Distributed Real-Time Energy Management in Data Center Microgrids

Liang Yu, *Member, IEEE*, Tao Jiang, *Senior Member, IEEE*, and Yulong Zou, *Senior Member, IEEE*

Abstract—Data center operators are typically faced with three significant problems when running their data centers, i.e., rising electricity bills, growing carbon footprints and unexpected power outages. To mitigate these issues, running data centers in microgrids is a good choice since microgrids can enhance the energy efficiency, sustainability and reliability of electrical services. Thus, in this paper, we investigate the problem of energy management for multiple data center microgrids. Specifically, we intend to minimize the long-term operational cost of data center microgrids by taking into account the uncertainties in electricity prices, renewable outputs and data center workloads. We first formulate a stochastic programming problem with the considerations of many factors, e.g., providing heterogeneous service delay guarantees for batch workloads, interactive workload allocation, batch workload shedding, electricity buying/selling, battery charging/discharging efficiency, and the ramping constraints of backup generators. Then, we design a realtime and distributed algorithm for the formulated problem based on Lyapunov optimization technique and a variant of alternating direction method of multipliers (ADMM). Moreover, the performance guarantees provided by the proposed algorithm are analyzed. Extensive simulation results indicate the effectiveness of the proposed algorithm in operational cost reduction for data center microgrids.

Index Terms—Data centers, energy management, microgrids, realtime and distributed algorithm

I. INTRODUCTION

WITH the development of Internet services and applications, massive geo-distributed data centers have been deployed. When running these data centers, a data-center operator is typically faced with three significant problems: (1) rising electricity bills, e.g., Google consumed 2260 GWh in 2010 and the corresponding electricity bill was larger than 1.35 billion dollars [1]; (2) growing carbon emission, e.g., data center carbon emissions are expected to reach 2.6% of the total emissions [1]; (3) unexpected power outages, e.g., Amazon experienced several power outages during 2010-2013 and knocked many customers offline [2]. Since microgrids could potentially provide cost savings, emission reduction and reliability enhancement for data centers [3]–[7], it is necessary to study the problem of energy management for data center microgrids.

There has been few work on the energy management in microgrids. In [10], Guan *et al.* investigated the scheduling

L. Yu and Y. Zou are with Key Laboratory of Broadband Wireless Communication and Sensor Network Technology of Ministry of Education, Nanjing University of Posts and Telecommunications, Nanjing 210003, P. R. China. (email: liang.yu@njupt.edu.cn)

T. Jiang is Wuhan National Laboratory for Optoelectronics, School of Electronics Information and Communications, Huazhong University of Science and Technology, Wuhan 430074, P. R. China.

problem of building energy supplies in a microgrid. In [11], Erol-Kantarci *et al.* developed the idea of resource sharing among microgrids for the sake of increased reliability. In [12], Huang *et al.* presented a novel energy management framework to minimize the operational cost of a microgrid by introducing a model of QoSE (quality-of-service in electricity). In [13], Zhang *et al.* considered an optimal energy management problem for both supply and demand of a grid-connected microgrid incorporating renewable energy sources. In [14], Zhang *et al.* proposed an online algorithm to minimize the total energy cost of the conventional energy drawn from the main grid over a finite horizon by scheduling energy storage devices in a microgrid. In [15], Wang *et al.* designed a distributed algorithm for online energy management in networked microgrids with a high penetration of distributed energy resources using online ADMM with Regret. In [16], Guo *et al.* proposed a two-stage adaptive robust optimization approach for the energy management of a microgrid. In [3], Salomonsson *et al.* designed an adaptive control system for a dc microgrid with data center loads. In [17], Shi *et al.* proposed an online energy management strategy for realtime operation of a microgrid with the considerations of the power flow and system operational constraints on a distribution network. In [6], Li *et al.* studied the problem of minimizing the operation cost of a data center microgrid. In [18], Chen *et al.* proposed a cooling-aware realtime algorithm to minimize the long-term operational cost of a data center microgrid. In [7], Thompson *et al.* presented a methodology for optimizing investment in data center battery storage capacity in a microgrid. Though some positive results have been obtained in the above works, there is no work that focuses on the realtime and distributed energy management for multiple data center microgrids. In our previous works [4] [5] [19], we mainly focus on the realtime energy management for multiple data center microgrids from different perspectives, e.g., energy cost reduction and carbon emission reduction. However, such previous works neglect heterogeneous service delay guarantees for batch workloads in all data centers [20] and distributed implementation for the proposed realtime algorithm.

Based on the above observation, this paper investigates the problem of realtime distributed energy management for multiple data center microgrids considering the drawbacks in our previous works. The resulting challenge consists of two aspects, i.e., spatial and temporal couplings [21]. On one hand, there are some spatial couplings among all microgrids due to the allocation of interactive workloads. On the other hand, to provide the heterogeneous service delay guarantees for batch workloads in all data centers and keep all energy storage

systems stable, several temporal couplings are incurred.

To deal with the above challenge, we first formulate a stochastic programming problem to minimize the time average expected operational cost by jointly capturing the constraints with geographical load balancing, batch workload allocation/shedding, heterogeneous service delay guarantees for batch workloads, electricity buying/selling, battery charging/discharging management, backup generators, and power balancing. Since the formulated optimization problem is a large-scale nonlinear stochastic programming with "time-coupling" constraints, we propose a realtime and distributed algorithm based on Lyapunov optimization technique [8] and a variant of alternating direction method of multipliers (ADMM) [31][1]. The key idea of the proposed algorithm is given as follows. Firstly, we propose a realtime algorithm for the formulated problem based on Lyapunov optimization technique so that "time-coupling" constraints could be avoided. Then, we present the distributed implementation of the proposed realtime algorithm without considering the nonlinear constraints based on a variant of ADMM. Next, a feasible solution to the original problem could be obtained by adjustment so that the nonlinear constraints in the formulated problem could be satisfied. Furthermore, the performance analysis of the proposed algorithm is carried out.

The main contributions of this paper are summarized below:

- We formulate a stochastic programming to minimize the long-term operational cost of multiple data center microgrids with the considerations of many factors, e.g., providing heterogeneous service delay guarantees for batch workloads, interactive workload allocation, batch workload shedding, electricity buying/selling, battery charging/discharging efficiency, and the ramping constraints of backup generators.
- We propose a realtime and distributed algorithm to solve the formulated problem based on Lyapunov optimization technique and a variant of ADMM. Moreover, we analyze the performance guarantees provided by the proposed algorithm. Note that the proposed algorithm does not require any prior knowledge of statistical characteristics associated with system parameters and has low computational complexity.
- We conduct extensive simulations to evaluate the performance of the proposed algorithm. Simulation results show that the proposed algorithm outperforms other benchmark schemes in operational cost reduction.

The rest of this paper is organized as follows. In Section II, we describe the system model and problem formulation.

Section III proposes a realtime and distributed algorithm to solve the formulated problem. Section IV gives the algorithmic performance analysis. Extensive simulations are conducted in Section V. Finally, conclusions are drawn in Section VI.

II. MODEL AND FORMULATION

We consider a data center operator that has some geo-distributed data centers located in different electric regions as shown in Fig. 1, where each data center operates in a smart microgrid (SMG) environment [11]. As far as the operation condition of a SMG is concerned, there are two modes, i.e., the islanded mode and the grid-connected mode. In the islanded mode, SMGs could supply their loads using multiple energy resources, e.g., energy storage devices, renewable and backup generators. In contrast, a SMG could sell (buy) energy to (from) a main grid in the grid-connected mode. A SMG considered in this paper consists of four main components, i.e., a generation system, a load, an energy storage system (ESS), and an energy management system (EMS). Specifically, a generation system consists of several renewable generators and a conventional generator (usually adopted as the backup generator), while the EMS is responsible for the energy scheduling of other components in the SMG. As the aggregated load in the SMG, a data center needs to finish the interactive workloads dispatched from front-end servers and the batch workloads within the data center. In this paper, we consider a time-slotted system and the length of each slot is assumed to be unit time. For easy reading, the main notations are introduced in Table I.
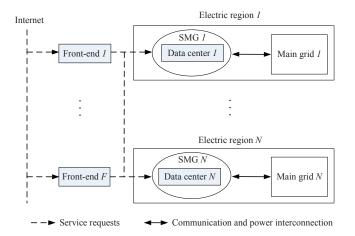

Fig. 1. System model

A. Models Associated with Data Centers and Front-end Servers

Suppose that there are N data centers geographically distributed in N SMGs, which connected to N main grids. Therefore, a common index i ($1 \leq i \leq N$) is adopted for data centers, SMGs and main grids. Moreover, we assume that data center i consists of C_i homogeneous servers[2]. In time slot t,

---

[1]In [22], Sun *et al.* adopted Lyapunov optimization technique and ADMM with two blocks to deal with the problem of power balancing in a renewable-integrated power grid with storage and flexible loads. Though data centers could be also regarded as a kind of flexible loads, it does not mean that the method in [22] could be applied to our problem directly. The reason is that this paper considers both multiple data center microgrids and heterogeneous service delay guarantees for batch workloads in all data centers. Specifically, to provide the heterogeneous service delay guarantees for batch workloads in all data centers and keep all energy storage systems stable, we adopt three queues and define a weighted quadratic Lyapunov function when proposing a realtime algorithm with Lyapunov optimization technique. Moreover, we design a distributed implementation of the proposed realtime algorithm based on ADMM with eleven blocks.

[2]Although all the servers at a data center are assumed to be homogeneous, the model could be extended to the case with heterogeneous servers by adopting a few additional notations.

TABLE I
NOTATIONS

Notation	Definition
t	Time slot index ($1 \leq t \leq T$)
f	front-end server index ($1 \leq f \leq F$)
i	A common index for data centers, SMGs and main grids
f	Front-end server f
$\lambda_{f,t}$	The number of interactive workloads at front-end server f at t
$d_{f,i,t}$	Interactive workload allocation from front-end server f to DC i at t
$\pi_{i,q,t}$	The quantity of batch workloads with type q at t ($1 \leq q \leq M_i$)
$Q_{i,q,t}$	Batch workload queue
$x_{i,q,t}$	The served workloads in batch workload queue at t
$e_{i,q,t}$	The quantity of dropped batch workloads at t
$R_{i,q}^{\max}$	The maximum queueing delay associated with $\pi_{i,q,t}$
$\mathcal{T}_{i,q}$	The tolerant service delay associated with $\pi_{i,q,t}$
$P_{i,\mathrm{idle}}$	Idle power of a server in data center i
$P_{i,\mathrm{peak}}$	Peak power of a server in data center i
$p_{i,t}$	Total power consumption in data center i at t
$r_{i,t}$	The total power output of the renewable generators in SMG i at t
$c_{i,t}$	The power output of the conventional generator in SMG i at t
ϵ_i	Ramping coefficient of the conventional generator in SMG i
$u_{c,i,t}$	The charging power for the ESS in SMG i at t
$u_{d,i,t}$	The discharging power for the ESS in SMG i at t
$D_{i,t}$	The stored energy level of the ESS i at t
$S_i(t)$	Purchasing electricity price from main grid i at t
$W_i(t)$	Selling electricity price to main grid i at t
$g_{i,t}$	Energy transactions between SMG i and main grid i at t
$\Gamma_{1,t}$	The cost incurred by electricity buying and selling at t
$\Gamma_{2,t}$	Total revenue loss of serving interactive requests at t
$\Gamma_{3,t}$	The penalty cost imposed on dropping batch workloads at t
$\Gamma_{4,t}$	Battery depreciation cost at t
$\Gamma_{5,t}$	Generation cost of the conventional generators at t
$H_{i,q,t}$	Delay-aware virtual queue
$Z_{i,t}$	Virtual energy queue
Δ_t	one-slot conditional Lyapunov drift
ΔV_t	drift-plus-penalty term

the total quantity of interactive workloads (in the number of servers required) at the front-end server f ($1 \leq f \leq F$) is $\lambda_{f,t}$. Let $d_{f,i,t}$ be the quantity of interactive workloads allocated from front-end server f to data center i at slot t. Then, we have [23] [24]

$$\sum_{i=1}^{N} d_{f,i,t} = \lambda_{f,t}, \ \forall f, t, \tag{1}$$

$$d_{f,i,t} \geq 0, \ \forall f, i, t. \tag{2}$$

Besides interactive workloads, some resource elastic batch workloads are commonly processed within data centers, e.g., scientific applications, data mining jobs. Batch workloads could be scheduled at any time slot as long as they are processed before their deadlines. Thus, batch workloads could be buffered and served in proper time slot. Let $\pi_{i,q,t}$ be the quantity of batch workloads at slot t (also in terms of the number of servers required) with type q ($1 \leq q \leq M_i$) in data center i. By storing batch workloads $\pi_{i,q,t}$ in a queue $Q_{i,q,t}$ according to its type q, we have

$$Q_{i,q,t+1} = \max[Q_{i,q,t} - x_{i,q,t}, 0] + \pi_{i,q,t}, \ \forall i, q, t \tag{3}$$

where $x_{i,q,t}$ denotes the served workloads in the queue q of data center i at slot t. Denote the maximum value of $x_{i,q,t}$ by

$x_{i,q}^{\max}$, where $x_{i,q}^{\max} \geq \pi_{i,q}^{\max}$ ($\pi_{i,q}^{\max} = \max_t \pi_{i,q,t}$) so that it is always possible to make the queue $Q_{i,q,t}$ stable (and this can be done with one slot delay if we choose $x_{i,q,t} = x_{i,q}^{\max}$ for all t). In addition, by observing the structure of $Q_{i,q,t}$, it can be found that there is no need to serve the batch workload that is larger than $Q_{i,q,t}$. Thus, we have

$$0 \leq x_{i,q,t} \leq \min\{x_{i,q}^{\max}, Q_{i,q,t}\}, \ \forall i, q, t. \tag{4}$$

To keep workload queues $Q_{i,q,t}$ stable, the batch workloads should be served without waiting for a long time. Since the summation of served batch workloads $\sum_{q=1}^{M_i} x_{i,q,t}$ and arrived interactive workloads $\sum_{f=1}^{F} d_{f,i,t}$ may exceed the processing capacity of data center i, some batch workloads have to be dropped at this time. Let $e_{i,q,t}$ be the quantity of dropped batch workloads, we have

$$\sum_{f=1}^{F} d_{f,i,t} + \sum_{q=1}^{M_i} (x_{i,q,t} - e_{i,q,t}) \leq C_i, \ \forall i, t. \tag{5}$$

$$0 \leq e_{i,q,t} \leq x_{i,q,t}, \ \forall i, q, t. \tag{6}$$

For any control algorithm, it is necessary to ensure that the average length of the workload queue q in data center i is finite so that batch workloads could be finished without waiting an arbitrarily long time, i.e.,

$$\overline{Q}_{i,q} = \limsup_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\{Q_{i,q,t}\} < \infty. \tag{7}$$

Note that (7) is not enough to ensure the heterogeneous service delay for batch workload $\pi_{i,q,t}$, we adopt the following constraint,

$$R_{i,q}^{\max} \leq \mathcal{T}_{i,q}, \ \forall i, q, t \tag{8}$$

where $R_{i,q}^{\max}$ and $\mathcal{T}_{i,q}$ are the maximum queueing delay and the tolerant service delay associated with the batch workload added into the queue $Q_{i,q,t}$ of data center i at slot t, respectively. In Section V, we will provide the specific expression of $R_{i,q}^{\max}$.

Let PUE_i be the PUE[3] of data center i, $P_{i,\mathrm{idle}}$ and $P_{i,\mathrm{peak}}$ represent the idle power and peak power of a server in data center i, respectively. Then, the total power consumption in data center i at slot t $p_{i,t}$ could be estimated by [25]

$$p_{i,t} = \alpha_i + \beta_i \Big(\sum_{f=1}^{F} d_{f,i,t} + \sum_{q=1}^{M_i} (x_{i,q,t} - e_{i,q,t}) \Big), \ \forall i, t, \tag{9}$$

where $\alpha_i \triangleq C_i(P_{i,\mathrm{idle}} + (\mathrm{PUE}_i - 1)P_{i,\mathrm{peak}})$, $\beta_i \triangleq P_{i,\mathrm{peak}} - P_{i,\mathrm{idle}}$.

B. Models Related to the Generation System and ESS

1) Generation Model: Let $r_{i,t}$ and $c_{i,t}$ be the total power output of the renewable generators and the power output of the conventional generator in SMG i at slot t, respectively. Then, we have

$$0 \leq c_{i,t} \leq c_{i,\max}, \ \forall i, t, \tag{10}$$

[3]PUE is defined as the ratio of the total power consumption at a data center to the power consumption at IT equipments

where $c_{i,\max}$ is the maximum power output associated with the conventional generator in SMG i. Considering the physical constraints of the conventional generator, the output change in two consecutive slots is limited instead of arbitrarily large, which is reflected by a so-called ramping constraint. Without loss of generality, the ramp-up and ramp-down constraints are regarded as the same [22]. Then, we have

$$|c_{i,t} - c_{i,t-1}| \leq \epsilon_i c_{i,\max}, \ \forall i, t, \tag{11}$$

where ϵ_i is the ramping coefficient associated with the conventional generator in SMG i.

2) ESS Model: We define $u_{c,i,t}$ and $u_{d,i,t}$ to represent the charging and discharging power for the ESS in SMG i at slot t. Then, we have

$$0 \leq u_{c,i,t} \leq u_{i,\mathrm{cmax}}, \ \forall i, t, \tag{12}$$

$$0 \leq u_{d,i,t} \leq u_{i,\mathrm{dmax}}, \ \forall i, t, \tag{13}$$

where $u_{i,\mathrm{cmax}}$ and $u_{i,\mathrm{dmax}}$ are maximum charging power and discharging power, respectively. Denote $\eta_{c,i}$ and $\eta_{d,i}$ be the charging and discharging efficiency of the ESS in SMG i at slot t, respectively. In addition, simultaneous charging and discharging are not allowed considering the round-trip inefficiency, i.e.,

$$u_{c,i,t} \cdot u_{d,i,t} = 0, \forall i, t. \tag{14}$$

Let $D_{i,t}$ be the stored energy of the ESS i, we have

$$D_{i,\min} \leq D_{i,t} \leq D_{i,\max}, \ \forall i, t, \tag{15}$$

where $D_{i,\max}$ and $D_{i,\min}$ denote the maximum and the minimum capacity of the ESS i, respectively. In addition, the storage dynamics of the ESS i could be modeled by

$$D_{i,t+1} = D_{i,t} + \eta_{c,i} u_{c,i,t} - \frac{1}{\eta_{d,i}} u_{d,i,t}, \ \forall i, t. \tag{16}$$

To satisfy the energy demand of data centers, SMGs may exchange energy with main grids. Denote the electricity price of buying and selling energy by $X_{i,t} \in [X_{i,\min}, \ X_{i,\max}]$ and $W_{i,t} \in [W_{i,\min}, \ W_{i,\max}]$, respectively. As in [13], the selling price is assumed to be strictly smaller than the purchasing price so that energy arbitrage could be avoided, i.e., $X_{i,t} > W_{i,t}$. To achieve the real-time power balancing, we have the following constraints, i.e.,

$$g_{i,t} + r_{i,t} + c_{i,t} + u_{d,i,t} = p_{i,t} + u_{c,i,t}, \ \forall i, t, \tag{17}$$

where $g_{i,t}$ denotes the energy transactions between SMG i and main grid i at slot t, which is bounded by

$$G_{i,\mathrm{smax}} \leq g_{i,t} \leq G_{i,\mathrm{bmax}}, \ \forall i, t, \tag{18}$$

where $G_{i,\mathrm{bmax}} > 0$ and $G_{i,\mathrm{smax}} < 0$ are determined by the physical limitations, e.g., transmission lines [12]. As in [4], $G_{i,\mathrm{bmax}}$ and $G_{i,\mathrm{smax}}$ are assumed to be large enough to support the normal operation of SMG i in the grid-connected mode.

C. Operational Cost Model

Denote the total operational cost of the data center operator at slot t by Γ_t, which includes several components, i.e., the cost of purchasing and selling electricity $\Gamma_{1,t}$, revenue loss associated with workload allocation $\Gamma_{2,t}$ and $\Gamma_{3,t}$, the battery depreciation cost $\Gamma_{4,t}$, and the total generation cost of conventional generators $\Gamma_{5,t}$. Specifically, the cost incurred by electricity buying and selling at slot t $\Gamma_{1,t}$ is obtained below,

$$\Gamma_{1,t} = \sum_{i=1}^{N} \left(\frac{X_{i,t} - W_{i,t}}{2} |g_{i,t}| + \frac{X_{i,t} + W_{i,t}}{2} g_{i,t} \right). \tag{19}$$

For interactive applications, latency is the most important performance metric and a moderate increase in user-perceived latency would translate into substantial revenue loss for the data center operator [26] [27]. To model the utility of the interactive workload, the convex function in [26] is adopted, which converts the mean propagation delay into revenue loss, i.e., $\omega \sum_{i=1}^{N} d_{f,i,t} L_{f,i}/\lambda_{f,t}$, where ω is a conversion factor; $L_{f,i}$ is the propagation latency between the front-end server f and data center i. Then, the total revenue loss of serving interactive requests is described by $\Gamma_{2,t} = \omega \sum_{f=1}^{F} \sum_{i=1}^{N} d_{f,i,t} L_{f,i}$.

In addition, to model the revenue loss of allocating processing servers for batch workload, the following function is adopted as in [27], $\Gamma_{3,t} = \sum_{i=1}^{N} \sum_{q=1}^{M_i} \theta_i e_{i,q,t}$, where θ_i is the penalty factor imposed on dropping batch workloads.

It is known that charging and discharging of batteries would affect their lifetime. To model such depreciation cost, the penalty function $B_i(u_{c,i,t}, u_{d,i,t})$ is adopted. Continually, we have $\Gamma_{4,t} = \sum_{i=1}^{N} B_i(u_{c,i,t}, u_{d,i,t})$.

Denote the generation cost function of the conventional generator at slot t by $A_i(c_{i,t})$. Then, $\Gamma_{5,t} = \sum_{i=1}^{N} A_i(c_{i,t})$.

With the above-mentioned cost components, the total operational cost of the data center operator is calculated by $\Gamma_t = \sum_{l=1}^{5} \Gamma_{l,t}$.

D. Operational Cost Minimization Problem

With the aforementioned models, we can formulate a stochastic programming problem to minimize the time average expected operational cost of data center microgrids as follows,

$$(\textbf{P1}) \quad \min \ \limsup_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\{\Gamma_t\}, \tag{20a}$$

$$s.t. \ (1) - (18), \tag{20b}$$

where $\mathbb{E}\{\cdot\}$ is the expectation operator; the decision variables are $d_{f,i,t}, x_{i,q,t}, e_{i,q,t}, c_{i,t}, g_{i,t}, u_{c,i,t}$ and $u_{d,i,t}$; the expectation in the objective function is taken over the randomness of the system parameters $\lambda_{f,t}, \pi_{i,q,t}, r_{i,t}, X_{i,t}$ and $W_{i,t}$, and the possibly random control actions at each time slot.

For simplicity, the cost functions $A_i(\cdot)$ and $B_i(\cdot)$ are assumed to be continuously differentiable and convex, which is reasonable since many practical costs could be well approximated by such functions [14] [28]. Let $A_i'(\cdot)$ and $B_i'(\cdot)$ be the derivatives of $A_i(\cdot)$ and $B_i(\cdot)$, respectively. In addition, we suppose that $A_i'(c_{i,t})$ and $B_i'(u_{c,i,t}, u_{d,i,t})$ are bounded within the intervals $[A_{i,\min}', \ A_{i,\max}']$ and $[B_{i,\min}', \ B_{i,\max}']$, respectively.
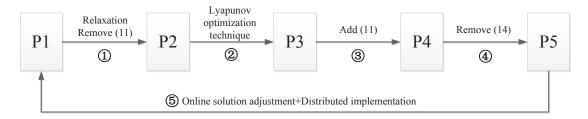
Fig. 2. An illustration of the key idea of the proposed algorithm

III. ALGORITHM DESIGN

There are three challenges to solve **P1**. Firstly, **P1** is a large-scale nonlinear optimization problem as the data center operator may deploy tens of geo-distributed data centers and hundreds of thousands of front-end servers around the world. Secondly, the future parameters are not known, including workload, renewable generation output and electricity price. Thirdly, the constraints (11) and (16) bring the "time coupling" property to **P1**, which means that the current decision can impact the future decision. Previous methods to handle the "time coupling" problem are usually based on dynamic programming, which suffers from the curse of dimensionality problem. The structure and size of **P1** motivates us to design a scalable distributed realtime algorithm that is applicable for practical applications.

The key idea of the proposed algorithm can be illustrated by Fig. 2. Specifically, we can first transform the original problem **P1** into a stochastic programming problem **P2** with time average constraints by removing the constraint (11). Then, we can transform **P2** into one-slot minimization problem **P3** using Lyapunov optimization technique. Next, by incorporating the constraint (11) into **P3**, we obtain **P4**. Since there are nonlinear constraints (14) in **P4**, we transform **P4** into **P5** by removing (14). After obtaining the solution of **P5**, we adjust the solution so that (14) could be satisfied. Finally, we provide the distributed implementation of the proposed online algorithm and prove that all constraints of **P1** could be satisfied by the proposed algorithm.

Since Lyapunov optimization technique (LOT) could be used to solve a stochastic programming problem with time average constraints, we need to transform (15) and (16) into the time average constraints. To be specific, we define $\overline{u_{c,i}}$ and $\overline{u_{d,i}}$ as follows,

$$\overline{u_{c,i}} = \limsup_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\{u_{c,i,t}\}, \qquad (21)$$

$$\overline{u_{d,i}} = \limsup_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\{u_{d,i,t}\}. \qquad (22)$$

It is not difficult to obtain that $\eta_{c,i}\overline{u_{c,i}} = \frac{1}{\eta_{d,i}}\overline{u_{d,i}}$. Continually, **P1** could be relaxed into **P2** below,

$$(\textbf{P2}) \quad \min \ \limsup_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\{\Gamma_t\}, \qquad (23a)$$

$$s.t. \ (1),(2),(4)-(10),(12)-(14),(17),(18), \qquad (23b)$$

$$\eta_{c,i}\overline{u_{c,i}} = \frac{1}{\eta_{d,i}}\overline{u_{d,i}}, \ \forall i. \qquad (23c)$$

To solve **P2**, LOT intends to transform time average constraints into queue stability problems. Thus, a virtual energy queue $Z_{i,t}$ is adopted to ensure the feasibility of $\eta_{c,i}\overline{u_{c,i}} = \frac{1}{\eta_{d,i}}\overline{u_{d,i}}$, i.e.,

$$Z_{i,t} = D_{i,t} - D_{i,\min} - V\eta_{d,i}\gamma_{i,\max} - \frac{1}{\eta_{d,i}}u_{i,\text{dmax}}, \qquad (24)$$

where $\gamma_{i,\max} = \max\{X_{i,\max}, W_{i,\max}, A'_{i,\max}\}$; $V \in [0, V_{\max}]$ is a control parameter that would be specified later. Continually, the update equation of $Z_{i,t}$ is obtained as follows,

$$Z_{i,t+1} = Z_{i,t} + \eta_{c,i}u_{c,i,t} - \frac{1}{\eta_{d,i}}u_{d,i,t}, \ \forall i, t. \qquad (25)$$

Similarly, to ensure the feasibility of (7), we need to keep the workload queue $Q_{i,q,t}$ stable. In addition, to ensure the feasibility of (8), we adopt a delay-aware virtual queue $H_{i,q,t}$. Specifically, for each i and q, $H_{i,q,t}$ with $H_{i,q,0} = 0$ and with dynamics as follows,

$$H_{i,q,t+1} = \begin{cases} [H_{i,q,t} - x_{i,q,t} + \varepsilon_{i,q}]^+, & Q_{i,q,t} > x_{i,q,t}, \\ 0, & Q_{i,q,t} \le x_{i,q,t}, \end{cases} \qquad (26)$$

where $[\diamond]^+ \triangleq \max\{\diamond, 0\}$; $\varepsilon_{i,q}$ is a fixed parameter, which would be specified later. It can be observed that $H_{i,q,t+1}$ has the same service rate as $Q_{i,q,t+1}$ but has a new arrival rate $\varepsilon_{i,q}$ when $Q_{i,q,t} > x_{i,q,t}$, which can ensure that $H_{i,q,t+1}$ grows when the batch workload added into the queue $Q_{i,q,t}$ at slot t is still waiting to be satisfied. If we can ensure that the queues $H_{i,q,t}$ and $Q_{i,q,t}$ have finite upper bounds, then the maximum queueing delay in queue $Q_{i,q,t}$ defined in the following Lemma could be guaranteed.

Lemma 1 (Maximum Queueing Delay) Suppose we can control the system so that $H_{i,q,t} \le H_{i,q}^{\max}$ and $Q_{i,q,t} \le Q_{i,q}^{\max}$ for all i, q and t. Then, all energy demands in the queue $Q_{i,q,t}$ would be served with a maximum queueing delay $R_{i,q}^{\max}$ slots, where

$$R_{i,q}^{\max} \triangleq \lceil (H_{i,q}^{\max} + Q_{i,q}^{\max})/\varepsilon_{i,q} \rceil. \qquad (27)$$

Proof: See Appendix A. In addition, in Section V, it can be proved that the constants $H_{i,q}^{\max}$ and $Q_{i,q}^{\max}$ indeed exist. ∎

According to the framework of LOT, solving **P2** is equivalent to solving **P2'** as follows,

$$(\textbf{P2'}) \quad \min \ \limsup_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\{\Gamma_t\}, \qquad (28a)$$

$$s.t. \ (1),(2),(4)-(6),(9),(10),(12)-(14),(17),(18),$$

Queues $Q_{i,q,t}$, $H_{i,q,t}$, and $Z_{i,q,t}$ are mean rate stable.

A. The Proposed Realtime Algorithm

Define $\Theta_t \triangleq (Q_t, H_t, Z_t)$ as the concatenated vector of the real workload queue, virtual workload queue and virtual energy queue, where

$$Q_t = (Q_{1,1,t}, \cdots, Q_{1,M_1,t}, \cdots, Q_{N,1,t}, \cdots, Q_{N,M_N,t}),$$
$$H_t = (H_{1,1,t}, \cdots, H_{1,M_1,t}, \cdots, Z_{N,1,t}, \cdots, Z_{N,M_N,t}),$$
$$Z_t = (Z_{1,t}, Z_{2,t}, \cdots, Z_{N,t}).$$

To keep the stability of all queues, we first define a weighted quadratic Lyapunov function as follows,

$$\mathcal{L}_t \triangleq \frac{1}{2} \sum_{i=1}^{N} \Big(\sum_{q=1}^{M_i} w(Q_{i,q,t}^2 + H_{i,q,t}^2) + Z_{i,t}^2 \Big), \quad (29)$$

where w is a positive weight for workload queues, which indicates the relative importance of the workload queues with respect to the energy queues.

Then, a one-slot conditional Lyapunov drift could be obtained below,

$$\Delta_t = \mathbb{E}\{\mathcal{L}_{t+1} - \mathcal{L}_t | \Theta_t\}, \quad (30)$$

where the expectation is taken with respect to the randomness of workloads, renewable generation outputs, electricity prices, and the randomness in control policies.

Next, by adding a function of the expected operational cost in a slot to (30), we can obtain a *drift-plus-penalty* term as follows,

$$\Delta V_t = \Delta_t + V\mathbb{E}\{\Gamma_t | \Theta_t\}. \quad (31)$$

Lemma 2 (Drift Bound) The *drift-plus-penalty* term satisfies the following inequality for all slots,

$$\Delta V_t \leq \Omega_0 + V\mathbb{E}\{\Gamma_t | \Theta_t\}$$
$$+ \mathbb{E}\{\sum_{i=1}^{N} \sum_{q=1}^{M_i} wQ_{i,q,t}(\pi_{i,q,t} - x_{i,q,t}) | \Theta_t\}$$
$$+ \mathbb{E}\{\sum_{i=1}^{N} \sum_{q=1}^{M_i} wH_{i,q,t}(\varepsilon_{i,q} - x_{i,q,t}) | \Theta_t\}$$
$$+ \mathbb{E}\{\sum_{i=1}^{N} Z_{i,t}(\eta_{c,i} u_{c,i,t} - \frac{1}{\eta_{d,i}} u_{d,i,t}) | \Theta_t\}. \quad (32)$$

where Ω_0 is given by

$$\Omega_0 = \sum_{i=1}^{N} \sum_{q=1}^{M_i} \left(w\frac{(\pi_{i,q}^{\max})^2 + (x_{i,q}^{\max})^2 + \max\{\varepsilon_{i,q}^2, (x_{i,q}^{\max})^2\}}{2} \right)$$
$$+ \sum_{i=1}^{N} \frac{\max\{(\eta_{c,i} u_{i,\text{cmax}})^2, (\frac{1}{\eta_{d,i}} u_{i,\text{dmax}})^2\}}{2}. \quad (33)$$

Proof: See Appendix B. ∎

Minimizing the R.H.S. of the upper bound of drift-plus-penalty term in each slot t, we have the following optimization

problem **P3** as follows,

$$(\textbf{P3}) \ \min \ V\Gamma_t - \sum_{i=1}^{N} \sum_{q=1}^{M_i} w(Q_{i,q,t} + H_{i,q,t})x_{i,q,t} +$$
$$\sum_{i=1}^{N} Z_{i,t}(\eta_{c,i} u_{c,i,t} - \frac{1}{\eta_{d,i}} u_{d,i,t}) \quad (34a)$$
$$s.t. \ (1), (2), (4) - (6), (9), (10), (12) - (14), (17), (18).$$

Since **P3** neglects the constraint (11), we can obtain **P4** by adding (11) into the constraints of **P3**, i.e.,

$$(\textbf{P4}) \ \min \ V\Gamma_t - \sum_{i=1}^{N} \sum_{q=1}^{M_i} w(Q_{i,q,t} + H_{i,q,t})x_{i,q,t} +$$
$$\sum_{i=1}^{N} Z_{i,t}(\eta_{c,i} u_{c,i,t} - \frac{1}{\eta_{d,i}} u_{d,i,t}) \quad (35a)$$
$$s.t. \ (1), (2), (4) - (6), (9) - (14), (17) - (18).$$

Since the constraint (14) is nonlinear, **P4** is a nonlinear programming problem. To simplify the computation, we can first ignore the nonlinear constraint (14), and then adjust the obtained solution to satisfy (14). Based on the above description, an algorithm for **P1** could be described by Algorithm 1, where **P5** is defined as follows,

$$(\textbf{P5}) \ \min \ V\Gamma_t - \sum_{i=1}^{N} \sum_{q=1}^{M_i} w(Q_{i,q,t} + H_{i,q,t})x_{i,q,t} +$$
$$\sum_{i=1}^{N} Z_{i,t}(\eta_{c,i} u_{c,i,t} - \frac{1}{\eta_{d,i}} u_{d,i,t}) \quad (36a)$$
$$s.t. \ (1), (2), (4) - (6), (9) - (13), (17) - (18). \quad (36b)$$

Algorithm 1 : Realtime Algorithm for Operational Cost Minimization Problem

---

1: **For** each slot t **do**
2: Observing system states at the starting point of time slot t: $Q_{i,q,t}$, $H_{i,q,t}$, $Z_{i,t}$, $\lambda_{f,t}$, $\pi_{i,q,t}$, $r_{i,t}$, $X_{i,t}$ and $W_{i,t}$;
3: Choose control decisions $d_{f,i,t}$, $x_{i,q,t}$, $c_{i,t}$, $u_{c,i,t}$, $u_{d,i,t}$, $e_{i,q,t}$, $g_{i,t}$, as the solution to **P5**;
4: Generate a new solution based on the following equations so that the constraint (14) could be satisfied: $\hat{u}_{c,i,t} = \max\{u_{c,i,t} - \frac{1}{\eta_{c,i}\eta_{d,i}} u_{d,i,t}, 0\}$, $\hat{u}_{d,i,t} = \max\{u_{d,i,t} - \eta_{c,i}\eta_{d,i} u_{c,i,t}, 0\}$, $\hat{d}_{f,i,t} = d_{f,i,t}$, $\hat{g}_{i,t} = g_{i,t}$, $\hat{e}_{i,q,t} = e_{i,q,t}$, $\hat{x}_{i,q,t} = x_{i,q,t}$, $\hat{c}_{i,t} = c_{i,t} + (\hat{u}_{c,i,t} - u_{c,i,t}) + (u_{d,i,t} - \hat{u}_{d,i,t})$.
5: Updating $Q_{i,q,t}$, $H_{i,q,t}$, $Z_{i,t}$ with the new solution according to (3), (25), and (24).
6: **End**

Remarks: Note that the constraints (7), (8), and (15) in **P1** are not considered in Algorithm 1, the solution generated by Algorithm 1 may be infeasible to **P1**. In Section V, we will show that Algorithm 1 can guarantee the feasibilities of (7), (8), and (15).

B. Distributed Implementation

To solve **P5** efficiently, we propose a distributed implementation for the proposed realtime algorithm. A possible way of obtaining a distributed algorithm for **P5** is based on dual decomposition, which decomposes the Lagrangian dual problem of **P5** into independent subproblems that could be solved in parallel. Unfortunately, the objective function in **P5** is not strictly convex since $\Gamma_{2,t}$ and $\Gamma_{3,t}$ are linear functions. As a result, dual decomposition cannot be applied, for otherwise the Lagrangian is unbounded below [9]. Since ADMM could be used to solve a large-scale convex optimization problem without assuming strict convexity of the separable objective function, we are thus motivated to design a ADMM-based distributed algorithm.

In order to utilize the ADMM framework, **P5** is transformed into the following problem equivalently.

$$(\textbf{P6}) \quad \min \ V\Gamma_t - \sum_{i=1}^{N}\sum_{q=1}^{M_i} w(Q_{i,q,t}+H_{i,q,t})b_{i,q,t}+$$

$$\sum_{i=1}^{N} Z_{i,t}(\eta_{c,i}u_{c,i,t} - \frac{1}{\eta_{d,i}}u_{d,i,t}) \tag{37a}$$

$$s.t. \ (1),(2),(4),(10)-(13),(18), \tag{37b}$$

$$\sum_{f=1}^{F} a_{f,i,t} + \sum_{q=1}^{M_i} (b_{i,q,t} - e_{i,q,t}) + h_i = C_i, \tag{37c}$$

$$g_{i,t} + c_{i,t} + u_{d,i,t} - u_{c,i,t} + \beta_i h_i = m_i, \tag{37d}$$

$$e_{i,q,t} + z_{i,q} = b_{i,q,t}, \tag{37e}$$

$$d_{f,i,t} = a_{f,i,t}, \tag{37f}$$

$$x_{i,q,t} = b_{i,q,t}, \tag{37g}$$

where h_i and $z_{i,q}$ are a set of nonnegative slack variables; $a_{f,i,t}$ and $b_{i,q,t}$ are nonnegative auxiliary variables; the constant $m_i = \alpha_i + \beta_i C_i - r_{i,t}$; the decision variables are $d_{f,i,t}, a_{i,q,t}, x_{i,q,t}, b_{i,q,t}, e_{i,q,t}, c_{i,t}, g_{i,t}, u_{c,i,t}, u_{d,i,t}, h_i, z_{i,q}$.

If ADMM framework applies to **P6** directly, eleven blocks would be generated since there are eleven kinds of variables. For ADMM with more than two blocks, the convergence is still an open question. In this paper, we adopt the algorithm in [31] to solve **P6**, which is called as ADM-G (ADM with Gaussian back substitution). The global convergence of ADM-G is provable under mild assumptions. Following the method in our previous work [32] [33], it is easy to check that ADM-G framework could result in an optimal solution of **P6** if the optimal solution is non-empty. Due to the space limit, we omit the proof for simplicity. Following the framework of ADM-G, we can obtain a distributed implementation of the proposed realtime algorithm in Appendix C.

IV. ALGORITHMIC PERFORMANCE ANALYSIS

In this section, we provide the performance analysis of the designed distributed realtime algorithm. Specifically, we first present a Lemma, which offers a sufficient condition for the charging and discharging of the ESS in SMG i at slot t under the proposed algorithm. Then, based on the Lemma, a Theorem is proposed to show the feasibility of the Algorithm 1 for **P1**.

Lemma 3. Define $\gamma_{i,\min}=\min\{X_{i,\min}, W_{i,\min}, A'_{i,\min}\}$. Then,

1) If $Z_{i,t} < -V\eta_{d,i}\gamma_{i,\max}$, the optimal discharging decision is $u^*_{d,i,t} = 0$,
2) If $Z_{i,t} > -\frac{V}{\eta_{c,i}}\gamma_{i,\min}$, the optimal charging decision is $u^*_{c,i,t} = 0$.

Proof: See Appendix D. ∎

With the above lemma, a theorem is provided to show the performance of the designed algorithm.

Theorem 1 Suppose $x^{\max}_{i,q} \geq \max[\pi^{\max}_{i,q}, \varepsilon_{i,q}]$. If $Q_{i,q,0} = H_{i,q,0} = 0$, the proposed algorithm can provide the following guarantees:

1) The queues $Q_{i,q,t}$ and $H_{i,q,t}$ are bounded by $Q^{\max}_{i,q}$ and $H^{\max}_{i,q}$, respectively. In particular, $Q^{\max}_{i,q} = V\beta_i X^{\max}_i/w + \pi^{\max}_{i,q}$, $H^{\max}_{i,q} = V\beta_i X^{\max}_i/w + \varepsilon_{i,q}$.
2) The maximum queueing delay $R^{\max}_{i,q} = \left\lceil \frac{2V\beta_i X^{\max}_i/w+\pi^{\max}_{i,q}+\varepsilon_{i,q}}{\varepsilon_{i,q}} \right\rceil$.
3) The energy queue $D_{i,t}$ satisfies the following for all time slot t: $D_{i,\min} \leq D_{i,t} \leq D_{i,\max}$.
4) The solution of the proposed algorithm is feasible to the original problem **P1**.
5) Compared with the optimal solution of **P3**, the maximum optimality loss due to the incorporation of ramping constraints in **P4** is $\Omega_1 = \sum_{i=1}^{N} V(1-\epsilon_i)c_{i,\max}\gamma_{i,\max}$.
6) Compared with the optimal solution of **P5**, the maximum optimality loss in the aspect of Γ_t caused by the online solution adjustment is $\Omega_2 = \sum_{i=1}^{N}\big(\sigma_i(u^2_{i,\mathrm{cmax}} + u^2_{i,\mathrm{dmax}}) + \delta_{1,i}c^2_{i,\max} + \delta_{2,i}c_{i,\max}\big)$.
7) If $\varepsilon_{i,q} \leq \mathbb{E}\{\pi_{i,q,t}\}$ and the uncertain parameters $\lambda_{f,t}$, $\pi_{i,q,t}$, $r_{i,t}$, $X_{i,t}$ and $W_{i,t}$ are i.i.d. over slots, the proposed algorithm offers the following performance guarantee, i.e., $\limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\{\Gamma_t\} \leq y_1 + \Omega_2 + \frac{\Omega_0+\Omega_1}{V}$, where y_1 is the optimal objective value of **P1**.

Proof: See Appendix E. ∎

V. PERFORMANCE EVALUATION

A. Simulation Setup

We intend to evaluate the performance of the proposed algorithm in six months with 4320 1-hour slots. To model the generation cost of conventional generator i, a quadratic polynomial is adopted as in [14], i.e., $A_i(c_{i,t}) = \delta_{1,i}c^2_{i,t}+\delta_{2,i}c_{i,t}+\delta_{3,i}$. For simplicity, we set $\delta_{1,i} = \delta_{3,i} = 0$, $\delta_{2,i} = 273\$/MW$ [29]. To model the battery depreciation cost, a function is considered as in [28], i.e., $B_i(u_{c,i,t}, u_{d,i,t}) = \sigma_i(u^2_{c,i,t}+u^2_{d,i,t})$. We set $\epsilon_i = 1$, $\sigma_i = 100$, $\eta_{c,i} = \eta_{d,i} = 1$. The parameters associated with data centers and front-end servers are given as follows, i.e., $F = 1$, $N = 3$, $M_1 = 40000$, $M_2 = 30000$, $M_3 = 30000$, $P_{i,\mathrm{peak}} = 200$ Watts, $P_{i,\mathrm{idle}} = 140$ Watts, $PUE_1 = 1.1$, $PUE_2 = 1.2$, $PUE_3 = 1.3$. $u_{i,\mathrm{cmax}} = u_{i,\mathrm{dmax}} = 0.5$ MW [4]. $\omega = 1\times10^{-4}$ [26], $\theta_i=0.1$, $V = V^{\max}$, $\varepsilon_{i,q} = (2V\beta_i X^{\max}_i/w + \pi^{\max}_{i,q})/(\mathcal{T}_{i,q} - 1)$, $x^{\max}_{i,q} = \pi^{\max}_{i,q}$. $D_{1,\max} = 8.8$ MWh, $D_{2,\max} = 7.2$ MWh, $D_{3,\max} = 7.8$ MWh (i.e., data centers could be supported by these ESSs for one hour). In addition, real-world workload traces[4] and

---

[4]http://ita.ee.lbl.gov/html/traces.html.

dynamic electricity price traces[5] are adopted in simulations. $W_{i,t} = 0.9X_{i,t}$ [13]. Suppose that there are two types of batch workloads, i.e., $M_i = 2$. To evaluate the impacts of tolerant service delays on the cost reduction under the proposed algorithm, two cases are considered, i.e., case1: $\mathcal{T}_{i,q} \in \{4, 8\}$; case2: $\mathcal{T}_{i,q} \in \{12, 24\}$. To model the batch workload with type q at data center i, we assume that it follows a uniform distribution with parameters 0 and $C_i/(5M_i)$.

To show the advantages of the proposed distributed realtime algorithm, three baselines are adopted.

- The first baseline (*B1*) intends to minimize the long-term operational cost with the considerations of energy storage and selling electricity, while batch workloads are processed immediately without delays.
- The second baseline (*B2*) intends to minimize the current operational cost considering selling electricity. Moreover, batch workloads are processed immediately. In addition, no energy storage is considered in *B2*.
- The three baseline (*B3*) intends to minimize the current operational cost without considering energy storage and selling electricity. Moreover, batch workloads are processed immediately.

For simplicity, *Proposed-1* and *Proposed-2* are adopted to denote the performance of the proposed algorithm under case1 and case2, respectively.

B. Simulation Results

1) Algorithmic feasibility: In this subsection, we show the feasibility of the proposed algorithm. Specifically, we need to show that the constraints (7), (8), (15) could be satisfied under the proposed algorithm. As indicated in Fig. 3 (a), the maximum queue lengths of $Q_{i,q,t}$ and $H_{i,q,t}$ are always smaller than their respective upper bounds (i.e., the constraint (7) holds in all time slots). Moreover, in Fig. 3 (b), maximum queueing delays are smaller than the corresponding tolerant service delays, which means that the proposed algorithm could provide the heterogeneous service delay guarantees for all batch workloads, i.e., (8) could be satisfied. In addition, the cumulative distribution functions (CDFs) of energy levels in ESSs are provided (note that just the results under *Proposed-2* with $w = 10^{-12}$ are given) in Fig. 3 (c), where energy levels fluctuate within their normal ranges, i.e., (15) could be guaranteed. Based on the above description, it can be known that the solution of the proposed algorithm is feasible to the original problem **P1**.

2) Convergence results: Before giving the performance comparisons between the proposed algorithm and other baselines, we first provide the convergence results of the proposed algorithm, which are illustrated in Figs. 4 (a)-(c). In Fig. 4 (a), the iterative process of the total operational cost in a time slot is shown, while Figs. 4 (b) and (c) show the trajectory of the primal residual and feasibility violation metric (which are defined in Appendix C), respectively. It can be observed that the proposed algorithm converges to the same optimal value (which is the same as the result generated by the
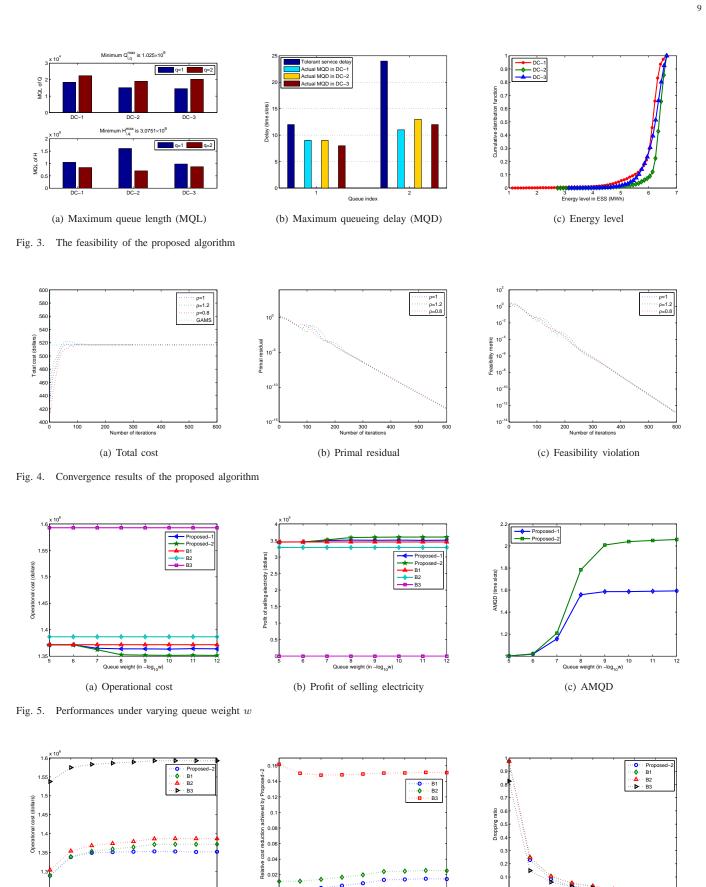
GAMS commercial solver[6]) given different penalty parameters ρ. Moreover, the computation complexity of the proposed algorithm is low since all subproblems in the distributed implementation could be solved in parallel based on closed-form expressions or binary search. Since we do not have enough hardware resources to conduct an experiment with a parallel implementation, the proposed algorithm is implemented on a single Intel Core i5-2410M 2.3GHz server (4G RAM), it takes 1.462 seconds to finish 600 iterations. Since the duration of a time slot is usually several minutes/hours (e.g., electricity prices in some deregulated electricity markets are updated every 5 minutes[7]), the time consumed by the proposed algorithm could be neglected when considering parallel implementation and "early braking" (i.e., terminating the algorithm before the convergence is reached once we obtain an acceptable solution, e.g., the primal residual and feasibility violation are small enough). Therefore, the proposed online distributed algorithm is very suitable for practical applications.

3) Queue weight w: In Fig. 5 (a), the operational costs under different algorithms are provided, and we find that the proposed algorithm achieves the best performance. Compared with *B1*, *B2*, and *B3*, *Proposed-2* with $w = 10^{-12}$ can reduce the operational cost by 1.48%, 2.55%, and 15.15%, respectively. The reason is that the proposed algorithm can fully utilize the temporal diversity of electricity price by serving batch workloads in proper time slots without violating their deadlines, by controlling the discharging/charging of ESSs in proper time slots, and by selling electricity to main grids when there are excess renewable energies. Thus, the proposed algorithm could obtain the largest profit of selling electricity among all algorithms as shown in Fig. 5 (b). In addition, it can be observed that larger w results in smaller AMQD (The Average value of Maximum Queueing Delays experienced by all workloads $\pi_{i,q,t}$), since larger w would lead to more frequent service for batch workloads as indicated in the objective function of **P5** in Appendix E, which means that less temporal diversity of electricity price could be utilized to reduce operational cost. Consequently, the proposed algorithm shows better performances given a smaller w.

4) Dropping penalty factor θ_i: We set $w = 10^{-12}$ in this scenario. In Figs. 6 (a) and (b), it can be seen that *Proposed-2* always achieves the lowest operational cost. By observing the objective function of **P6**, it can be known that the proposed algorithm intends to discard less batch workloads given a larger θ_i, resulting in a smaller dropping ratio (i.e., $\sum_i \sum_q \sum_t (e_{i,q,t}/a_{i,q,t})$) as shown in Fig. 6 (c). Therefore, the proposed algorithm would reduce to be *B1* if θ_i is approaching to zero, since all batch workloads would be dropped and no energy queue is needed under this situation.

5) Tolerant service delay $\mathcal{T}_{i,q}$: For simplicity, we assume that $\mathcal{T}_{i,q}$ is the same for all i and q. As shown in Figs. 7 (a) and (b), the operational cost becomes lower and the profit of selling electricity become larger with the increase of tolerant service delay if $w = 10^{-10}$, while those values are almost unchanged if $w = 10^{-5}$. The reason is that the proposed algorithm puts

---

(a) Maximum queue length (MQL)

(b) Maximum queueing delay (MQD)

(c) Energy level

Fig. 3. The feasibility of the proposed algorithm

(a) Total cost

(b) Primal residual

(c) Feasibility violation

Fig. 4. Convergence results of the proposed algorithm

(a) Operational cost

(b) Profit of selling electricity

(c) AMQD

Fig. 5. Performances under varying queue weight w

(a) Operational cost
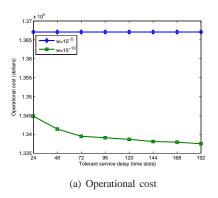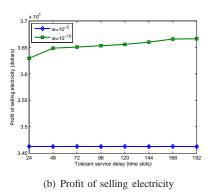
(b) Relative cost reduction

(c) Dropping ratio

Fig. 6. Performances under varying penalty factor θ_i

(a) Operational cost
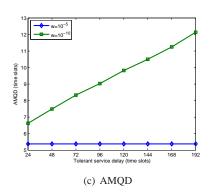
(b) Profit of selling electricity

(c) AMQD

Fig. 7. Performances under varying tolerant service delay $\mathcal{T}_{i,q}$

very large "weight" on maintaining the stability of workload queue $Q_{i,q,t}$ and virtual queue $H_{i,q,t}$ when $w = 10^{-5}$, resulting in very small queueing delay and AMQD as shown in Fig. 7 (c). Consequently, low utilization of temporal price diversity is incurred even the tolerant service delays of batch workloads are large. Thus, choosing a proper queue weight w is critical to utilize the heterogeneous tolerant service delays for operational cost reduction.

VI. Conclusions

This paper proposed a distributed realtime algorithm for minimizing the long-term operational cost of multiple data center microgrids with the considerations of many factors, e.g., providing heterogeneous service delay guarantees for batch workloads, interactive workload allocation, batch workload shedding, electricity buying/selling, battery charging/discharging efficiency, and the ramping constraints of backup generators. The proposed algorithm does not require any prior knowledge of statistical characteristics related to system parameters and has low computational complexity. Extensive simulation results showed that the proposed algorithm could reduce the operational cost of data center microgrids effectively.

APPENDIX A
PROOF OF LEMMA 1

Proof: Given a slot t, it can be proved that the energy demand $\pi_{i,q,t}$ could be satisfied before $t + R_{i,q}^{\max}$. If the above declaration is not true (a contradiction would be reached), we have $Q_{i,q,\tau} > x_{i,q,\tau}$ for all slots $\tau \in \{t+1, t+2, \cdots, t + R_{i,q}^{\max}\}$. According to (17), we can obtain that $H_{i,q,\tau+1} = [H_{i,q,\tau} - x_{i,q,\tau} + \varepsilon_{i,q}]^+$ for all slots $\tau \in \{t+1, t+2, \cdots, t + R_{i,q}^{\max}\}$. Continually, we have

$$H_{i,q,\tau+1} \geq H_{i,q,\tau} - x_{i,q,\tau} + \varepsilon_{i,q}, \tag{38}$$

Summing the above equation from slot $t + 1$ to $t + R_{i,q}^{\max}$, we have

$$H_{i,q,t+R_{i,q}^{\max}+1} - H_{i,q,t+1} \geq R_{i,q}^{\max}\varepsilon_{i,q} - \sum_{\tau=t+1}^{t+R_{i,q}^{\max}} x_{i,q,\tau}. \tag{39}$$

Since $H_{i,q,t+1} \geq 0$ and $H_{i,q,t+R_{i,q}^{\max}+1} \leq H_{i,q}^{\max}$, (39) could be transformed into (40),

$$\sum_{\tau=t+1}^{t+R_{i,q}^{\max}} x_{i,q,\tau} + H_{i,q}^{\max} \geq R_{i,q}^{\max}\varepsilon_{i,q}. \tag{40}$$

In addition, the summation of $x_{i,q,\tau}$ over the interval $\{t+1, t+2, \cdots, t + R_{i,q}^{\max}\}$ is strictly smaller than $Q_{i,q,t+1}$. Otherwise, $\pi_{i,q,t}$ would be served within the interval. Thus, we have

$$\sum_{\tau=t+1}^{t+R_{i,q}^{\max}} x_{i,q,\tau} < Q_{i,q,t+1} \leq Q_{i,q}^{\max}. \tag{41}$$

Finally, combining (40) and (41), we obtain

$$R_{i,q}^{\max} < \lceil (H_{i,q}^{\max} + Q_{i,q}^{\max})/\varepsilon_{i,q} \rceil. \tag{42}$$

Note that (42) contradicts the definition of $R_{i,q}^{\max}$. Thus, the workload batch $\pi_{i,q,t}$ must be served before $t + R_{i,q}^{\max}$. ∎

APPENDIX B
PROOF OF LEMMA 2

Proof: According to the definition of $Q_{i,q,t}$, we have

$$Q_{i,q,t+1}^2 = \left(\max\{Q_{i,q,t} - x_{i,q,t}, 0\} + \pi_{i,q,t} \right)^2$$
$$\leq Q_{i,q,t}^2 + x_{i,q,t}^2 + \pi_{i,q,t}^2 + 2Q_{i,q,t}(\pi_{i,q,t} - x_{i,q,t}).$$

Then, we can obtain

$$\frac{Q_{i,q,t+1}^2 - Q_{i,q,t}^2}{2} \leq \frac{(x_{i,q}^{\max})^2 + (\pi_{i,q}^{\max})^2}{2} + Q_{i,q,t}(\pi_{i,q,t} - x_{i,q,t}).$$

For the queue $H_{i,q,t}$, we have

$$H_{i,q,t+1}^2 \leq \left(\max[H_{i,q,t} - x_{i,q,t} + \varepsilon_{i,q}, 0] \right)^2$$
$$\leq \left(H_{i,q,t} - x_{i,q,t} + \varepsilon_{i,q} \right)^2.$$

Then, we have

$$\frac{H_{i,q,t+1}^2 - H_{i,q,t}^2}{2} \leq \frac{(\varepsilon_{i,q} - x_{i,q,t})^2}{2} + H_{i,q,t}(\varepsilon_{i,q} - x_{i,q,t}),$$
$$\leq \frac{\max\{\varepsilon_{i,q}^2, (x_{i,q}^{\max})^2\}}{2} + H_{i,q,t}(\varepsilon_{i,q} - x_{i,q,t}).$$

Similarly, for the queue $Z_{i,t}$, we have

$$\frac{Z_{i,t+1}^2 - Z_{i,t}^2}{2} \leq \frac{\max\{(\eta_{c,i}u_{i,\mathrm{cmax}})^2, (\frac{1}{\eta_{d,i}}u_{i,\mathrm{dmax}})^2\}}{2}$$
$$+ Z_{i,t}(\eta_{c,i}u_{c,i,t} - \frac{1}{\eta_{d,i}}u_{d,i,t}).$$

Combining three upper bounds mentioned above together, we have the following inequality,

$$\begin{aligned}\Delta_t \leq &\mathbb{E}\{\sum_{i=1}^{N}\sum_{q=1}^{M_i} wQ_{i,q,t}(\pi_{i,q,t} - x_{i,q,t})|\mathbf{\Theta_t}\}\\ &+ \mathbb{E}\{\sum_{i=1}^{N}\sum_{q=1}^{M_i} wH_{i,q,t}(\varepsilon_{i,q} - x_{i,q,t})|\mathbf{\Theta_t}\}\\ &+ \mathbb{E}\{\sum_{i=1}^{N} Z_{i,t}(\eta_{c,i}u_{c,i,t} - \frac{1}{\eta_{d,i}}u_{d,i,t})|\mathbf{\Theta_t}\} + \Omega_0.\end{aligned} \quad (43)$$

By adding $V\mathbb{E}\{\Gamma_t|\mathbf{\Theta_t}\}$ to the both sides of the above equation, we could complete the proof. ∎

APPENDIX C
THE DISTRIBUTED IMPLEMENTATION OF ALGORITHM 1

Proof: **1. Initialization**: Decision variables of **P6** are initialized with zero. In each iteration k, two steps (i.e., prediction step and correction step) are repeated until convergence.

2. ADMM step (prediction step). Obtain all decision variables in the forwarding order:

2.1 $d_{f,i,t}$**-minimization**: each front-end server f solves **P7** in parallel to obtain $\tilde{d}_{f,i,t}^k$.

$$(\mathbf{P7}) \quad \min \quad \Phi_1(d_{f,i,t}, \chi_{f,i}^k, a_{f,i,t}^k) \quad (44a)$$
$$s.t. \ (1), (2), \quad (44b)$$

where $\Phi_1(d_{f,i,t}, \chi_{i,q}^k, a_{f,i,t}^k) = \sum_{i=1}^{N}((V\omega L_{f,i} + \chi_{f,i}^k - \rho a_{f,i,t}^k)d_{f,i,t} + \frac{\rho}{2}d_{f,i,t}^2)$; ρ is the penalty parameter in the augmented Lagrangian for **P6**, while $\phi_i, \varphi_i, \kappa_{i,q}, \chi_{f,i}, \psi_{i,q}$ are dual variables associated with (37c,-37g), respectively.

2.2 $x_{i,q,t}$**-minimization**: each queue controller q in data center i solves **P8** in parallel to obtain $\tilde{x}_{i,q,t}^k$.

$$(\mathbf{P8}) \quad \min \quad \Phi_2(x_{i,q,t}, \psi_{i,q}^k, b_{i,q,t}^k) \quad (45a)$$
$$s.t. \ (4), \quad (45b)$$

where $\Phi_2(x_{i,q,t}, \psi_{i,q}^k, b_{i,q,t}^k) = \psi_{i,q}^k x_{i,q,t} + \frac{\rho}{2}(x_{i,q,t} - b_{i,q,t}^k)^2$.

2.3 $c_{i,t}$**-minimization**: each conventional generator in SMG i solves **P9** in parallel to obtain $\tilde{c}_{i,t}^k$.

$$(\mathbf{P9}) \quad \min \quad \Phi_3(c_{i,t}, \varphi_i^k, g_{i,t}^k, u_{d,i,t}^k, u_{c,i,t}^k, h_i^k) \quad (46a)$$
$$s.t. \ (10), (11), \quad (46b)$$

where $\Phi_3(c_{i,t}, \varphi_i^k, g_{i,t}^k, u_{d,i,t}^k, u_{c,i,t}^k, h_i^k) = VA_i(c_{i,t}) + \frac{\rho}{2}c_{i,t}^2 + (\varphi_i^k + \rho(g_{i,t}^k + u_{d,i,t}^k - u_{c,i,t}^k + \beta_i h_i^k - m_i))c_{i,t}$.

2.4 $u_{c,i,t}$**-minimization**: each ESS in SMG i solves **P10** in parallel to obtain $\tilde{u}_{c,i,t}^k$.

$$(\mathbf{P10}) \quad \min \quad \Phi_4(u_{c,i,t}, \varphi_i^k, u_{d,i,t}^k, g_{i,t}^k, \tilde{c}_{i,t}^k, h_i^k) \quad (47a)$$
$$s.t. \ (12), \quad (47b)$$

where $\Phi_4(u_{c,i,t}, \varphi_i^k, u_{d,i,t}^k, g_{i,t}^k, \tilde{c}_{i,t}^k, h_i^k) = \frac{\rho}{2}u_{c,i,t}^2 + VB_i(u_{c,i,t}, u_{d,i,t}^k) + (Z_{i,t}\eta_{c,i} - \varphi_i^k - \rho(g_{i,t}^k + \tilde{c}_{i,t}^k + u_{d,i,t}^k + \beta_i h_i^k - m_i))u_{c,i,t}$.

2.5 $u_{d,i,t}$**-minimization**: each ESS in SMG i solves **P11** in parallel to obtain $\tilde{u}_{d,i,t}^k$.

$$(\mathbf{P11}) \quad \min \quad \Phi_5(u_{d,i,t}, \tilde{u}_{c,i,t}^k, \varphi_i^k, g_{i,t}^k, \tilde{c}_{i,t}^k, h_i^k) \quad (48a)$$
$$s.t. \ (13), \quad (48b)$$

where $\Phi_5(u_{d,i,t}, \tilde{u}_{c,i,t}^k, \varphi_i^k, g_{i,t}^k, \tilde{c}_{i,t}^k, h_i^k) = \frac{\rho}{2}u_{d,i,t}^2 + VB_i(\tilde{u}_{c,i,t}^k, u_{d,i,t}) - (Z_{i,t}/\eta_{d,i} - \varphi_i^k - \rho(g_{i,t}^k + \tilde{c}_{i,t}^k - \tilde{u}_{c,i,t}^k + \beta_i h_i^k - m_i))u_{d,i,t}$.

2.6 $a_{f,i,t}$**-minimization**: each EMS in SMG i solves **P12** in parallel to obtain $\tilde{a}_{f,i,t}^k$.

$$(\mathbf{P12}) \quad \min \quad \Phi_6(a_{f,i,t}, \phi_i^k, b_{i,q,t}^k, e_{i,q,t}^k, h_i^k, \chi_{f,i}^k, \tilde{d}_{f,i,t}^k) \quad (49a)$$
$$s.t. \ a_{f,i,t} \geq 0, \quad (49b)$$

where $\Phi_6(a_{f,i,t}, \phi_i^k, b_{i,q,t}^k, e_{i,q,t}^k, h_i^k, \chi_{f,i}^k, \tilde{d}_{f,i,t}^k) = \frac{\rho}{2}(\sum_{f=1}^{F} a_{f,i,t}^2 + (\sum_{f=1}^{F} a_{f,i,t})^2) + \sum_{f=1}^{F}(\phi_i^k - \chi_{f,i}^k - \rho\tilde{d}_{f,i,t}^k + \rho(\sum_{q=1}^{M_i}(b_{i,q,t}^k - e_{i,q,t}^k) + h_i^k - C_i)a_{f,i,t}$.

2.7 $b_{i,q,t}$**-minimization**: each EMS in SMG i solves **P13** in parallel to obtain $\tilde{b}_{i,q,t}^k$.

$$(\mathbf{P13}) \quad \min \quad \Phi_7(b_{i,q,t}, \phi_i^k, \kappa_{i,q}^k, \psi_{i,q}^k, \tilde{a}_{f,i,t}^k, e_{i,q,t}^k, h_i^k, z_{i,q}^k) \quad (50a)$$
$$s.t. \ b_{i,q,t} \geq 0, \quad (50b)$$

where $\Phi_7(b_{i,q,t}, \phi_i^k, \kappa_{i,q}^k, \tilde{a}_{f,i,t}^k, e_{i,q,t}^k, h_i^k, z_{i,q}^k) = \frac{\rho}{2}(\sum_{q=1}^{M_i} 2b_{i,q,t}^2 + (\sum_{q=1}^{M_i} b_{i,q,t})^2) - \sum_{q=1}^{M_i}(w(Q_{i,q,t} + H_{i,q,t}) - \phi_i^k + \kappa_{i,q}^k + \psi_{i,q}^k + \rho(e_{i,q,t}^k + z_{i,q}^k + \tilde{x}_{i,q,t}^k) - \rho(\sum_{f=1}^{F}\tilde{a}_{f,i,t}^k - \sum_{q=1}^{M_i}e_{i,q,t}^k + h_i^k - C_i))b_{i,q,t}$.

2.8 $e_{i,q,t}$**-minimization**: each EMS in SMG i solves **P14** in parallel to obtain $\tilde{e}_{i,q,t}^k$.

$$(\mathbf{P14}) \quad \min \quad \Phi_8(e_{i,q,t}, \kappa_{i,q}^k, \phi_i^k, z_{i,q}^k, \tilde{b}_{i,q,t}^k, \tilde{a}_{f,i,t}^k, h_i^k) \quad (51a)$$
$$s.t. \ e_{i,q,t} \geq 0, \quad (51b)$$

where $\Phi_8(e_{i,q,t}, \kappa_{i,q}^k, \phi_i^k, z_{i,q}^k, \tilde{b}_{i,q,t}^k, \tilde{a}_{f,i,t}^k, h_i^k) = \frac{\rho}{2}(\sum_{q=1}^{M_i} e_{i,q,t}^2 + (\sum_{q=1}^{M_i} e_{i,q,t})^2) + \sum_{q=1}^{M_i}(V\theta_i - \phi_i^k + \kappa_{i,q}^k + \rho(z_{i,q}^k - \tilde{b}_{i,q,t}^k) - \rho(\sum_{f=1}^{F}\tilde{a}_{f,i,t}^k + \sum_{q=1}^{M_i}\tilde{b}_{i,q,t}^k + h_i^k - C_i))e_{i,q,t}$.

2.9 h_i**-minimization**: each EMS in SMG i solves **P15** in parallel to obtain $\tilde{h}_i^k$.

$$(\mathbf{P15}) \quad \min \quad \Phi_9(h_i, \phi_i^k, \varphi_i^k, g_{i,t}^k, \tilde{c}_{i,t}^k, \tilde{u}_{c,i,t}^k, \tilde{u}_{d,i,t}^k, \tilde{a}_{f,i,t}^k, \tilde{b}_{i,q,t}^k, \tilde{e}_{i,q,t}^k) \quad (52a)$$
$$s.t. \ h_i \geq 0, \quad (52b)$$

where $\Phi_9(h_i, \phi_i^k, \varphi_i^k, g_{i,t}^k, \tilde{c}_{i,t}^k, \tilde{u}_{c,i,t}^k, \tilde{u}_{d,i,t}^k, \tilde{a}_{f,i,t}^k, \tilde{b}_{i,q,t}^k, \tilde{e}_{i,q,t}^k) = \frac{\rho}{2}(1 + \beta_i^2)h_i^2 + (\phi_i^k + \beta_i\varphi_i^k + \rho\beta_i(g_{i,t}^k + \tilde{c}_{i,t}^k + \tilde{u}_{d,i,t}^k - \tilde{u}_{c,i,t}^k - m_i) + \rho(\sum_{f=1}^{F}\tilde{a}_{f,i,t}^k + \sum_{q=1}^{M_i}(\tilde{b}_{i,q,t}^k - \tilde{e}_{i,q,t}^k) - C_i))h_i$.

2.10 $z_{i,q}$**-minimization**: each EMS in SMG i solves **P16** in parallel to obtain $\tilde{z}_{i,q}^k$.

$$(\mathbf{P16}) \quad \min \quad \Phi_{10}(z_{i,q}, \tilde{e}_{i,q,t}^k, \tilde{b}_{i,q,t}^k, \kappa_{i,q}^k) \quad (53a)$$
$$s.t. \ z_{i,q} \geq 0, \quad (53b)$$

where $\Phi_{10}(z_{i,q}, \tilde{e}^k_{i,q,t}, \tilde{b}^k_{i,q,t}, \kappa^k_{i,q}) = \frac{\rho}{2} z^2_{i,q} + (\rho(\tilde{e}^k_{i,q,t} - \tilde{b}^k_{i,q,t}) + \kappa^k_{i,q}) z_{i,q}$.

2.11 $g_{i,t}$-**minimization**: each EMS in SMG i solves **P17** in parallel to obtain $\tilde{g}^k_{i,t}$.

$$(\textbf{P17}) \quad \min \quad \Phi_{11}(g_{i,t}, \varphi^k_i, \tilde{c}^k_{i,t}, \tilde{u}^k_{c,i,t}, \tilde{u}^k_{d,i,t}, \tilde{h}^k_i) \quad (54a)$$
$$s.t. \ (18), \quad\quad\quad (54b)$$

where $\Phi_{11}(g_{i,t}, \tilde{u}^k_{c,i,t}, \tilde{u}^k_{d,i,t}, \tilde{h}^k_i) = \frac{\rho}{2} g^2_{i,t} + (\varphi^k_i + \rho(\tilde{c}^k_{i,t} + \tilde{u}^k_{d,i,t} - \tilde{u}^k_{c,i,t} + \beta_i \tilde{h}^k_i - m_i)) g_{i,t} + \frac{X_{i,t} - W_{i,t}}{2} |g_{i,t}| + \frac{X_{i,t} + W_{i,t}}{2} g_{i,t}$.

Note that **P7**-**P17** are convex optimization problems and their solutions could be obtained easily based on closed-form expressions or binary search. Thus, the algorithms for them are omitted for brevity. Similar algorithms could be found in [33].

2.12 Dual update: the EMS in SMG i updates $\tilde{\phi}^k_i, \tilde{\varphi}^k_i, \tilde{\kappa}^k_{i,q}$ as follows: $\tilde{\phi}^k_i = \phi^k_i + \rho(\sum^F_{f=1} \tilde{a}_{f,i,t} + \sum^{M_i}_{q=1}(\tilde{b}_{i,q,t} - \tilde{e}_{i,q,t}) + \tilde{h}_i - C_i)$; $\tilde{\varphi}^k_i = \varphi^k_i + \rho(\tilde{g}^k_{i,t} + \tilde{c}^k_{i,t} + \tilde{u}^k_{d,i,t} - \tilde{u}^k_{c,i,t} + \beta_i \tilde{h}^k_i - m_i)$; $\tilde{\kappa}^k_{i,q} = \kappa^k_{i,q} + \rho(\tilde{e}^k_{i,q,t} + \tilde{z}^k_{i,q} - \tilde{b}^k_{i,q,t})$; each front-end server f updates $\tilde{\chi}^k_{f,i}$ as follows, i.e., $\tilde{\chi}^k_{f,i} = \chi^k_{f,i} + \rho(\tilde{d}^k_{f,i,t} - \tilde{a}^k_{f,i,t})$; each queue controller q in data center i updates $\tilde{\psi}^k_{i,q}$ as follows: $\tilde{\psi}^k_{i,q} = \psi^k_{i,q} + \rho(\tilde{x}^k_{i,q,t} - \tilde{b}^k_{i,q,t})$.

3. Gaussian back substitution step (correction step):
Obtain the input parameters of iteration $k + 1$ according to the Gaussian back substitution step (3.5b) in [31], where the constant α in (3.5b) is set to one based on the practical experience [33]. Then, we have

$$\phi^{k+1}_i = \tilde{\phi}^k_i, \varphi^{k+1}_i = \tilde{\varphi}^k_i, \kappa^{k+1}_{i,q} = \tilde{\kappa}^k_{i,q},$$
$$\tilde{\chi}^k_{f,i} = \chi^k_{f,i}, \ \psi^{k+1}_{i,q} = \tilde{\psi}^k_{i,q}, g^{k+1}_{i,t} = \tilde{g}^k_{i,t}, z^{k+1}_{i,q} = \tilde{z}^k_{i,q},$$
$$h^{k+1}_i = \tilde{h}^k_i - \frac{\beta_i}{1+\beta^2_i}(\tilde{g}^k_{i,t} - g^k_{i,t}),$$
$$e^{k+1}_{i,q,t} = \tilde{e}^k_{i,q,t} + \frac{(h^{k+1}_i - h^k_i) + \sum^{M_i}_{q=1}(\tilde{z}^k_{i,q} - z^k_{i,q})}{M_i + 1} - (\tilde{z}^k_{i,q} - z^k_{i,q})$$
$$b^{k+1}_{i,q,t} = \tilde{b}^k_{i,q,t} + \frac{\sum^{M_i}_{q=1}(e^{k+1}_{i,q,t} - e^k_{i,q,t} - z^{k+1}_{i,q} + z^k_{i,q}) - 2(h^{k+1}_i - h^k_i)}{2(M_i + 2)}$$
$$+ \frac{1}{2}(z^{k+1}_{i,q} - z^k_{i,q} + e^{k+1}_{i,q,t} - e^k_{i,q,t}),$$
$$a^{k+1}_{f,i,t} = \tilde{a}^k_{f,i,t} + \frac{\sum^{M_i}_{q=1}(e^{k+1}_{i,q,t} - e^k_{i,q,t} - b^{k+1}_{i,q,t} + b^k_{i,q,t}) - (h^{k+1}_i - h^k_i)}{F + 1}$$
$$u^{k+1}_{d,i,t} = \tilde{u}^k_{d,i,t} - \beta_i(h^{k+1}_i - h^k_i) - (g^{k+1}_{i,t} - g^k_{i,t}),$$
$$u^{k+1}_{c,i,t} = \tilde{u}^k_{c,i,t} + (\tilde{u}^k_{d,i,t} - u^k_{d,i,t}), \ c^{k+1}_{i,t} = \tilde{c}^k_{i,t} + (\tilde{u}^k_{c,i,t} - u^k_{c,i,t}),$$
$$x^{k+1}_{i,q,t} = \tilde{x}^k_{i,q,t} + (b^{k+1}_{i,q,t} - b^k_{i,q,t}), \ d^{k+1}_{f,i,t} = \tilde{d}^k_{f,i,t}.$$

4. Stopping criterion: As in our previous work [33], we terminate the designed algorithm before the convergence is reached once we obtain an acceptable feasible solution, e.g., when the primal residual Ξ is small enough and the obtained solution is feasible. Specifically, the primal residual is defined in (55). Moreover, a feasibility metric ℓ is adopted as in (56) to indicate the feasibility of the obtained solution.

$$\Xi^2 = \sum^N_{i=1}\Big(\sum^F_{f=1} a^k_{f,i,t} + \sum^{M_i}_{q=1}(b^k_{i,q,t} - e^k_{i,q,t}) + h^k_i - C_i\Big)^2$$
$$+ \sum^N_{i=1}\Big(g^k_{i,t} + c^k_{i,t} + u^k_{d,i,t} - u^k_{c,i,t} + \beta_i h^k_i - m_i\Big)^2$$
$$+ \sum^N_{i=1}\sum^{M_i}_{q=1}\Big(e^k_{i,q,t} + z^k_{i,q} - b^k_{i,q,t}\Big)^2$$
$$+ \sum^N_{i=1}\sum^{M_i}_{q=1}\Big(x^k_{i,q,t} - b^k_{i,q,t}\Big)^2$$
$$+ \sum^N_{i=1}\sum^F_{f=1}\Big(d^k_{f,i,t} - a^k_{f,i,t}\Big)^2. \quad (55)$$

$$\ell = \sum^N_{i=1} \max\Big(\sum^F_{f=1} d^k_{f,i,t} + \sum^{M_i}_{q=1}(x^k_{i,q,t} - e^k_{i,q,t}) - C_i, 0\Big)$$
$$+ \sum^N_{i=1}\Big|g^k_{i,t} + r_{i,t} + c^k_{i,t} + u^k_{d,i,t} - p^k_{i,t} - u^k_{c,i,t}\Big|$$
$$+ \sum^N_{i=1}\sum^{M_i}_{q=1} \max\Big(e^k_{i,q,t} - x^k_{i,q,t}, 0\Big). \quad (56)$$

Note that the implementation of one iteration in the proposed algorithm could be described as follows. At initial iteration, each front-end server f, each queue controller q in data center i, each conventional generator in SMG i make their local and parallel decisions independently to obtain $\tilde{a}^k_{f,i,t}$, $\tilde{x}^k_{i,q,t}$, $\tilde{c}^k_{i,t}$, respectively. Then, such decisions are broadcasted to other components in SMGs, e.g., ESSs and EMS. After receiving the broadcasted decisions, each ESS and EMS in SMG i make their local decisions on $\tilde{a}^k_{f,i,t}$ and $\tilde{u}^k_{c,i,t}$, and $\tilde{u}^k_{d,i,t}$. Then, ESS i broadcasts $\tilde{u}^k_{c,i,t}$, and $\tilde{u}^k_{d,i,t}$ to the EMS i. Next, EMS i could obtains other decisions on $\tilde{b}^k_{i,q,t}, \tilde{e}^k_{i,q,t}, \tilde{h}^k_i$, $\tilde{z}^k_{i,q}$ and $\tilde{g}^k_{i,t}$. Finally, EMS i broadcasts all obtained decision variables at iteration $k+1$ (i.e., $g^{k+1}_{i,t}, z^{k+1}_{i,q}, h^{k+1}_i, e^{k+1}_{i,q,t}, b^{k+1}_{i,q,t}$, $a^{k+1}_{f,i,t}, u^{k+1}_{d,i,t}, u^{k+1}_{c,i,t}$) so that all entities (i.e., front-end servers, queue controllers, conventional generators, and ESSs) could update their respective decisions in iteration $k + 1$ according to the Gaussian back substitution step. ∎

APPENDIX D
PROOF OF LEMMA 3

Proof:

1) Let $(u^*_{c,i,t}, u^*_{d,i,t}, x^*_{i,q,t}, e^*_{i,q,t}, d^*_{f,i,t}, \pi^*_{i,t}, g^*_{i,t}, c^*_{i,t})$ be the optimal decision vector obtained from the Algorithm 1. For SMG i, suppose $Z_{i,t} < -V\eta_{d,i}\gamma_{i,\max}$ and $u^*_{d,i,t} > 0$, then, $u^*_{c,i,t} = 0$. Then, we can prove the non-optimality of the above decision by choosing another decision vector $(0, 0, x^*_{i,q,t}, e^*_{i,q,t}, d^*_{f,i,t}, \pi^*_{i,t}, \tilde{g}^*_{i,t}, \tilde{c}^*_{i,t})$. Suppose the objective values corresponding to the above decision vectors under the Algorithm 1 are $\Upsilon_{1,i}$ and $\Upsilon_{2,i}$, respectively. Given the same energy demand $p_{i,t}$,

there are three kinds of the decisions for energy supply:

Case 1: If $g_{i,t}^* = 0$, we choose $\tilde{g}_{i,t}^* = 0$, then, $\tilde{c}_{i,t}^* = c_{i,t}^* + u_{d,i,t}^*$. Next, $\Upsilon_{1,i} - \Upsilon_{2,i} > (-\frac{Z_{i,t}}{\eta_{d,i}} - VA'_{i,\max})u_{d,i,t}^* > 0$.

Case 2: If $g_{i,t}^* > 0$, we choose $\tilde{g}_{i,t}^* = 0$, $\tilde{c}_{i,t}^* = c_{i,t}^*$, then, $\tilde{g}_{i,t}^* = g_{i,t}^* + u_{d,i,t}^*$. Next, $\Upsilon_{1,i} - \Upsilon_{2,i} > (-\frac{Z_{i,t}}{\eta_{d,i}} - VX_{i,\max})u_{d,i,t}^* > 0$.

Case 3: If $g_{i,t}^* < 0$, we choose $\tilde{g}_{i,t}^* = 0$, $\tilde{c}_{i,t}^* = c_{i,t}^*$, then, $\tilde{g}_{i,t}^* = g_{i,t}^* - u_{d,i,t}^*$. Next, $\Upsilon_{1,i} - \Upsilon_{2,i} > (-\frac{Z_{i,t}}{\eta_{d,i}} - VW_{i,\max})u_{d,i,t}^* > 0$.

In summary, when $Z_{i,t} < -V\eta_{d,i}\gamma_{i,\max}$, the optimal discharging decision is $u_{d,i,t}^* = 0$.

2) The proof of part 2 is similar to that of part 1. Thus, it is omitted for brevity. ∎

APPENDIX E
PROOF OF THEOREM 1

Proof:

1) The objective value of **P5** could be rewritten as follows by discarding some constant items,

$$\bigsqcup(d_{f,i,t}, e_{i,q,t}, c_{i,t}, g_{i,t}, u_{c,i,t}, u_{d,i,t})$$
$$+ \sum_{i=1}^{N}\sum_{q=1}^{M_i}(V\beta_i X_{i,t} - w(Q_{i,q,t} + H_{i,q,t}))x_{i,q,t}\mathcal{I}_1,$$
$$+ \sum_{i=1}^{N}\sum_{q=1}^{M_i}(V\beta_i W_{i,t} - w(Q_{i,q,t} + H_{i,q,t}))x_{i,q,t}\mathcal{I}_2,$$
$$+ \sum_{i=1}^{N}\sum_{q=1}^{M_i}(-w(Q_{i,q,t} + H_{i,q,t}))x_{i,q,t}\mathcal{I}_3,$$

where $\bigsqcup(\upsilon)$ is the function of υ; $\mathcal{I}_1$, $\mathcal{I}_2$, $\mathcal{I}_3$ denote $g_{i,t} > 0$, $g_{i,t} < 0$, and $g_{i,t} = 0$, respectively. It can be observed that the proposed algorithm would choose the maximum possible $x_{i,q,t}$ when $Q_{i,q,t} > V\beta_i X_i^{\max}/w$. In the following parts, we would use the induction method to prove $Q_{i,q}^{\max} = V\beta_i X_i^{\max}/w + \pi_{i,q}^{\max}$ for all slots. It is obvious that $Q_{i,q,0} \leq Q_{i,q}^{\max}$. Suppose $Q_{i,q,t} \leq Q_{i,q}^{\max}$, we will show that $Q_{i,q,t+1} \leq Q_{i,q}^{\max}$. If $Q_{i,q,t} \leq V\beta_i X_i^{\max}/w$, the maximum queue growth is $\pi_{i,q}^{\max}$. Thus, we have $Q_{i,q,t+1} \leq Q_{i,q,t} + \pi_{i,q}^{\max} \leq V\beta_i X_i^{\max}/w + \pi_{i,q}^{\max}$. If $Q_{i,q,t} \geq V\beta_i X_i^{\max}/w$, the proposed algorithm would choose $x_{i,q,t} = \min\{Q_{i,q,t}, x_{i,q}^{\max}\}$. Thus, $Q_{i,q,t+1} \leq \max\{Q_{i,q,t}, \pi_{i,q}^{\max}\} \leq Q_{i,q}^{\max}$. Similarly, we can prove that $Z_{i,q,t} \leq Z_{i,q}^{\max}$ for any slot t. The proof detail is omitted for brevity. Continually, it can be known that (7) could be satisfied.

2) According to Lemma 1 and the part 1 of Theorem 1, we have $R_{i,q}^{\max} = \lceil(2V\beta_i X_{i,q}^{\max}/w + \pi_{i,q}^{\max} + \varepsilon_{i,q})/\varepsilon_{i,q}\rceil$. Therefore, we can construct an algorithm to ensure that all charging requests have delay less than or equal to $R_{i,q}^{\max}$ slots, where $R_{i,q}^{\max} \geq 2$. When choosing $x_{i,q,t} = x_{i,q}^{\max}$ in each time slot t, we can guarantee that all charging requests have one slot delay. In summary,

the proposed algorithm could be constructed to ensure the heterogeneous service delays for all EV charging requests, i.e., (8) could be satisfied under the proposed algorithm.

3) Proving $D_{i,t} \in [D_{i,\min}, D_{i,\max}]$ is equivalent to satisfying the following constraints: $Z_{i,t} \geq -V\eta_{d,i}\gamma_{i,\max} - \frac{1}{\eta_{d,i}}u_{i,\text{dmax}}$, and $Z_{i,t} \leq D_{i,\max} - D_{i,\min} - V\eta_{d,i}\gamma_{i,\max} - \frac{1}{\eta_{d,i}}u_{i,\text{dmax}}$. Because $D_{i,\min} \leq D_{i,0} \leq D_{i,\max}$, the above inequalities hold for $t=0$. Suppose the above-mentioned inequalities hold for the time slot t, we should verify that they hold for the time slot $t+1$.

- If $-V\eta_{d,i}\gamma_{i,\max} - \frac{1}{\eta_{d,i}}u_{i,\text{dmax}} \leq Z_{i,t} < -V\eta_{d,i}\gamma_{i,\max}$, then, according to the Lemma 3, $u_{d,i,t}^* = 0$. As a result, $Z_{i,t+1} = Z_{i,t} + \eta_{c,i}u_{c,i,t}^* \geq Z_{i,t} \geq -V\eta_{d,i}\gamma_{i,\max} - \frac{1}{\eta_{d,i}}u_{i,\text{dmax}}$. If $-V\eta_{d,i}\gamma_{i,\max} \leq Z_{i,t} < D_{i,\max} - D_{i,\min} - V\eta_{d,i}\gamma_{i,\max} - \frac{1}{\eta_{d,i}}u_{i,\text{dmax}}$, then, $Z_{i,t+1} \geq -V\eta_{d,i}\gamma_{i,\max} - \frac{1}{\eta_{d,i}}u_{d,i,t}^* > -V\eta_{d,i}\gamma_{i,\max} - \frac{1}{\eta_{d,i}}u_{i,\text{dmax}}$.

- If $-\frac{V}{\eta_{d,i}}\gamma_{i,\min} < Z_{i,t} \leq D_{i,\max} - D_{i,\min} - V\eta_{d,i}\gamma_{i,\max} - \frac{1}{\eta_{d,i}}u_{i,\text{dmax}}$, then, $u_{c,i,t}^* = 0$. Consequently, $Z_{i,t+1} \leq Z_{i,t} \leq D_{i,\max} - D_{i,\min} - V\eta_{d,i}\gamma_{i,\max} - \frac{1}{\eta_{d,i}}u_{i,\text{dmax}}$. If $-V\eta_{d,i}\gamma_{i,\max} - \frac{1}{\eta_{d,i}}u_{i,\text{dmax}} \leq Z_{i,t} \leq -\frac{V}{\eta_{c,i}}\gamma_{i,\min}$, then, $Z_{i,t+1} \leq -\frac{V}{\eta_{c,i}}\gamma_{i,\min} + \eta_{c,i}u_{i,\text{cmax}} \leq D_{i,\max} - D_{i,\min} - V\eta_{d,i}\gamma_{i,\max} - \frac{1}{\eta_{d,i}}u_{i,\text{dmax}}$, where

$$V \leq \frac{D_{i,\max} - D_{i,\min} - (\eta_{c,i}u_{i,\text{cmax}} + \frac{1}{\eta_{d,i}}u_{i,\text{dmax}})}{\eta_{d,i}\gamma_{i,\max} - \frac{1}{\eta_{c,i}}\gamma_{i,\min}}.$$

Continually, $V_{\max}$ is obtained as follows,

$$V_{\max} = \min_i \frac{D_{i,\max} - D_{i,\min} - (\eta_{c,i}u_{i,\text{cmax}} + \frac{1}{\eta_{d,i}}u_{i,\text{dmax}})}{\eta_{d,i}\gamma_{i,\max} - \frac{1}{\eta_{c,i}}\gamma_{i,\min}}.$$

Based on the above proof, it can be known that (15) could be satisfied.

4) From the parts 1-3, we know that the constraints (7),(8),(15) could be satisfied under the proposed algorithm. Since other constraints in **P1** could be guaranteed, the solution of the proposed algorithm is feasible to the original problem **P1**.

5) Let $(x_{i,q,t}^*, u_{c,i,t}^*, u_{d,i,t}^*, d_{f,i,t}^*, \pi_{i,t}^*, g_{i,t}^*, c_{i,t}^*)$ and $(\bar{x}_{i,q,t}, \bar{u}_{c,i,t}, \bar{u}_{d,i,t}, \bar{d}_{f,i,t}, \bar{\pi}_{i,t}, \bar{g}_{i,t}, \bar{c}_{i,t})$ denote the optimal solution of **P3** and **P4**, respectively. Since the adoption of ramping constraints in **P4** would or would not change the value of $\bar{c}_{i,t}$, three cases would be incurred.

Case 1: when $c_{i,t}^* = \bar{c}_{i,t}$, we have $y_{3,i}^* = \bar{y}_{4,i}$, where $y_{3,i}^*$ and $\bar{y}_{4,i}$ are the optimal objective value associated with the SMG i, respectively.

Case 2: when $c_{i,t}^* > \bar{c}_{i,t}$, the effective range of $c_{i,t}$ in **P4** is $\max\{c_{i,t-1} - \epsilon_i c_{i,\max}, 0\} \leq c_{i,t} \leq c_{i,t-1} + \epsilon_i c_{i,\max}$. We choose a feasible solution to **P4** as follows, i.e., $(x_{i,q,t}^*, u_{c,i,t}^*, u_{d,i,t}^*, d_{f,i,t}^*, \pi_{i,t}^*, g_{i,t}^* + c_{i,t}^* - c_{i,t-1} - \epsilon_i c_{i,\max}, c_{i,t-1} + \epsilon_i c_{i,\max})$, which means that the conventional generator must generate less energy due to the ramping constraint and more energy should be purchased from the main grid i to balance power. Then, we have

$\bar{y}_{4,i} - y^*_{3,i} \le V(1-\epsilon_i)c_{i,\max}X_{i,\max}.$

Case 3: when $c^*_{i,t} < \bar{c}_{i,t}$, the effective range of $c_{i,t}$ in **P4** is $c_{i,t-1} - \epsilon_i c_{i,\max} \le c_{i,t} \le \min\{c_{i,\max}, c_{i,t-1} + \epsilon_i c_{i,\max}\}$. Set a feasible solution of **P4** as $(x^*_{i,q,t}, u^*_{c,i,t}, u^*_{d,i,t}, d^*_{f,i,t}, \pi^*_{i,t}, g^*_{i,t} + c^*_{i,t} - c_{i,t-1} + \epsilon_i c_{i,\max}, c_{i,t-1} - \epsilon_i c_{i,\max})$, which means that the conventional generator must generate more energy due to the ramping constraint and more energy should be sold to the main grid i to balance power. As a result, $\bar{y}_{4,i} - y^*_{3,i} \le V(1-\epsilon_i)c_{i,\max}A'_{i,\max}.$

In summary, $\bar{y}_4 \le y^*_3 + \sum_{i=1}^N V(1-\epsilon_i)c_{i,\max}\gamma_{i,\max}$, which completes the proof.

6) Let $(x_{i,q,t}, u_{c,i,t}, u_{d,i,t}, d_{f,i,t}, \pi_{i,t}, g_{i,t}, c_{i,t})$ and $(\hat{x}_{i,q,t}, \hat{u}_{c,i,t}, \hat{u}_{d,i,t}, \hat{d}_{f,i,t}, \hat{\pi}_{i,t}, \hat{g}_{i,t}, \hat{c}_{i,t})$ denote the optimal solution of **P5** and the proposed algorithm, respectively. According to the online adjustment in Algorithm 1, we have $\Gamma_{p,t} - \Gamma_{5,t} \le \sum_{i=1}^N \left(\sigma_i(\hat{u}^2_{c,i,t} + \hat{u}^2_{d,i,t}) + \delta_{1,i}\hat{c}^2_{i,t} + \delta_{2,i}\hat{c}_{i,t}\right) \le \Omega_2$, where $\Gamma_{5,t}$ and $\Gamma_{p,t}$ are the values of Γ_t corresponding to the solutions of **P5** and the proposed algorithm, respectively.

7) Let y_1 and y_2 denote the optimal solution of **P1** and **P2**, respectively. Since **P2** is a relaxation of **P1**, we have $y_2 \le y_1$. Since **P5** is a relaxation of **P4**, we have

$$\Delta_t + V\mathbb{E}\{\Gamma_{5,t}|\boldsymbol{\Theta_t}\} \tag{57}$$
$$\le \Delta_t + V\mathbb{E}\{\Gamma_{4,t}|\boldsymbol{\Theta_t}\} \tag{58}$$
$$\le \Omega_1 + \Omega_0 + V\mathbb{E}\{\tilde{\Gamma}_{3,t}|\boldsymbol{\Theta_t}\}$$
$$+ \mathbb{E}\{\sum_{i=1}^N \sum_{q=1}^{M_i} wQ_{i,q,t}(\pi_{i,q,t} - x^*_{i,q,t})|\boldsymbol{\Theta_t}\}$$
$$+ \mathbb{E}\{\sum_{i=1}^N \sum_{q=1}^{M_i} wH_{i,q,t}(\varepsilon_{i,q} - x^*_{i,q,t})|\boldsymbol{\Theta_t}\}$$
$$+ \mathbb{E}\{\sum_{i=1}^N Z_{i,t}(\eta_{c,i}u^*_{c,i,t} - \frac{1}{\eta_{d,i}}u^*_{d,i,t})|\boldsymbol{\Theta_t}\} \tag{59}$$
$$\le \Omega_1 + \Omega_0 + Vy_2 \tag{60}$$
$$\le \Omega_1 + \Omega_0 + Vy_1, \tag{61}$$

where $\Gamma_{4,t}$ and $\Gamma_{3,t}$ are the values of Γ_t corresponding to the solutions of **P4** and **P3**, respectively; $x^*_{i,q,t}, u^*_{c,i,t}, u^*_{d,i,t}$ are the elements in the solution vector of **P3**; (58) is derived by the part 5 of Theorem 1; (59) is obtained by incorporating the results of a stationary, randomized control strategy associated with **P2** [8]. By arranging the both sides of the above equations, we have $\mathbb{E}[\Delta_t] + V\mathbb{E}[\Gamma_{5,t}] \le \Omega_1 + \Omega_0 + Vy_1$. Continually, we have $V\sum_{t=0}^{T-1}\mathbb{E}\{\Gamma_{5,t}\} \le \Omega_1 T + YT + VTy_1 - \mathbb{E}\{L_T\} + \mathbb{E}\{L_0\}$. Dividing both side by VT, and taking a lim sup of both sides. Then, let $T \to \infty$, we have $\limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\{\Gamma_{5,t}\} \le y_1 + \frac{\Omega_0+\Omega_1}{V}$. By taking the part 6 of Theorem 1 into consideration, we have $\limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\{\Gamma_{p,t}\} \le y_1 + \Omega_2 + \frac{\Omega_0+\Omega_1}{V}$, which completes the proof. ∎

REFERENCES

[1] P. X. Gao, A. R. Curtis, B. Wong, and S. Keshav, "It's not easy being green," *Proc. of ACM SIGCOMM*, Helsinki, Finland, August 13-17, 2012.
[2] Amazon Addresses EC2 Power Outages, 2016 [Online]. Available: http://www.datacenterknowledge.com
[3] D. Salomonsson, L. Soder, and A. Sannino, "An adaptive control system for a dc microgrid for data centers," *IEEE Trans. Industry Applications*, vol. 44, no. 6, Nov./Dec. 2008.
[4] L. Yu, T. Jiang and Y. Cao, "Energy cost minimization for distributed internet data centers in smart microgrids considering power outages," *IEEE Trans. Parallel and Distributed Systems*, vol. 26, no. 1, pp. 120-130, Jan. 2015.
[5] L. Yu, T. Jiang, and Y. Zou, "Real-time energy management for cloud data centers in smart microgrids," *IEEE Access*, vol. 4, pp. 941-950, 2016.
[6] J. Li and W. Qi, "Towards optimal operation of internet data center microgrid," *IEEE Trans. Smart Grid*, DOI: 10.1109/TSG.2016.25722402, 2016.
[7] C.C. Thompson, P.E. Konstantinos Oikonomou, A.H. Etemadi, V.J. Sorger, "Optimization of data center battery storage investments for microgrid cost savings, emissions reduction, and reliability enhancement," *IEEE Trans. Industry Applications*, vol. 52, no. 3, pp. 2053-2060, MAY/JUNE 2016.
[8] M. J. Neely, *Stochastic network optimization with application to communication and queueing systems*. Morgan & Claypool, 2010.
[9] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1-122, 2011.
[10] X. Guan, Z. Xu, and Q. Jia, "Energy-efficient buildings facilitated by microgrid", *IEEE Trans. Smart Grid*, vol. 1, no. 3, pp. 243-252, Dec. 2010.
[11] M. Erol-Kantarci, B. Kantarci, and H.T. Mouftah, "Reliable overlay topology design for the smart microgrid network", *IEEE Network*, vol. 25, no. 5, pp. 38-43, Sep./Oct. 2011.
[12] Y. Huang, S. Mao, and R.M. Nelms, "Adaptive electricity scheduling in microgrids," *IEEE Trans. Smart Grid*, vol. 5, no. 1, pp. 27-281, 2014.
[13] Y. Zhang, N. Gatsis, and G. B. Giannakis, "Robust management of distributed energy resources for microgrids with renewables," *IEEE Trans. Sustainable Energy*, vol. 4, no. 4, pp. 944-953, Oct. 2013.
[14] K. Rahbar, J. Xu, and R. Zhang, "Real-time energy storage management for renewable integration in microgrid: an off-line optimization approach", *IEEE Trans. Smart Grid*, vol. 6, no. 1, pp. 124-134, Jan. 2015.
[15] W. Ma, J. Wang, V. Gupta, and C. Chen, "Distributed energy management for networked microgrids using online admm with regret," *IEEE Trans. Smart Grid*, DOI: 10.1109/TSG.2016.2569604, 2016.
[16] Y. Guo and C. Zhao, "Islanding-aware robust energy management for microgrids," *IEEE Trans. Smart Grid*, DOI: 10.1109/TSG.2016.2585092, 2016.
[17] W. Shi, N. Li, C.C. Chu, and R. Gadh, "Real-time energy management in microgrids," *IEEE Trans. Smart Grid*, DOI:10.1109/TSG.2015.2462294, 2016.
[18] T. Chen, X. Wang, G.B. Giannakis, "Cooling-aware energy and workload management in data centers via stochastic optimization," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 2, pp. 402-415, March 2016.
[19] L. Yu, T. Jiang, Y. Cao, and Q. Qi, "Carbon-aware energy cost minimization for distributed internet data centers in smart microgrids," *IEEE Internet of Things Journal*, vol 1, no. 3, pp. 255-264, June 2014.
[20] L. Yu, T. Jiang, Y. Cao, and Q. Qi, "Joint workload and battery scheduling with heterogeneous service delay guarantees for data center energy cost minimization," *IEEE Trans. Parallel and Distributed Systems*, vol. 26, no. 7, pp. 1937-1947, July 2015.
[21] R. Deng, G. Xiao, R. Lu, and J. Chen, "Fast distributed demand response with spatially- and temporally-coupled constraints in smart grid", *IEEE Trans. Industrial Informatics*, vol. 11, no. 6, pp. 1597-1606, 2015.
[22] S. Sun, M. Dong, B. Liang, "Distributed real-time power balancing in renewable-integrated power grids with storage and flexible loads," *IEEE Trans. Smart Grid*, vol. 7, no. 5, pp. 2337-2349, Sept. 2016.
[23] L. Yu, T. Jiang, and Y. Zou, "Price-sensitivity aware load balancing for geographically distributed internet data centers in smart grid environment", *IEEE Trans. Cloud Computing*, DOI: 10.1109/TCC.2016.2564406, 2016.
[24] L. Yu, T. Jiang, Y. Cao, and Q. Zhang, "Risk-constrained Operation For Distributed Internet Data Centers In Deregulated Electricity Markets,"

IEEE Trans. Parallel and Distributed Systems, vol. 25, no. 5, pp. 1306-1316, May 2014.

[25] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, and B. Maggs, "Cutting the electric bill for internet-scale systems," *Proc. of ACM SIGCOMM*, Barcelona, Spain, Aug. 17-21, 2009.

[26] H. Xu and B. Li, "Joint request mapping and response routing for geo-distributed cloud services," *Proc. of IEEE INFOCOM*, 2013.

[27] Z. Zhou, F. Liu, Z. Li and H. Jin, "When smart grid meets geo-distributed cloud an auction approach to datacenter demand response," *Proc. of IEEE INFOCOM*, 2015.

[28] J. Rivera, P. Wolfrum, S. Hirche, C. Goebel, and H. Jacobsen, "Alternating direction method of multipliers for decentralized electric vehicle charging control," *Proc. of IEEE CDC*, 2013.

[29] The fuel consumption of a diesel generator, 2016 [Online]. Available: http://generatorjoe.net/html/fueluse.asp

[30] D. P. Bertsekas, Dynamic programming and optimal control, second edition, Athena Scientific, 2000.

[31] B. He, M. Tao, and X. Yuan, "Alternating direction method with gaussian back substitution for separable convex programming," *SIAM Journal on Optimization*, vol. 22, no. 2, pp. 313-340, 2012.

[32] L. Yu, T. Jiang, and Y. Zou, "Distributed online energy management for data centers and electric vehicles in smart grid," *IEEE Internet of Things Journal*, DOI: 10.1109/JIOT.2016.2602846, 2016.

[33] L. Yu, T. Jiang, Y. Zou and Z. Sun, "Joint energy management strategy for geo-distributed data centers and electric vehicles in smart grid environment," *IEEE Trans. Smart Grid*, vol. 7, no. 5, pp. 2378-2392, Sept. 2016.