost.

Impact of Parameter V. Since our algorithm depends on the control parameter V, we compare its average social cost and queue length under different values of V. As shown in Fig. 3b, as we increase V, the average social cost decreases while the average queue length increases (i.e., workload delay becomes larger according to Little's law), and vice versa. This validates the results of Theorem 1. Intuitively, we get more opportunities to reduce cost if workloads are more delay-tolerant.

Algorithm Convergence. Fig. 4 plots the convergence property of our distributed algorithm when executed at one time slot. We observe that our algorithm converges to the optimal solution very fast, usually within 10 iterations, and thus its effectiveness is validated. Note that the social cost achieved by our algorithm can be lower than the optimal cost at the beginning of iterations. The reason is that our algorithm may not yield feasible solution at all iterations (i.e., constraints in (14e) are not always satisfied). However, after a few iterations, by enforcing the regularization terms for coupling constraints, our algorithm would meet all constraints while optimizing the objective function.

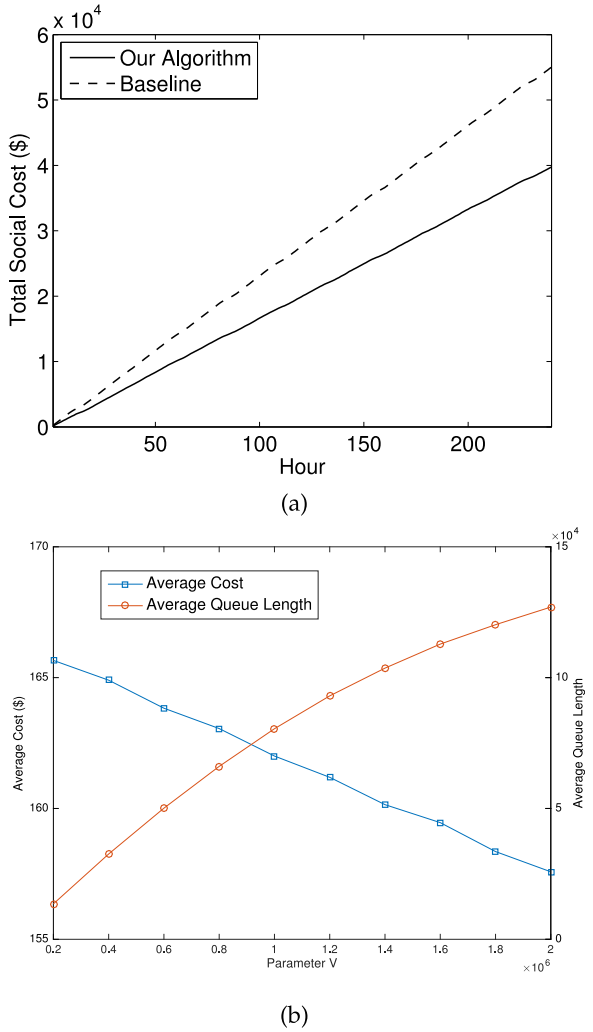


Fig. 3. (a) Comparisons of social cost between our algorithm and the uncoordinated approach with $V = 10^5$. (b) Trade-off between delay and cost in our algorithm with different values of V.

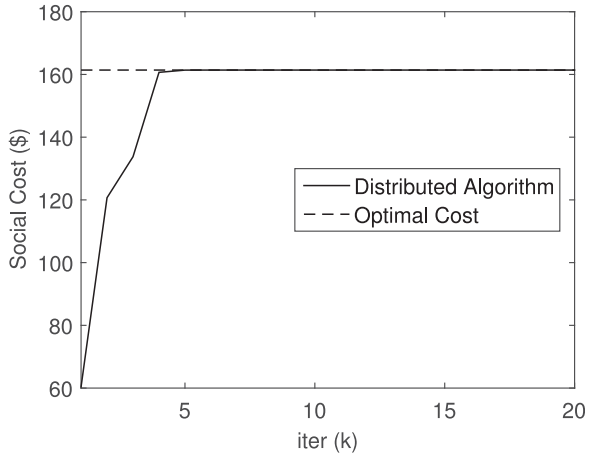


Fig. 4. Convergence of the distributed algorithm.

8 Conclusion

In this paper, we have investigated the problem of coordinated energy management for colocation data centers and formulated it as a stochastic optimization problem. An online and distributed control algorithm based on Lyapunov optimization and ADMM has been proposed to solve the problem efficiently. We have shown the effectiveness of the proposed approach through numerical evaluations based on real-world traces. In the future, we plan to consider the setting of geo-distributed colocation data centers and investigate the multi-tenant coordination issue while considering the geographical load balancing opportunity.

Acknowledgments

This work was partially supported by the U.S. National Science Foundation under Grants CNS-1343361, CNS-1350230 (CAREER), CNS-1343356, CNS-1423165, and CNS-1646607.

References

[1] J. Whitney and P. Delforge, "Data center efficiency assessment—scaling up energy efficiency across the data center industry: Evaluating key drivers and barriers," NRDC, Anthesis, Tech. Rep., 2014, http://www.nrdc.org/energy/files/datacenter-efficiency-assessment-IP.pdf

[2] Z. Liu, M. Lin, A. Wierman, S. Low, and L. Andrew, "Geographical load balancing with renewables," in *ACM GreenMetrics Perform. Eval. Rev.*, vol. 39, pp. 62–66, 2011.

[3] Z. Liu, M. Lin, A. Wierman, S. Low, and L. Andrew, "Greening geographical load balancing," in *Proc. ACM SIGMETRICS Joint Int. Conf. Meas. Modeling Comput. Syst.*, 2011, pp. 233–244.

[4] Z. Liu, et al., "Renewable and cooling aware workload management for sustainable data centers," in *Proc. 12th ACM Sigmetrics/Perform. Joint Int. Conf. Meas. Modeling Comput. Syst.*, 2012, pp. 175–186.

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

[6] M. Lin, A. Wierman, L. L. H. Andrew, and E. Thereska, "Dynamic right-sizing for power-proportional data centers," *IEEE/ACM Trans. Netw.*, vol. 21, no. 5, pp. 1378–1391, Oct. 2013.

[7] Y. Guo and Y. Fang, "Electricity cost saving strategy in data centers by using energy storage," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 6, pp. 1149–1160, Jun. 2013.

[8] Y. Guo, Y. Gong, Y. Fang, P. P. Khargonekar, and X. Geng, "Energy and network aware workload management for sustainable data centers with thermal storage," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 8, pp. 2030–2042, Aug. 2014.

[9] F. Kong and X. Liu, "A survey on green-energy-aware power management for datacenters," *ACM Comput. Surveys*, vol. 47, no. 2, pp. 30:1–30:38, Nov. 2015.

[10] Q. Wu, "Making facebook's software infrastructure more energy efficient with autoscale," Menlo Park, CA, USA: Facebook, 2014.

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

[12] P. X. Gao, A. R. Curtis, B. Wong, and S. Keshav, "It's not easy being green," in *Proc. ACM SIGCOMM Conf. Appl. Technol. Archit. Protocols Comput. Commun.*, 2012, pp. 211–222.

[13] Y. Yao, L. Huang, A. Sharma, L. Golubchik, and M. Neely, "Data centers power reduction: A two time scale approach for delay tolerant workload," in *Proc. IEEE INFOCOM*, 2012, pp. 1431–1439.

[14] R. Urgaonkar, B. Urgaonkary, M. J. Neely, and A. Sivasubramaniam, "Optimal power cost management using stored energy in data centers," in *Proc. ACM SIGMETRICS Joint Int. Conf. Meas. Modeling Comput. Syst.*, 2011, pp. 221–232.

[15] P. Wang, L. Rao, X. Liu, and Y. Qi, "D-Pro: Dynamic data center operations with demand-responseive," *IEEE Trans. Smart Grid*, vol. 3, no. 4, pp. 1743–1754, Dec. 2012.

[16] Z. Liu, I. Liu, S. Low, and A. Wierman, "Pricing data center demand response," in *Proc. ACM Int. Conf. Meas. Modeling Comput. Syst.*, 2014, pp. 111–123.

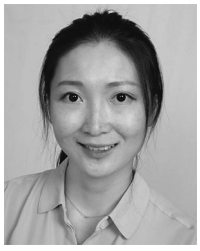
[17] A. Wierman, Z. Liu, I. Liu, and H. Mohsenian-Rad, "Opportunities and challenges for data center demand response," in *Proc. Int. Green Comput. Conf.*, 2014, pp. 1–10.

[18] S. Ren and M. Islam, "Colocation demand response: Why do I turn off my servers," in *Proc. 11th Int. Conf. Autonomic Comput.*, 2014, pp. 201–208.

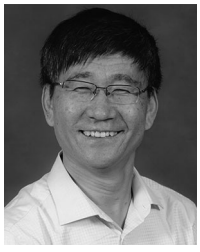
[19] L. Zhang, S. Ren, C. Wu, and Z. Li, "A truthful incentive mechanism for emergency demand response in colocation data centers," in *Proc. IEEE Conf. Comput. Commun.*, 2015, pp. 2632–2640.
[20] N. Chen, X. Ren, S. Ren, and A. Wierman, "Greening multi-tenant data center demand response," in *Proc. SIGMETRICS Perform. Eval. Rev.*, 2015, pp. 36–38.
[21] N. H. Tran, C. T. Do, S. Ren, Z. Han, and C. S. Hong, "Incentive mechanisms for economic and emergency demand responses of colocation datacenters," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 12, pp. 2892–2905, Dec. 2015.
[22] M. Islam, S. Ren, and X. Wang, "Greencolo: A novel incentive mechanism for minimizing carbon footprint in colocation data center," in *Proc. Int. Green Comput. Conf.*, 2014, pp. 1–8.
[23] M. Islam, H. Mahmud, S. Ren, and X. Wang, "Paying to save: Reducing cost of colocation data center via rewards," in *IEEE HPCA*, 2015.
[24] Y. Guo and M. Pan, "Coordinated energy management for colocation data centers in smart grids," in *Proc. IEEE Int. Conf. Smart Grid Commun.*, 2015, pp. 840–845.
[25] N. Li, L. Chen, and S. H. Low, "Optimal demand response based on utility maximization in power networks," in *Proc. IEEE Power Energy Soc. Gen. Meet.*, Jul. 2011, pp. 1–8.
[26] Y. Guo, M. Pan, Y. Fang, and P. P. Khargonekar, "Decentralized coordination of energy utilization for residential housholds in the smart grid," *IEEE Trans. Smart Grid*, vol. 4, no. 3, pp. 1341–1350, Sept. 2013.
[27] A. H. Mahmud and S. Ren, "Online capacity provisioning for carbon-neutral data center with demand-responsive electricity prices," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 41, pp. 26–37, 2013.
[28] M. Lin, A. Wierman, L. L. H. Andrew, and E. Thereska, "Dynamic right-sizing for power-proportional data centers," in *Proc. IEEE/ACM Trans. Netw.*, 2011, pp. 1098–1106.
[29] M. J. Neely, *Stochastic Network Optimization with Application to Communication and Queueing Systems*. San Rafael, CA, USA: Morgan & Claypool Publishers, 2010.
[30] S. P. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004.
[31] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trend in Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.
[32] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Englewood Cliffs, NJ, USA: Prentice Hall, 1989.
[33] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," 2014. [Online]. Available: http://cvxr.com/cvx
[34] NREL: Measurement and Instrumentation Data Center. (2016). [Online]. Available: http://www.nrel.gov/midc/
[35] D. Narayanan, A. Donnelly, and A. Rowstron, "Write off-loading: Practical power management for enterprise storage," in *Proc. USENIX Conf. File Storage Technol.*, Feb. 2008, pp. 253–267.
[36] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters," in *Proc. 6th Conf. Symp. Oper. Syst. Des. Implementation*, 2004, pp. 137–149.
[37] Y. Chen, A. Ganapathi, R. Griffith, and R. Katz, "The case for evaluating mapReduce performance using workload suites," in *Proc. IEEE 19th Annu. Int. Symp. Modelling Anal. Simul. Comput. Telecommun. Syst.*, 2011, pp. 390–399.

Yuanxiong Guo (M'14) received the BEng degree in electronics and information engineering from Huazhong University of Science and Technology, Wuhan, China, in 2009, and the MS and PhD degrees in electrical and computer engineering from the University of Florida, Gainesville, FL, USA, in 2012 and 2014, respectively. Since 2014, he has been an assistant professor with the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK, USA. His current research interests include resource management and cybersecurity for cyber-physical energy systems, wireless networks, and cloud data centers. He is a recipient of the best paper award in the IEEE Global Communications Conference 2011. He is a member of the IEEE and ACM.

Miao Pan (M'12) received the BSc degree in electrical engineering from Dalian University of Technology, China, in 2004, MASc degree in electrical and computer engineering from Beijing University of Posts and Telecommunications, China, in 2007 and the PhD degree in electrical and computer engineering from the University of Florida, in 2012, respectively. He is now an assistant professor in the Department of Electrical and Computer Engineering, University of Houston. He was an assistant professor in the Computer Science, Texas Southern University from 2012 to 2015. His research interests include cognitive radio networks, cybersecurity, and cyber-physical systems. His work on cognitive radio network won best paper award in Globecom 2015. He is currently associate editor for *IEEE Internet of Things (IoT)* Journal. He is a member of the IEEE.



Yanmin Gong (M'16) received the BEng degree in electronics and information engineering from Huazhong University of Science and Technology, China, in 2009, the MS degree in electrical engineering from Tsinghua University, China, in 2012, and the PhD degree in electrical and computer engineering from the University of Florida, Gainesville, FL, USA, in 2016. She has been an assistant professor in the School of Electrical and Computer Engineering, Oklahoma State University since August 2016. Her research interests include information security and privacy and mobile and wireless security and privacy, such as security in Internet-of-Things and privacy-preserving big data analytics. She has served as the technical program committee members for several conferences including the IEEE INFOCOM and CNS. She is a member of the IEEE and ACM.



Yuguang Fang (F'08) received the MS degree from Qufu Normal University, Shandong, China in 1987, the PhD degree from Case Western Reserve University, in 1994, and the PhD degree from Boston University in 1997. He joined the Department of Electrical and Computer Engineering, University of Florida, in 2000 and has been a full professor since 2005. He held a University of Florida Research Foundation (UFRF) Professorship (2017-2020, 2006-2009), University of Florida Term Professorship (2017-2019), a Changjiang scholar chair professorship (Xidian University, Xi'an, China, 2008-2011; Dalian Maritime University, Dalian, China, 2015-present), overseas academic master (Dalian University of Technology, Dalian, China, 2016-2018), and a guest chair professorship with Tsinghua University, China (2009-2012). He received the US national science foundation career award in 2001, the office of naval research young investigator award, in 2002, the 2015 the IEEE Communications Society CISTC technical recognition award, the 2014 the IEEE communications society WTC recognition award, and the Best Paper Award from IEEE ICNP (2006). He has also received a 2010-2011 UF doctoral dissertation advisor/mentoring award, a 2011 Florida Blue Key/UF Homecoming Distinguished Faculty Award, and the 2009 UF College of Engineering Faculty Mentoring Award. He is the editor-in-chief of the IEEE Transactions on Vehicular Technology (2013-present), was the editor-in-chief of the IEEE Wireless Communications (2009-2012), and serves/served on several editorial boards of journals including *IEEE Transactions on Mobile Computing* (2003-2008, 2011-present), *IEEE Transactions on Communications* (2000-2011), and *IEEE Transactions on Wireless Communications* (2002-2009). He has been actively participating in conference organizations such as serving as the technical program co-chair for the IEEE INOFOCOM'2014 and the technical program vice-chair for the IEEE INFOCOM'2005. He is a fellow of the american association for the advancement of science (AAAS). He is a fellow of the IEEE.