IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 15, 2020, accepted April 30, 2020, date of publication May 19, 2020, date of current version June 4, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2995806

Eco-Friendly Powering and Delay-Aware Task Scheduling in Geo-Distributed Edge-Cloud System: A Two-Timescale Framework

CHUNLEI SUN[1,2,3], XIANGMING WEN[1,2,3], (Senior Member, IEEE), ZHAOMING LU[1,2,3], (Member, IEEE), WENPENG JING[1,2,3], (Member, IEEE), AND MICHELE ZORZI[4], (Fellow, IEEE)

[1]Beijing University of Posts and Telecommunications, Beijing 100876, China
[2]Beijing Key Laboratory of Network System Architecture and Convergence (BUPT), Beijing 100876, China
[3]Beijing Laboratory of Advanced Information Networks (BUPT), Beijing 100876, China
[4]Information Engineering Department, University of Padova, 35131 Padova, Italy

Corresponding author: Zhaoming Lu (lzy_0372@163.com)

This work was supported in part by the Overseas Expertise Introductions Center for Discipline Innovation (“111 Center”) in China, State Scholarship Fund awarded by the China Scholarship Council under Grant 201806470032, in part by the State Grid Corporation of China through the project “The Evolution of Power Wireless Private Network and the Application Analysis of 4G and 5G Techniques” under Grant 5700-201955234A-0-0-00, in part by the National Natural Science Foundation of China under Grant 61801036, and in part by the National Science and Technology Major Project under Grant 2018ZX03001018.

ABSTRACT Power consumption and task latency are two crucial issues in edge-cloud computing. This paper mainly aims to promote the use of clean power in geo-distributed data centers (DCs) in a deregulated electricity market where customers are allowed to buy power from multiple suppliers, combined with the guarantee of task latency. To alleviate the conflict between frequent switches of servers and the uncertainty of task arrivals in DCs, this paper proposes a two-timescale framework consisting of the long-term capacity planning of geo-distributed DCs and the real-time task dispatching from edge gateways (EGs) to DCs. First, DCs make long-term plans on the number of active servers aiming at the eco-friendly and delay-aware power cost minimization, which is formulated as problem $\mathcal{P}$. Specifically, we introduce a convex pollution indicator function (PIF) to measure the pollution cost of the various types of powers sold by different suppliers, which can encourage the use of cleaner power and improve power savings. Second, in each sub-slot, each EG separately optimizes its individual mixed strategies of task dispatching to DCs with the knowledge of the planned capacities and the real-time queue backlogs of DCs, where a Lyapunov optimization framework is applied. Finally, we give the corresponding distributed algorithm design. Simulation results reveal that our method can realize the trade-off between the power cost and the delay cost of requests, and improve the clean power usage by up to 50%–60% of the total power usage in DCs. Additionally, comparisons with other schemes show that our approach can provide more stable guarantees of task latency in different situations of workload density, which benefits from the diverse-timescale optimizations of capacities of DCs and task routing from EGs to DCs.

INDEX TERMS Distributed task scheduling, edge-cloud system, two-timescale framework, pollution indicator function, eco-friendly and delay-aware, Lyapunov optimization.

I. INTRODUCTION

With the penetration of edge-cloud computing, the requests from users are explosively growing and the power consumptions in DCs are becoming enormous. On one hand, given the limitations of system's computation capacity and users' latency tolerance, it is of great significance to improve the efficiency of task scheduling inside the edge-cloud system in a situation of increasingly growing requests. On the other hand, the huge power consumptions of expanding DCs can not only increase the power cost of service providers, but also bring about massive carbon emissions along with severe air pollutions of sulfide and dust. According to [1], the total number of servers in the geo-distributed data centers of Google, Microsoft and Akamai were almost 1 million, 200,000 and 70,000 in 2010, respectively, and their corresponding power costs were on the order of millions of dollars per year. As reported in [2], the annual power consumption of DCs

The associate editor coordinating the review of this manuscript and approving it for publication was Ahmed F. Zobaa.

across China had reached up to 160 million MW.h, which was almost equivalent to the annual generation of the Three Gorges Dam. It is estimated that just a 1-MW data center powered by thermal power can cause over 10,000 metric tons of CO_2 emissions annually [3].

We consider a deregulated electricity market throughout this paper, which has been gradually popularized across the world with the development of energy Internet in recent years, e.g., in the south of China [4], the state of Texas in the US [5], several Nordic countries [6], etc. In a deregulated electricity market, local electricity companies (ECs) can buy wholesale electricity from different plants and then sell it to customers. At the same time, customers are allowed to freely choose and buy power from one or more ECs according to the price, cleanness and quality of the powers supplied by them. In this situation, geo-distributed DCs can take advantage of diversities of price and cleanness of electricity in different regions to optimize their power uses.

This paper mainly focuses on promoting the use of clean power in geo-distributed data centers (DCs) in a deregulated electricity market, combined with guarantees on task latency. A two-timescale framework is proposed, consisting of long-term capacity planning of geo-distributed DCs and real-time task dispatching from edge gateways (EGs) to DCs. To the best of the authors' knowledge, this is the first time that the number of active servers and the task scheduling strategies are optimized in diverse timescales, which can not only alleviate the conflict between frequent switches of servers and the uncertainty of task arrivals in DCs, but also provide more stable task latency guarantees in different situations of workload density, as shown by our simulation results.

A. RELATED WORK

First, we consider the literature about cost-aware task scheduling in edge-cloud systems. Reference [7] studied the minimization of power-cost via task scheduling among geo-distributed DCs by exploiting the spatial diversity of electricity price, where a boundary constraint of expected sojourn time in DCs was considered. Reference [8] investigated the temporal task scheduling with strict sojourn latency restrictions in a DC aiming to reduce the electricity bill by following the time-varying electricity price, where the considered DC should make decisions on task admissions and executions in each discrete time slot according to a multi-step ahead power cost. Reference [9] considered the spatial task scheduling among geo-distributed DCs on the basis of [8]. However, the transmission delay from sources to DCs was not considered in the literature mentioned above, which generally depended on the propagation distance, link bandwidth, traffic density, etc. Reference [10] modeled the transmission delay as an implicit function of the total task arrivals to a DC. The authors proposed an HBBF algorithm to search for the optimal solution by iteratively feeding back the value of transmission delay associated with the newly updated task scheduling strategies. Additionally, [11] investigated the offloading route selections and the task scheduling among fog nodes, and [12] discussed the joint optimization of link resources in a cloud radio access network (C-RAN), computation resources in mobile edge cloud (MEC) and task scheduling strategies between C-RAN and MEC, both of which applied Lyapunov optimization techniques to design online algorithms of power-delayed balanced online task scheduling. All of the above made great contributions on the cost-aware task scheduling in edge-cloud systems. However, they only considered electricity price in the power cost formulation, while clean power uses, power storage and multi-source power supplies were not mentioned. As stated in [13], the geo-distributed task scheduling in the case of static electricity price or the one decoupled from clean power generations was of nearly no social benefits. In addition, different from the majority of works on geographical task scheduling, [13] focused on the joint optimization of power cost and task latency, instead of the power cost minimization subject to task latency conditions.

Then, in the area of eco-friendly power usage, [14] proposed a bid mechanism for the colocation of data centers where equipment, space, and bandwidth were available for rental, and operators could realize carbon-aware task scheduling by using economic incentives to reshape the tenants' demand. Reference [15] investigated the balance of power demands and supplies with the aid of task scheduling among geographically green DCs powered by fuel cells. References [16]–[18] assumed that DCs could harvest power from their private green micro-grids. They proposed online algorithms to minimize the carbon emission and power cost by controlling the number of active servers, and the amount of power bought from power grids and collected from micro-grids. Reference [19] studied the joint scheduling of tasks and power storage over a time horizon to reduce the power cost of a DC, considering the temporal diversity of electricity. Reference [20] assumed that DCs could buy both clean power and brown power, and optimized the task scheduling among geo-distributed DCs besides the number of active servers and the purchase amounts of clean power and brown power, aiming to realize the trade-off among the request cost and carbon-inclusive power cost. However, it only simply modeled the cost of requesting servers as a linear function of the number of active servers, while ignoring the task latency experienced by users. Reference [21] studied the match between DCs and power suppliers in the background of the deregulated electricity market, which referred to the optimization of task arrivals to DCs from the perspective of demand response. All of the literature mentioned above only roughly considered the total expected task arrivals to DCs in the considered time slot, while ignoring the detailed process of task routing from different EGs to DCs, which on the other hand would be needed to characterize some crucial factors in practice, such as transmission delay and real time task arrivals to EGs.

To the best of the authors' knowledge, only a small number of papers in the previous literature jointly considered eco-friendly power usage and efficient task scheduling.

TABLE 1. Relative comparisons of the proposed scheme with the current state-of-the-art.

References	Task scheduling		Power consumption			Multi-time scale
	Transmission latency	Task routing	Power Storage	Sustainable DCs	Deregulated market	
[6][8]		✓				
[9][12][22]	✓	✓				
[14][16]		✓	✓	✓		
[15][17]			✓	✓		
[19]		✓	✓		✓	
[20]			✓		✓	
[21]		✓	✓			
[23][24]		✓				✓
Ours	✓	✓	✓	✓	✓	✓

Reference [22] focused on a carbon-aware power cost minimization problem with sojourn latency constraints of tasks in geo-distributed DCs. Reference [23] pointed out that large financial losses were inevitable once the failure of DCs occurred. To address this issue, it proposed a fault-tolerant task scheduling scheme from users to geo-distributed DCs, accounting for the trade-off between carbon-inclusive power costs and request costs. However, to avoid extreme complexities resulting from the integration of task scheduling and carbon-aware power usage, they ignored some crucial factors, such as the transmission delay, power storage, etc.

Finally, it is worth noting that all the above literature ignored the switch cost of servers in the sense that the operation timescale of switching servers on or off is the same with that of task scheduling. To address this issue, we concentrate our attention on a two-timescale framework design. Papers [24] and [25] were also interested in this. They used an extended Lyapunov optimization framework to investigate the trade-off between power costs and queue backlogs of DCs, which consisted of operations on two different timescales. Specifically, the number of active servers and the strategies for task routing were determined in large time slots, while the CPU frequency was optimized in each small time slot. However, neither transmission delay nor clean power usages were covered in [24] and [25]. In this paper, we determine the capacity provisioning of DCs in large time slots by jointly optimizing the task latency and power cost, as shown in problem $\mathcal{P}$. Then, we apply a Lyapunov optimization framework to optimize EG's task dispatching strategies in each small time slot in a distributed manner. Different from references [24] and [25], we optimize the number of active servers and the task scheduling strategies on diverse timescales, and only use the Lyapunov optimization framework in small time slots to conduct delay-aware task routing in a distributed manner. In our proposed scheme, the uses of clean power, power storage and transmission delay are all considered.

B. OUR CONTRIBUTIONS

Our main objective is to ameliorate the situation of power consumption and task scheduling in geo-distributed edge-cloud systems. To this aim, we propose a two-timescale scheme of eco-friendly powering and delay-aware task scheduling for edge-cloud systems in a deregulated electricity market. Specifically, (1) in large time slots, DCs optimize the number of active servers aiming to realize the trade-off between task latency and power costs, using a weighted sum of monetary cost and pollution cost. This power-latency trade-off is formulated as problem $\mathcal{P}$ and can be efficiently solved by our proposed Sequential Convex Programming (SCP) algorithm. We also introduce a convex pollution indicator function (PIF) to measure the pollution of the powers supplied by different ECs, which can encourage the uses of cleaner power and improve power savings. (2) In small time slots, called sub-slots, EGs separately optimize their strategies of task dispatching with the observation of capacities and queue backlogs of DCs. A Lyapunov optimization framework is applied for the design of a distributed task scheduling mechanism. To the best of the authors' knowledge, this is the first time the number of active servers and the task scheduling strategies are optimized on diverse timescales, which can efficiently alleviate the conflict between frequent switches of servers and the uncertainty of task arrivals in DCs. After that, we investigate the distributed algorithm design for capacity planning of DCs and task routing. It is worth noting that our scheme is a comprehensive approach, as shown by the illustrative comparisons of the proposed scheme with the current state-of-the-art summarized in Table 1.

Our main contributions can be summarized as follows.

- We propose a two-timescale framework for the capacity planning of geo-distributed DCs and task routing from edge-ends to cloud-ends. Benefiting from the diverse-timescale optimization of DC capacities and task routing, our scheme can provide stable and energy-efficient task latency guarantees.
- We formulate the long-term eco-friendly and delay-aware capacity planning problem of DCs in a deregulated electricity market as problem $\mathcal{P}$, and propose a low-complexity SCP algorithm to find the globally optimal non-integer solution of problem $\mathcal{P}$.
- We introduce a convex PIF to measure the pollution cost of powers supplied by different ECs, by which our proposed strategies can improve the usage of clean power by up to 50%–60% of the total bought power in DCs, as well as encourage power savings

and multiple-source power buying in a deregulated market.

The rest of this paper is organized as follows. First, we introduce the system architecture and give a formulation of the two-timescale framework in Section II. Second, we discuss the solution and the distributed algorithm design in Section III. Third, we give simulation results about the performance of our model in Section IV. Finally, we conclude the paper in Section V. For clarity, abbreviations and notations frequently used in this paper are listed in Table 2 and Table 3, respectively.

TABLE 2. Abbreviations.

Abbreviations	Full name
DC	Data center
EC	Electricity company
EG	Edge gateway
FCFS	First Come First Served
PIF	Pollution indicator function
PoA	Price of Anarchy
REG	Renewable energy generator
RMSE	Root-mean-square error
SCP	Sequential Convex Programming
SQP	Sequential Quadratic Programming
SP	Solar power
TP	Thermal power
WP	Wind power

II. MODELING AND FORMULATION

A. OVERVIEW OF THE SYSTEM MODEL

We consider an edge-cloud system consisting of I geographically separate DCs and J EGs, as shown in Fig. 1, where $J \gg I$. EGs are responsible for collecting requests from users and then dispatching tasks to local edge servers or distant DCs. Although the computation capacities of DCs are enormous, the remote distances between cloud-end and edge-end usually result in large transmission latency. Additionally, for the purpose of power conservation, we assume that only some of the servers in DCs are activated according to the instantaneous workload.

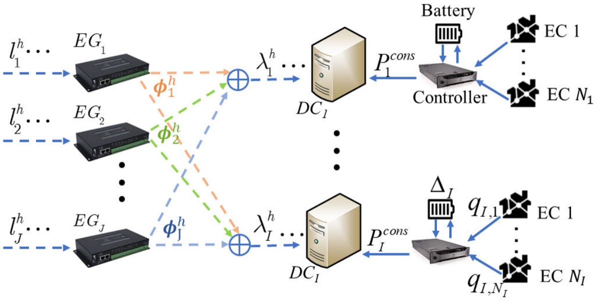


FIGURE 1. Architecture of geo-distributed edge-cloud system.

TABLE 3. Notations.

Notations	Definitions
DC_i	The i-th DC
EG_j	The j-th EG
$f_{i,n}^{PI}(\cdot)$	PIF of DC_i's n-th candidate EC
$L(\cdot)$	Lyapunov function
$\mathcal{L}(\cdot)$, $\mathcal{L}_i(\cdot)$	Lagrange function
$\mathcal{D}(\cdot)$	Lagrange dual function
$\Delta(\cdot)$	Lyapunov drift
i, I, $\mathcal{I}$	Index, number and set of DCs
j, J, $\mathcal{J}$	Index, number and set of EGs
n, N_i, $\mathcal{N}_i$	Index, number and set of candidate ECs for DC_i
h, H	Index and number of sub-slots in a large slot
$a_{i,n}$	Quadratic coefficient in PIF
$\bar{b}_j^h$	Average task length of EG_j in sub-slot h
c_i, C_i	Current state and capacity of batteries in DC_i
D_i	Sojourn time of tasks in DC_i
F_i^{PC}	Power cost of DC_i
l_s	Step length in Lagrange method
l_j^h	Number of task arrivals to EG_j in sub-slot h
L_σ	Lower bound of system task arrivals satisfying threshold probability σ
M_i, m_i^{act}	Number of servers and active servers of DC_i
P_i^{max}	Maximum power of DC_i
Q_i^h	Queue backlog of DC_i in sub-slot h
Q_i^{bat}	Amount of (dis)charged power of DC_i
Q_i^{buy}	Amount of bought power of DC_i
Q_i^{cons}	Amount of power consumption of DC_i
Q_i^{rew}	Amount of renewable power of DC_i
$p_{i,n}$, $q_{i,n}$	Price and amount of power bought by DC_i from its n-th candidate EC
s_i	Average CPU-cycle frequency of DC_i
T	Length of each large time slot
u_i^L, π	Lagrange multiplier
w_j	Lyapunov coefficient
W_{ji}^h	Effective bandwidth between EG_j and DC_i in sub-slot h
$\mathbf{q}_i$, $\mathbf{Q}^h$, ϕ_j^h	Vector of $q_{i,n}$, Q_i^h and ϕ_{ji}^h
α	Index for mapping CUP frequency to power consumption
β_i	Basic power consumption of DC_i
Δ_i	Energy variation of batteries in DC_i
Δ_i^{lb}, Δ_i^{ub}	Lower bound and upper bound of Δ_i
δ_i	C rate of batteries in DC_i
φ_i	Fraction of cooling power to total power in DC_i
ϕ_{ji}^h	Probability of EG_j dispatching tasks to DC_i in sub-slot h
Φ_i, Φ	Cost of DC_i and all DCs
$\gamma_{i,n}$	Pollution factor of power sold by DC_i's n-th candidate EC
η_i, η_i^i	(dis)charging efficiency of DC_i
λ_i^h	Task arrivals to DC_i in sub-slot h
λ_{ji}^h, λ_{-ji}^h	Task arrivals to DC_i from EG_j and EGs other than EG_j in sub-slot h
$\hat{\lambda}_i$, $\hat{\mu}_i$	Planned receiving and processing rate of tasks in DC_i during a large slot
Λ_{ji}^h	Communication latency between EG_j and DC_i in sub-slot h
$\theta_{1,i}$, $\theta_{2,i}$	Weighted parameter in cost function Φ_i
$\hat{\varepsilon}_i$	Estimated price of stored power in DC_i
τ_{ji}	Propagation time between EG_j and DC_i

We regard this edge-cloud system as a time-discrete system. To reduce the system power cost and task latency, we will develop a two-timescale framework, where DCs make decisions on the number of active servers at the beginning of large time slots, and then EGs can adjust their task dispatching strategies in each sub-slots according to the planned available capacities of DCs and the random request arrivals in real time. In this paper, we only concentrate on task routing from EGs to DCs rather than on the inner scheduling of various tasks in DCs (such as workflows [26], delay-sensitive tasks, etc.).

Let DC_i, $i \in \mathcal{I}$, represent the i-th DC, where $\mathcal{I} = \{1, \ldots, I\}$. All DCs are assumed to be equipped with smart energy controllers for the coordination of buying power from the market, charging or discharging batteries, and supplying power for loads. Specifically, we assume that DC_i can buy power from multiple ECs in a deregulated electricity market. Let $\mathcal{N}_i$ represent the set of candidate ECs for DC_i, and $q_{i,n}$ denote the amount of power bought from the n-th candidate EC by DC_i, where $q_{i,n} \geq 0$, $n \in \mathcal{N}_i$ and $|\mathcal{N}_i| = N_i$. Besides, we let $Q_i^{bat} > 0$ represent the amount of power consumed to charge batteries, $Q_i^{bat} < 0$ represent the power amount discharged from the batteries, and $Q_i^{bat} = 0$ represent that no power is charged or discharged. Then, the total power that DC_i needs to buy can be calculated by

$$Q_i^{buy} = \sum_{n=1}^{N_i} q_{i,n} = Q_i^{cons} + Q_i^{bat} - \mathbb{E}\{Q_i^{rew}\}, \tag{1}$$

where Q_i^{cons} is the power consumption of DC_i and $\mathbb{E}\{Q_i^{rew}\}$ is the expected amount of renewable power generated by the green micro-grid of DC_i. Given that the power consumption of DCs is very large in general, while the discharged power and generated renewable power are usually limited, we typically have $Q_i^{buy} \geq 0$.

Denote M_i and m_i^{act} as the number of total servers and active servers of DC_i in the considered time slot, respectively. According to [27], the power consumption of DC_i can be approximately calculated as

$$Q_i^{cons} = T \times (1+\varphi_i)(m_i^{act} s_i^{\alpha} + \beta_i), \quad m_i^{act} \in \{1, 2, \ldots, M_i\}, \tag{2}$$

where T is the length of one large slot, s_i and its index α are parameters related to the CPU frequency,[1] β_i represents the basic power consumption of servers, and $\varphi_i > 0$ is a constant to indicate the fraction of cooling power to servers' power in DC_i. Given that the price and unit pollution cost of electricity usually vary in different ECs and in different regions, we need to optimize the number of active servers in geo-distributed DCs aiming at power cost savings in the whole system.

On the other hand, let EG_j, $j \in \mathcal{J}$ represent the j-th EG, $\mathcal{J} = \{1, \ldots, J\}$. Without loss of generality, we assume that the request arrivals to different EGs are independent stochastic progresses. Dividing the considered time slot into H sub-slots, each of equal length T/H, let the random variable l_j^h represent the request rate (per sub-slot) to EG_j in sub-slot $h \in \{1, \cdots, H\}$. We assume that EG_j dispatches tasks to DC_i with probability ϕ_{ji}^h on sub-slot h, where $\sum_{i=1}^{I} \phi_{ji}^h = 1$, $j \in \{1, \cdots, J\}$ and $h \in \{1, \cdots, H\}$. Denoting $\boldsymbol{\phi}_j^h = (\phi_{ji}^1, \cdots, \phi_{ji}^H)$, we emphasize that EGs should make their optimal decisions on $\boldsymbol{\phi}_j^h$ in each sub-slot according to the real-time information of task arrivals to EGs and queue backlogs of DCs in order to minimize the related task latency.

B. LARGE-SLOT LAYER: CAPACITY PROVISIONING OF DCs

Although less active servers contribute to more power savings, this has the risk of increasing the task latency, so that decisions on the number of active servers should take into account the trade-off between power cost and task latency.

1) POWER COST

It is noted that the power cost considered in this paper includes both pollution cost and monetary cost.

First, we use a pollution indicator function (PIF) $f^{PI}(\cdot)$ to measure the pollution cost of the powers supplied by different ECs. In order to encourage eco-friendly power consumption, the PIF is assumed to meet the following conditions: (i) the more polluting the power, the larger the pollution cost, and (ii) the more the power bought, the larger the total and unit pollution cost. The simplest case of PIF is a quadratic function as

$$f_{i,n}^{PI}(q_{i,n}) = a_{i,n} q_{i,n}^2, \quad q_{i,n} \geq 0, \; a_{i,n} > 0, \; i \in \mathcal{I}, \; n \in \mathcal{N}_i, \tag{3}$$

where $a_{i,n} \overset{\triangle}{=} \frac{\gamma_{i,n}}{TP_i^{max}}$ is a positive coefficient, $\gamma_{i,n}$ is the pollution factor of the power supplied by the n-th EC in the region of DC_i, and P_i^{max} is the maximum power of DC_i. We assume that pollution factors for different types of powers are proportional to their pollutant emissions. Then, the pollution factor of an electricity company depends on the proportions of different power types in its power structure. It is noted that a more polluting power always leads to a larger $\gamma_{i,n}$. Introducing the divisor TP_i^{max} makes the pollution cost of DC_i penalized by the proportion of the purchased amount $q_{i,n}$ with respect to its maximum power consumption TP_i^{max}.

Second, we will address the monetary cost. Let $p_{i,n}$ and $\hat{\varepsilon}_i$ denote the electricity price published by the n-th EC and the estimated price of stored power in the region of DC_i, respectively. The reason for considering storage price is that the stored power and bought power are homogeneous commodities. Present decisions on charging or discharging will influence the future purchases of power [28], so that the storage price is a kind of potential cost. Let Δ_i denote the variation of the energy stored in the batteries during the (dis)charging procedure. When $\Delta_i \geq 0$, batteries charge. Otherwise, batteries discharge. Then, the monetary cost of DC_i can be given by

$$\sum_{n=1}^{N_i} p_{i,n} q_{i,n} - \hat{\varepsilon}_i \Delta_i. \tag{4}$$

When we consider the efficiency of charging and discharging, noted as $\eta_i \in (0, 1)$, the relationships between Δ_i and the (dis)charged power amount Q_i^{bat} can be expressed by

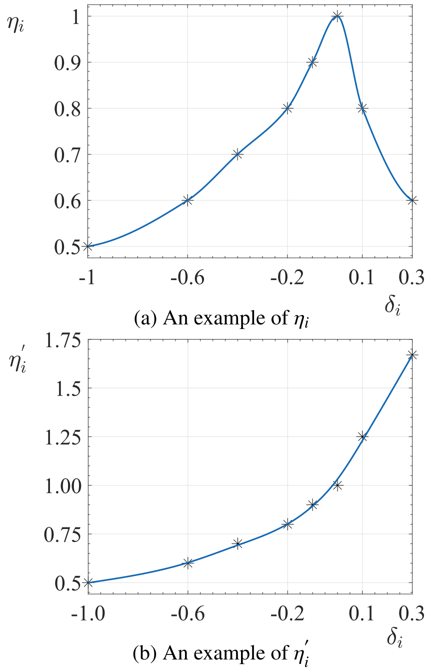
$$\begin{aligned} \Delta_i &= \eta_i Q_i^{bat}, \quad \Delta_i \geq 0 \\ Q_i^{bat} &= \eta_i \Delta_i, \quad \Delta_i < 0 \end{aligned} \quad . \tag{5}$$

[1] Although we ignore the differences among servers in the same DC, this will not affect the generality of our model, which can be readily extended.

If we make $\eta_i' = \eta_i$ when $\Delta_i \le 0$, while $\eta_i' = \frac{1}{\eta_i}$ when $\Delta_i > 0$, then (5) is equal to

$$Q_i^{bat} = \eta_i' \Delta_i \quad , \tag{6}$$

where $\eta_i' > 0$. Fig. 2a shows an example of η_i [29] and the corresponding η_i', where $\delta_i = \frac{\Delta_i}{TC_i}$ is the C rate of batteries' charging or discharging and C_i is the battery capacity of DC_i. We note that η_i' is a non-linear function of δ_i.[2]



(a) An example of η_i

(b) An example of η_i'

FIGURE 2. **Example of η_i and η_i'.**

Based on the above, the total power cost of DC_i can be formulated as the sum of pollution cost and monetary cost

$$F_i^{PC}(\mathbf{q}_i, \Delta_i) = \sum_{n=1}^{N_i} a_{i,n} q_{i,n}^2 + p_{i,n} q_{i,n} - \hat{\varepsilon}_i \Delta_i \,, \tag{7}$$

where $\mathbf{q}_i$ is the vector of $q_{i,n}$, $n = 1, \cdots, N_i$. The required constraints are

$$\sum_{n=1}^{N_i} q_{i,n} = Q_i^{cons} + \eta_i' \Delta_i - \mathbb{E}\{Q_i^{rew}\} \tag{8a}$$

$$-c_i \le \Delta_i \le C_i - c_i \tag{8b}$$

$$\Delta_i^{lb} \le \Delta_i \le \Delta_i^{ub} \tag{8c}$$

$$q_{i,n} \ge 0, \quad \forall n \in \mathcal{N}_i \,, \tag{8d}$$

where c_i is the current battery state of DC_i. (8a) is obtained from (1) and (6). (8b) represents that batteries can discharge no more than the stored power and can charge no more than the remaining capacity. In addition, due to the fact that an excessively high speed of charge and discharge will cause severe damage to storage devices as well as exorbitant wastes of energy [30], we set lower and upper bounds for Δ_i in (8c) as $\Delta_i^{lb} = -TC_i$ and $\Delta_i^{ub} = 0.3TC_i$, respectively.

2) TASK LATENCY

Let $\hat{\lambda}_i$ and $\hat{\mu}_i$ represent the planned receiving rate of tasks and the estimated service rate per server in DC_i in the considered large time slot. Denote $l = \frac{1}{H}\sum_{h=1}^{H}\sum_{j=1}^{J} l_j^h$ as the average task arrival rate of the whole system in the considered large time-slot, and let L_σ represent the quantile satisfying $\Pr(l < L_\sigma) = \sigma$, where σ is a predetermined probability threshold. As a latency guarantee, we consider the constraint of $\sum_{i=1}^{I} \hat{\lambda}_i \ge L_\sigma$.

We assume that the request arrival process to a DC can be modeled as a Poisson process, which is reasonable if the request arrivals are independent and stationary, and then we can use an M/M/n queuing model to estimate the queuing delay in DCs. Although some researches show that the dependency among request arrivals is likely to result in a deviation from the Poisson distribution, it is still useful in the area of queuing theory for its generality and tractability. In this part, our main purpose is to optimize the number of active servers, which can tolerate the estimation errors of queuing delay to some degree, so that we use the assumption of Poisson distribution here. In the sub-slot layer of task scheduling, a more precise approach will be implemented. According to the M/M/n queuing model [7], the task latency in DC_i is given by

$$D_i = \frac{T}{H} \times \left\{ \frac{1}{m_i^{act}\hat{u}_i - \hat{\lambda}_i} + \frac{1}{\hat{u}_i} \right\} \,, \tag{9}$$

where T/H is used for unit conversion.

3) JOINT OPTIMIZATION OF POWER COST AND TASK LATENCY

Incorporating the power cost in Eq. (7) and the latency cost in Eq. (9), the cost function of DC_i, $i \in \mathcal{I}$, can be formulated as

$$\Phi_i = \theta_{1,i} D_i + \theta_{2,i} F_i^{PC} \,, \tag{10}$$

where $\theta_{1,i} > 0$ and $\theta_{2,i} > 0$ are weight parameters. Then, the total cost function of the whole DC population can be given by $\Phi = \sum_{i=1}^{I} \Phi_i$. Besides, considering that Φ_i is a monotonically increasing function of $\hat{\lambda}_i$, $\forall i \in \mathcal{I}$, the inequality constraint $\sum_{i=1}^{I} \hat{\lambda}_i \ge L_\delta$ is equivalent to $\sum_{i=1}^{I} \hat{\lambda}_i = L_\delta$. Based on all of the above, an eco-friendly and delay-aware power cost minimization problem in the large-slot layer, denoted as problem $\mathcal{P}$, can be formulated as follows:

$$\min_{\{m_i^{act}, \hat{\lambda}_i, \Delta_i, \mathbf{q}_i\}, \forall i \in \mathcal{I}} \quad \Phi$$

$$s.t. \; \sum_{i=1}^{I} \hat{\lambda}_i = L_\sigma \tag{11a}$$

$$\sum_{n=1}^{N_i} q_{i,n} = Q_i^{cons} + \eta_i' \Delta_i - \mathbb{E}\{Q_i^{rew}\} \tag{11b}$$

$$m_i^{act}\hat{\mu}_i - \hat{\lambda}_i > 0 \tag{11c}$$

$$m_i^{act} \in \{1, 2, \ldots, M_i\} \tag{11d}$$

$$\hat{\lambda}_i \ge 0 \tag{11e}$$

[2] Throughout this paper, we will interchangeably use η_i', $\eta_i'(\delta_i)$ and $\eta_i'(\Delta_i)$ where Δ_i is a linearly scaled version of δ_i, specified by $\Delta_i = TC_i\delta_i$.

$$\max\{\Delta_i^{lb}, -c_i\} \le \Delta_i \le \min\{C_i - c_i, \Delta_i^{ub}\} \quad (11f)$$

$$q_{i,n} \ge 0, \quad n \in \mathcal{N}_i,\ i \in \mathcal{I}, \quad (11g)$$

where $Q_i^{cons} = T \times (1 + \varphi_i)(m_i^{act} s_i^{\alpha} + \beta_i)$.

C. SUB-SLOT LAYER: DISTRIBUTED TASK DISPATCHING

We focus on a mixed strategy of task dispatching in which EG_j dispatches tasks to DCs[3] with probability $\boldsymbol{\phi}_j^h$ in sub-slot h. A distributed mechanism of the mixed strategy generation based on the Lyapunov optimization framework is proposed in this part. Specifically, the task latency considered by EGs mainly includes the communication time from EGs to DCs and the sojourn time in DCs. We use the queue backlogs of DCs to measure the sojourn time in DCs and build the Lyapunov function associated with the queue backlogs. Then, we formulate the drift-plus-penalty function as the weighted sum of Lyapunov drift and communication latency. Each EG makes decisions on the mixed strategy of task dispatching separately by minimizing the upper bound of their own drift-plus-penalty function on each sub-slot.

1) COMMUNICATION LATENCY

The communication latency mainly results from the transmission latency and propagation time which can not be ignored due to far distances between EGs and DCs in general. Let W_{ji}^h (in bits/s) and τ_{ji} represent the effective bandwidth and the one-way propagation time between EG_j and DC_i, respectively. It is noted that W_{ji}^h changes over slots in the sense that the network congestion depends on the traffic state. Then, the communication latency between EG_j and DC_i in sub-slot h can be given by

$$\Lambda_{ji}^h = \frac{\bar{b}_j^h}{W_{ji}^h} + 2\tau_{ji}\,, \quad (12)$$

where $\bar{b}_j^h$ is the average length (in bits) of tasks from EG_j in sub-slot h. Considering that the data coming back from DCs is generally much less than that sent out by EGs, we ignore the transmission delay of back-stream from DCs to EGs.

2) SOJOURN TIME IN DCs

All tasks to a DC should be first put in a queue waiting for service with the principle of First Come First Served (FCFS) if no idle server is available immediately, where the queue backlog of a DC is proportional to the sojourn time in it. Let Q_i^h represent the queue backlog of DC_i in sub-slot h. Suppose the service process of DC_i in the considered time slot is stationary with average rate $\hat{\mu}_i$, which is independent of the task arrivals and the queue backlog of DC_i, then the queue evolution over time can be modeled as

$$Q_i^h = \max\{Q_i^{h-1} - m_i^{act}\hat{\mu}_i, 0\} + \lambda_i^h\,, \quad (13)$$

where λ_i^h is the total arrival rate of tasks (per sub-slot) to DC_i in sub-slot h. Denoting $\kappa_i^h = \min\{Q_i^{h-1}, m_i^{act}\hat{\mu}_i\}$, Eq.(13) can be rewritten by

$$Q_i^h = Q_i^{h-1} - \kappa_i^h + \lambda_i^h\,. \quad (14)$$

Our consideration of incorporating latency minimization with queue stability leads us to a design approach based on the Lyapunov optimization framework [31], which has been widely used in the stability control of dynamic systems. We define a Lyapunov function $L(\mathbf{Q}^h)$ as the sum of the squares of queue backlogs in DCs in sub-slot h

$$L(\mathbf{Q}^h) \overset{\triangle}{=} \frac{1}{2}\sum_{i=1}^{I}(Q_i^h)^2\,, \quad (15)$$

where $\mathbf{Q}^h$ is the vector of Q_i^h. $L(\mathbf{Q}^h)$ is a scalar measure of network congestion in the sense that it is small if all queues are small, and it is large if one or more queue is large. The difference in Lyapunov function from one sub-slot to the next and its upper bound can be given by

$$\begin{aligned} &L(\mathbf{Q}^h) - L(\mathbf{Q}^{h-1}) \\ &= \frac{1}{2}\sum_{i=1}^{I}\left[(Q_i^h)^2 - (Q_i^{h-1})^2\right] \\ &= \frac{1}{2}\sum_{i=1}^{I}\left\{\left[\max[Q_i^{h-1} - m_i^{act}\hat{\mu}_i, 0] + \lambda_i^h\right]^2 - [Q_i^{h-1}]^2\right\} \\ &\overset{(a)}{\le} \frac{1}{2}\sum_{i=1}^{I}\left\{(m_i^{act}\hat{\mu}_i)^2 + (\lambda_i^h)^2 + 2Q_i^{h-1}(\lambda_i^h - m_i^{act}\hat{\mu}_i)\right\}, \end{aligned} \quad (16)$$

where $\overset{(a)}{\le}$ is based on the fact that $[\max[a-b, 0] + c]^2 \le a^2 + b^2 + c^2 + 2a(c-b)$. Then, we define the Lyapunov drift $\Delta(\mathbf{Q}^{h-1})$ as the expected difference in Lyapunov function over one sub-slot given the current queue backlog $\mathbf{Q}^{h-1}$:

$$\Delta(\mathbf{Q}^{h-1}) \overset{\triangle}{=} \mathbb{E}\left\{L(\mathbf{Q}^h) - L(\mathbf{Q}^{h-1})|\mathbf{Q}^{h-1}\right\}. \quad (17)$$

With the hypothesis that the service rate $\hat{\mu}_i$ is independent of the queue backlog Q_i^{h-1}, the upper bound of $\Delta(\mathbf{Q}^{h-1})$ can be obtained by

$$\begin{aligned} \Delta(\mathbf{Q}^{h-1}) \le &\frac{1}{2}\mathbb{E}\left\{\sum_{i=1}^{I}(\lambda_i^h)^2|\mathbf{Q}^{h-1}\right\} \\ &+ \mathbb{E}\left\{\sum_{i=1}^{I}Q_i^{h-1}\lambda_i^h|\mathbf{Q}^{h-1}\right\} \\ &+ \frac{1}{2}\sum_{i=1}^{I}(m_i^{act}\hat{\mu}_i)^2 - \sum_{i=1}^{I}Q_i^{h-1}m_i^{act}\hat{\mu}_i. \end{aligned} \quad (18)$$

Taking into account the mixed strategy $\boldsymbol{\phi}_j^h$, tasks dispatched from EG_j to DC_i can be expressed by $\lambda_{ji}^h = l_j^h\phi_{ji}^h$. Let $\lambda_{-ji}^h = \sum_{m\neq j}\lambda_{mi}^h$ represent tasks dispatched to DC_i by EGs other than EG_j, and then there is $\lambda_i^h = \lambda_{-ji}^h + l_j^h\phi_{ji}^h$. From the perspective of EG_j, we record $\Delta(\mathbf{Q}^{h-1})$ as $\Delta_j(\boldsymbol{\phi}_j^h|\mathbf{Q}^{h-1})$, which depends on actions of other EGs. To decouple it in a tractable manner, we estimate λ_{-ji}^h by historical data, e.g., $\hat{\lambda}_{-ji}^h = \sum_{m=1}^{h-1} w_m(\lambda_i^m - \lambda_{ji}^m)$, where $w_m \ge 0$ is a weighted

[3] For clarity, the task dispatching to local edge servers is not incorporated, but our model can be easily extended.

parameter constrained by $\sum_{m=1}^{h-1} w_m = 1$. Besides, with the assumption that the task arrival rate l_j^h to EG_j is independent of Q^{h-1}, Eq.(18) can be rewritten as

$$\begin{aligned}
&\Delta_j(\boldsymbol{\phi}_j^h|\mathbf{Q}^{h-1})\\
&\le \frac{1}{2}\mathbb{E}\Big\{\sum_{i=1}^{I}(l_j^h\phi_{ji}^h)^2|\mathbf{Q}^{h-1}\Big\}\\
&\quad +\mathbb{E}\Big\{\sum_{i=1}^{I}(\hat{\lambda}_{-ji}^h+Q_i^{h-1})l_j^h\phi_{ji}^h|\mathbf{Q}^{h-1}\Big\}\\
&\quad +\sum_{i=1}^{I}\left\{\frac{(\hat{\lambda}_{-ji}^h)^2+(m_i^{act}\hat{\mu}_i)^2}{2}+Q_i^{h-1}(\hat{\lambda}_{-ji}^h-m_i^{act}\hat{\mu}_i)\right\}\\
&=\sum_{i=1}^{I}\left\{\frac{A_j^h}{2}(\phi_{ji}^h)^2+(\hat{\lambda}_{-ji}^h+Q_i^{h-1})B_j^h\phi_{ji}^h\right\}+U_j^h, \qquad (19)
\end{aligned}$$

where $A_j^h = \mathbb{E}\{(l_j^h)^2\}$, $B_j^h = \mathbb{E}\{l_j^h\}$ and $U_j^h = \sum_{i=1}^{I}\{[(\hat{\lambda}_{-ji}^h)^2+(m_i^{act}\hat{\mu}_i)^2]/2+Q_i^{h-1}(\hat{\lambda}_{-ji}^h-m_i^{act}\hat{\mu}_i)\}$.

3) JOINT OPTIMIZATION OF COMMUNICATION LATENCY AND SOJOURN TIME

We build a drift-plus-penalty function for EG_j as $\Delta_j(\boldsymbol{\phi}_j^h|\mathbf{Q}^{h-1}) + w_j\mathbb{E}\{\sum_{i=1}^{I}\Lambda_{ji}^h\}$ by incorporating the communication latency and Lyapunov drift, where $w_j > 0$ is a weighted parameter, $j \in \mathcal{J}$. The upper bound of $\Delta_j(\boldsymbol{\phi}_j^h|\mathbf{Q}^{h-1}) + w_j\mathbb{E}\{\sum_{i=1}^{I}\Lambda_{ji}^h\}$ is given by

$$\sum_{i=1}^{I}\left\{\frac{A_j^h(\phi_{ji}^h)^2}{2}+[w_j\sum_{i=1}^{I}\Lambda_{ji}^h+(\hat{\lambda}_{-ji}^h+Q_i^{h-1})B_j^h]\phi_{ji}^h\right\}+U_j^h. \qquad (20)$$

Following the design principle of Lyapunov optimization framework, we can set our objective to minimize the upper bound of the drift-plus-penalty term shown in (20) in each sub-slot. Ignoring the constant U_j^h, the task dispatching problem of EG_j, $j \in \mathcal{J}$ based on the Lyapunov optimization framework can be formulated as problem $\mathcal{Q}_j$:

$$\begin{aligned}
\min_{\boldsymbol{\phi}_j^h}\ &\sum_{i=1}^{I}\left\{\frac{A_j^h(\phi_{ji}^h)^2}{2}+[w_j\sum_{i=1}^{I}\Lambda_{ji}^h+(\hat{\lambda}_{-ji}^h+Q_i^{h-1})B_j^h]\phi_{ji}^h\right\}\\
s.\,t.\ &\sum_{i=1}^{I}\phi_{ji}^h=1\\
&\phi_{ji}^h\ge 0\,,\, i\in\mathcal{I}\,. \qquad (21)
\end{aligned}$$

Apparently, problem $\mathcal{Q}_j$ is a quadratic programming problem, which can be easily solved by some standard algorithms, such as the interior-point method or the sequential quadratic programming (SQP) method.

III. TWO-TIMESCALE FRAMEWORK OF CAPACITY PLANNING AND TASK DISPATCHING

Problem $\mathcal{P}$ in (11) is non-convex, due to the non-linear equality constraint (11b) and the integer restriction for m_i^{act} [32], while problem $\mathcal{Q}_j$ in (21) is a typical quadratic programming problem. Before giving the distributed algorithm for the DC capacity planning and task routing from EGs to DCs, we intend to study the solution of problem $\mathcal{P}$.

A. SOLUTIONS OF PROBLEM $\mathcal{P}$

In this part, we first apply continuity relaxation to problem $\mathcal{P}$ and propose an SCP algorithm to find its optimal non-integer solution. Without special statement, 'solution' mentioned in this subpart means non-integer solution. We find that the nonlinearity of constraint (11b) results from the non-linear function η_i' with respect to Δ_i, which is tightly associated with the part of power cost F_i^{PC} in Φ. Therefore, we intend to solve $F_i^{PC}(\cdot)$ and plug the solution into (11) to eliminate variables $q_{i,n}, i \in \mathcal{I}, n \in \mathcal{N}_i$. Then, the continuity-relaxed problem $\mathcal{P}$ can be transformed into a sequence of convex problems. This process is summarized in the proposed SCP algorithm. It is shown that the mixed-integer solution derived from the optimal continuous solution is quasi-optimal.

First, ignoring the inequality constraint $\mathbf{q}_i \succeq 0$, the partial Lagrange function of F_i^{PC} with respect to $\mathbf{q}_i$ constrained by (8a) can be established as

$$\begin{aligned}
\mathcal{L}_i(\mathbf{q}_i, u_i^L) = &\sum_{n=1}^{N_i}\{a_{i,n}q_{i,n}^2+p_{i,n}q_{i,n}-\hat{\varepsilon}_i\Delta_i\}\\
&-u_i^L\Big\{\sum_{n=1}^{N_i}q_{i,n}-Q_i^{cons}-Q_i^{bat}+\mathbb{E}\{Q_i^{rew}\}\Big\},
\end{aligned} \qquad (22)$$

where u_i^L is the Lagrange factor. According to the Lagrange Multiplier method [32], we set $\frac{\partial\mathcal{L}_i}{\partial q_{i,n}} = 0$ and $\frac{\partial\mathcal{L}_i}{\partial u_i^L} = 0$ for $\forall n \in \mathcal{N}_i$. Then, we can obtain the partial Lagrange dual solution of F_i^{PC} as

$$\begin{aligned}
q_{i,n}^{*L} &= \frac{v_i^{*L}-p_{i,n}}{2a_{i,n}}\\
&= \frac{2(Q_i^{cons}+Q_i^{bat}-\mathbb{E}\{Q_i^{rew}\})+Y_i-p_{i,n}X_i}{2a_{i,n}X_i}, \qquad (23)
\end{aligned}$$

The corresponding function value can be given by

$$\begin{aligned}
F_i^{PC}(\mathbf{q}_i^{*L}) = &\frac{(Q_i^{cons}+Q_i^{bat}-\mathbb{E}\{Q_i^{rew}\})^2+Y_i(Q_i^{cons}+Q_i^{bat})}{X_i}\\
&+\frac{Y_i^2-4Y_i\mathbb{E}\{Q_i^{rew}\}}{4X_i}-Z_i-\hat{\varepsilon}_i\Delta_i,
\end{aligned} \qquad (24)$$

where

$$X_i=\sum_{n=1}^{N_i}\frac{1}{a_{i,n}},\ Y_i=\sum_{n=1}^{N_i}\frac{p_{i,n}}{a_{i,n}},\ Z_i=\sum\nolimits_{n=1}^{N_i}\frac{(p_{i,n})^2}{a_{i,n}}. \qquad (25)$$

Then, the marginal power cost v_i^{*L} of DC_i when $q_{i,n} = q_{i,n}^{*L}$ for $\forall n \in \mathcal{N}_i$ is

$$v_i^{*L} = u_i^{L*} = \frac{2(Q_i^{cons}+Q_i^{bat}-\mathbb{E}\{Q_i^{rew}\})+Y_i}{X_i}, \qquad (26)$$

which is independent of n. Note that $X_i > 0$, $Y_i > 0$, $Z_i > 0$ and $v_i^{*L} > 0$ since $a_{i,n} > 0$, $p_{i,n} > 0$ and $Q_i^{cons} + Q_i^{bat} - \mathbb{E}\{Q_i^{rew}\} \geq 0$.

Second, substituting $F_i^{PC}(\mathbf{q}_i^{*L})$ into Φ_i, we can rewrite Φ_i as

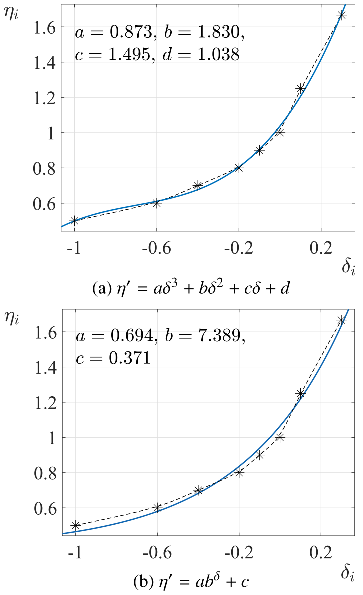
$$\begin{aligned}
&\Phi_i'(\hat{\lambda}_i, m_i^{act}, \Delta_i|\mathbf{q}_i^{*L}) \\
&= \theta_{1,i}\left\{\frac{1}{m_i^{act}\hat{\mu}_i - \hat{\lambda}_i} + \frac{1}{\hat{\mu}_i}\right\} \\
&\quad + \theta_{2,i}\left\{\frac{(Q_i^{cons} + \eta_i'\Delta_i - \mathbb{E}\{Q_i^{rew}\})^2}{X_i}\right. \\
&\quad + \frac{Y_i(Q_i^{cons} + \eta_i'\Delta_i - \mathbb{E}\{Q_i^{rew}\}) + 0.25Y_i^2 - X_iZ_i}{X_i} \\
&\quad \left. -\hat{\varepsilon}_i\Delta_i\right\}, \qquad (27)
\end{aligned}$$

where $\mathbf{q}_i^{*L}$ is given. Then, we formulate a new problem

$$\begin{aligned}
\min_{\{\hat{\lambda}_i, m_i^{act}, \Delta_i\}, \forall i \in \mathcal{I}} \quad & \sum_{i=1}^{I} \Phi_i'(\hat{\lambda}_i, m_i^{act}, \Delta_i|\mathbf{q}_i^{*L}) \\
s.t. \quad & \sum_{i=1}^{I} \hat{\lambda}_i = L_\sigma \\
& m_i^{act}\hat{\mu}_i > \hat{\lambda}_i \\
& m_i^{act} \in \{1, 2, \ldots, M_i\} \\
& \hat{\lambda}_i \geq 0 \\
& \max\{\Delta_i^{lb}, -c_i\} \leq \Delta_i \leq \min\{C_i - c_i, \Delta_i^{ub}\}, i \in \mathcal{I}, \qquad (28)
\end{aligned}$$

which is denoted as problem $\mathcal{P}1$. The feasible region of continuity-relaxed problem $\mathcal{P}1$ is a convex set on account of the elimination of the nonlinear equality constraint (11b). In addition, we prove in Appendix B that the continuity-relaxed objective function of $\mathcal{P}1$ is convex when the fitted function of η_i' meets the condition shown in Proposition 1. We observe from experiments that some cubic functions and exponential functions can both fit well the curve of η_i' shown in Fig. 2, under the convexity condition described in Proposition 1. Fig. 3 gives the optimal fitting curves of η_i' by cubic function and exponential function, respectively. In this sense, we can regard the continuity-relaxed problem $\mathcal{P}1$ as a convex programming problem [32].

Third, we reconsider the inequality constraint $q_{i,n} \geq 0$. It is easily inferred that if all $q_{i,n}^{*L}$ are non-negative, then they are the exactly optimal solutions to Φ given Δ_i, $i \in \mathcal{I}$, in which case problems $\mathcal{P}$ and $\mathcal{P}1$ are equivalent. For $\forall n \in \mathcal{N}_i$, $\forall i \in \mathcal{I}$, if $q_{i,n}^{*L} < 0$, then the optimal non-negative solution to the purchase amount of power $q_{i,n}$ is zero, which is proved in Appendix A. Under this principle, we propose the SCP algorithm to solve the global optimal solution of problem $\mathcal{P}$. As is shown in Algorithm 1, lines 6 to 13 are used to revise the values of $q_{i,n}$ by updating the set $\mathcal{N}_i^-$ and making all $q_{i,n} = 0$ for $n \in \mathcal{N}_i^-$, where $\mathcal{N}_i^- = \{q_{i,n} \mid q_{i,n}^{*L} < 0,\ n \in \mathcal{N}_i\}$ and $\mathcal{N}_i = \mathcal{N}_i^- \cup \widetilde{\mathcal{N}}_i^-$. Line 3 is used to update the value of F_i^{PC} according to the revised $\mathbf{q_i}$. Every time we solve a new version of $\mathbf{q_i}$, we plug the corresponding F_i^{PC} into (28) and obtain a new version of problem $\mathcal{P}1$, which can be solved by standard convex programming methods, such as the SQP method shown in Line 4. In Appendix A, we give mathematical proofs that our proposed SCP algorithm can always find the globally optimal solution of problem $\mathcal{P}$. Besides, the SCP algorithm converges fast, as it can find the optimal solution of problem $\mathcal{P}$ by solving problem $\mathcal{P}1$ no more than $2 \times \sum_{i=1}^{I} N_i$ times.



(a) $\eta' = a\delta^3 + b\delta^2 + c\delta + d$

(b) $\eta' = ab^\delta + c$

FIGURE 3. Optimal fitting curves of η' by cubic function and exponential function.

Finally, we consider the integer restrictions of m_i^{act} by rounding all m_i^{act} to the nearest integers. Simulation results show that the gaps between the function values of Φ with respect to the optimal non-integer solution and this derived mixed-integer solution are on the order of 0.01%, which is negligible.

B. DISTRIBUTED SOLUTIONS OF PROBLEM $\mathcal{P}$

Although problem $\mathcal{Q}_j$ can be solved by each EG separately, problem $\mathcal{P}$ is associated with the whole DC population, so that we should first study the decomposition of problem $\mathcal{P}$. We find that the coupling among DCs in problem $\mathcal{P}$ results from the linear equality constraint (11a), which can be easily decoupled by the method of partial Lagrange dual [32]. First, denoting π as the Lagrange multiplier associated with constraint (11a), the partial Lagrange function of problem $\mathcal{P}$ is given by

$$\begin{aligned}
\mathcal{L}(\pi, \{m_i^{act}, \hat{\lambda}_i, \Delta_i, \mathbf{q}_i\}_{i\in\mathcal{I}}) &= \sum_{i=1}^{I} \Phi_i - \pi(\sum_{i=1}^{I} \hat{\lambda}_i - L_\sigma) \\
&= \sum_{i=1}^{I} \left\{\Phi_i - \pi(\hat{\lambda}_i - L_\sigma)\right\} \qquad (29)
\end{aligned}$$

Algorithm 1 Sequential Convex Programming Algorithm

```
Initialize DC set $\mathcal{I}$ and other relevant parameters. Make $\mathcal{N}_i^- = \emptyset$ and $\widetilde{\mathcal{N}}_i^- = \mathcal{N}_i, \forall i \in \mathcal{I}$.
Repeat
   Calculate $X_i$, $Y_i$ and $Z_i$ on subset $\widetilde{\mathcal{N}}_i^-$ according to (25) for $\forall i \in \mathcal{I}$, and plug them into problem $\mathcal{P}1$.
   Solve problem $\mathcal{P}1$ with a convex programming method, such as SQP.
   Calculate $q_{i,n}^{*L}$ and $\nu_i^{*L}$ as in (23) and (25), respectively, on subset $\widetilde{\mathcal{N}}_i^-, \forall i \in \mathcal{I}$.
   For $i = 1 : |\mathcal{I}|$
      If there is any $q_{i,n}^{*L} < 0, n \in \widetilde{\mathcal{N}}_i^-$ then
         Move all indices $n$ of $q_{i,n}^{*L} < 0$ from $\widetilde{\mathcal{N}}_i^-$ to $\mathcal{N}_i^-$
      End If
      If there is any $p_{i,n} < \nu_i^{*L}, n \in \mathcal{N}_i^-$ then
         Move all indices $n$ of $p_{i,n} < \nu_i^{*L}$ back to $\widetilde{\mathcal{N}}_i^-$
      End If
   End For
Until No $n$ is newly moved into $\mathcal{N}_i^-$ or $\widetilde{\mathcal{N}}_i^-$, $\forall i \in \mathcal{I}$
Output the renewed $\hat{\lambda}_i$, $m_i^{act}$, $\Delta_i$, $q_{i,n\in\mathcal{N}_i^-}^* = 0$ and $q_{i,n\in\widetilde{\mathcal{N}}_i^-}^* = q_{i,n}^{L*}, \forall i \in \mathcal{I}$.
```

Then, the corresponding partial Lagrange dual function is

$$\mathcal{D}(\pi) = \inf_{\{m_i^{act}, \hat{\lambda}_i, \Delta_i, \mathbf{q}_i\}_{i\in\mathcal{I}}} \mathcal{L} = \sum_{i=1}^{I} D_i(\pi) + \pi L_\sigma , \tag{30}$$

where the subproblem $D_i(\pi)$ represents

$$\inf_{m_i^{act}, \hat{\lambda}_i, \Delta_i, \mathbf{q}_i} \left\{ \Phi_i(m_i^{act}, \hat{\lambda}_i, \Delta_i, \mathbf{q}_i) - \pi \hat{\lambda}_i \right\} \tag{31}$$

It is noted that $D_i(\pi)$ is only related to the individual decision variables of DC_i, so that subproblem (31) can be separately solved by $DC_i, i \in \mathcal{I}$ in a parallel manner. Furthermore, the partial dual problem of problem $\mathcal{P}$ can be formulated by

$$\begin{aligned} &\max_{\pi} \mathcal{D}(\pi) \\ &s.t. \quad (11b) - (11g). \end{aligned} \tag{32}$$

From the perspective of the system operator, we can update the value of π in an iterative manner according to the method of gradient descent:

$$\begin{aligned} \pi(t+1) &= \pi(t) + l_s \left(\frac{\partial \mathcal{D}(\pi(t))}{\partial \pi(t)} \right) \\ &= \pi(t) + l_s \left(\sum_{i=1}^{I} \hat{\lambda}_i - L_\sigma \right), \end{aligned} \tag{33}$$

where $t = 1, 2, \cdots$ is the count of iterations, $l_s > 0$ is the step length and $\pi(t)$ represents the updated value of π in the $t-$th iteration.

Next, to make our proposed SCP algorithm valid for solving $D_i(\pi)$ shown in (31), the subproblem of the dual problem (32), we conduct the same operations described in subsection 3.1 on $D_i(\pi)$ to eliminate q_i, so that

$$D_i(\pi | q_i^{*L}) = \inf_{m_i^{act}, \hat{\lambda}_i, \Delta_i} \left\{ \Phi_i'(m_i^{act}, \hat{\lambda}_i, \Delta_i | q_i^{*L}) - \pi \hat{\lambda}_i \right\}, i \in \mathcal{I} \tag{34}$$

subject to the boundary restrictions (11d)–(11f). $\Phi_i' - \pi \hat{\lambda}_i$ is convex given π, since the summation of a convex function and a linear function is still convex [32]. Then, for a given π, we can solve $D_i(\pi)$ by our proposed SCP algorithm as long as the set $\mathcal{I}$ in the SCP algorithm is initialized as $\{i\}$. In each iteration round of the SCP algorithm, an updated version of problem (34) will be solved rather than problem $\mathcal{P}1$. We emphasize that although problem $\mathcal{P}1$ is convex, the original problem $\mathcal{P}$ is non-convex, so that the optimal solution to the dual problem (32) can only provide a lower bound for the original problem $\mathcal{P}$. The gap between the optimal function values of the dual problem (32) and the original problem $\mathcal{P}$ is regarded as the cost of distributed algorithm design.

C. TWO-TIMESCALE FRAMEWORK DESIGN

Fig. 4 shows the two-timescale framework of the distributed capacity planning and task dispatching algorithm. The procedure is as follows.

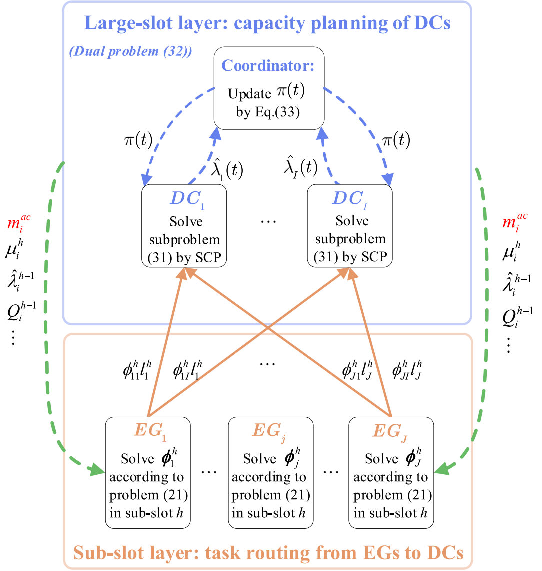

Large-slot layer: capacity planning of DCs
(Dual problem (32))
Coordinator:
Update $\pi(t)$ by Eq.(33)
$\pi(t)$
$\pi(t)$
$\hat{\lambda}_1(t)$
$\hat{\lambda}_I(t)$
DC_1
Solve subproblem (31) by SCP
…
DC_I
Solve subproblem (31) by SCP
m_i^{ac}
μ_i^h
$\hat{\lambda}_i^{h-1}$
Q_i^{h-1}
⋮
m_i^{ac}
μ_i^h
$\hat{\lambda}_i^{h-1}$
Q_i^{h-1}
⋮
$\phi_{11}^h l_1^h$
$\phi_{1I}^h l_1^h$
…
$\phi_{J1}^h l_J^h$
$\phi_{JI}^h l_J^h$
EG_1
Solve $\boldsymbol{\phi}_1^h$ according to problem (21) in sub-slot h
…
EG_j
Solve $\boldsymbol{\phi}_j^h$ according to problem (21) in sub-slot h
…
EG_J
Solve $\boldsymbol{\phi}_J^h$ according to problem (21) in sub-slot h
Sub-slot layer: task routing from EGs to DCs

FIGURE 4. **Two-timescale framework of the distributed capacity planning and task dispatching algorithm in an edge-cloud system.**

(i) Large-slot layer. At the beginning of the considered large slot, the cloud layer should optimize the strategies of power purchasing and make decisions on the number of active servers by solving the partial dual problem (32) of problem $\mathcal{P}$. The distributed solution procedure of problem (32) is iterative as follows.

- Initialize the iteration count $t = 1$, the operating factor $\pi(1) = 1$, etc. Then, repeat the following three steps:
  - $DC_i, \forall i \in \mathcal{I}$ receives the updated value of operating factor $\pi(t)$, and then solves its individual

problem (31) by the SCP algorithm. After that, each DC sends its updated solution $\hat{\lambda}_i$ to the coordinator.

- Based on the feedback $\hat{\lambda}_i$ from each DC, the coordinator updates $\pi(t+1)$ according to (33).
- The coordinator checks the termination condition. If it holds, then this iteration procedure will be finished. Otherwise, the coordinator will broadcast the updated operating factor $\pi(t+1)$ to all DCs and make $t = t+1$.

- Output the power purchasing strategy and planned number of active servers for each DC.

It is noted that this iterative procedure is run by the coordinator. The termination condition referred to in the repeated round is that either the maximum iteration count has been reached or the descent distance of operation factor π is lower than a predetermined threshold.

For clarity, we give some necessary illustrations of the above procedure. Problem (32) is the partial dual problem of the original problem $\mathcal{P}$, and can be solved in a distributed manner by iteratively solving its two sub-problems (31) and (33). When we solve sub-problem (31) by our proposed SCP algorithm, an updated version of problem (34) will be involved in each iteration round of the SCP algorithm. Actually, the original problem $\mathcal{P}$ could be directly solved in a centralized manner, where a sequence of problems $\mathcal{P}1$ will be involved in the SCP algorithm rather than problem (34). However, the centralized mechanism requires large amounts of individual information of the DCs to be published, such as the storage capacity, computation capacity, etc., which violates the principle of privacy protection, especially when the involved DCs are affiliated with different service providers.

(ii) Sub-slot layer. At first, $EG_j, j \in \mathcal{J}$ obtains the number of active servers m_i^{act} of $DC_i, i \in \mathcal{I}$. Then, in each sub-slot, the EG population conducts the following two steps in a parallel manner.

- $EG_j, j \in \mathcal{J}$ observes the processing speed μ_i^h and queue backlog Q_i^{h-1} of DC_i, and estimates the task arrivals $\hat{\lambda}_{-ji}^h$ to DC_i according to historical data, $\forall i \in \mathcal{I}$.
- Accordingly, $EG_j, j \in \mathcal{J}$ optimizes its dispatching strategy $\boldsymbol{\phi}_j^h$ by solving problem $\mathcal{Q}_j$, and then dispatches tasks according to the optimized $\boldsymbol{\phi}_j^h$.

We assume that DCs can dynamically control CPU frequencies of servers according to real-time queue backlogs of themselves, so that EGs should update the knowledge of the processing speed μ_i^h of DC_i in each sub-slot, $\forall i \in \mathcal{I}$. The cooperation of the dynamical CPU-frequency mechanism and the Lyapunov optimization technique can further improve the stability of queues in DCs, which brings significant benefits in terms of guarantees for the task latency. According to reference [18], the optimal strategy of a DC in the situation of dynamical CPU frequency is that all active servers have identical CPU frequencies. Due to space limitations, we will not elaborate on the dynamical CPU-frequency mechanism of DCs, which is an issue independent of our main concerns.

So far, we have finished the discussion on two-timescale framework design. It is observed that only little information needs to be exchanged among agents in the above distributed mechanism, including $\pi(t)$, $\hat{\lambda}_i(t)$, m_i^{act}, μ_i^h, λ_i^{h-1} and Q_i^{h-1}, which is beneficial for the protection of privacy and autonomy of agents. Besides, it can reduce communication costs in the control procedure and difficulties in the implementation. Although iterative updates of the operation factor π in the distributed scheme always consume extra communication costs compared to the centralized scheme, this is acceptable when the length of the large slot is large enough (e.g. one hour).

Finally, we briefly analyze the complexity of our proposed scheme. We roughly classify the complexity into three levels according to the adopted optimization technique and the problem scale, i.e., low, medium and high complexity. Table 4 gives a complexity analysis of some related state-of-the-art approaches. Specifically, the Lyapunov optimization technique is typically used in real-time system control, and the related schemes are of low complexity only if their real-time control problems are simple. The sub-slot scheme proposed in our work and those in [23][24] only refer to small-scale linear or quadratic programming problems, so that all of them can be regarded as low-complexity approaches. Additionally, in the large-slot layer, the properties and solution approach (SCP algorithm) of problem $\mathcal{P}$ are similar to those in reference [17], while the problem scale is smaller. Thus, the large-slot scheme proposed in our work has lower complexity than that in [17]. Moreover, the problem scale of our proposed sub-slot scheme and large-slot scheme depends linearly on $I \times J$ and I, respectively, where I is the number of DCs and J is the number of EGs.

TABLE 4. Complexity analysis of some related state-of-the-art approaches.

References	Techniques	Scale	Complexity
[6]	Mixed-integer nonlinear programming (MINLP)	Small	Medium
[8]	Linear programming & lumped queue	Medium	Medium
[9]	MINLP & heuristic branch and bound with feedback	Small	High
[14]	Lyapunov optimization framework	Medium	Medium
[17]	Heuristic based convex programming	Medium	Medium
[19]	Mixed-integer linear programming (MILP)	Small	Low
[21]	MINLP &Lyapunov & Benders decomposition	Small	High
[23][24]	Extended Lyapunov optimization framework	Small	Low

IV. NUMERICAL ANALYSIS

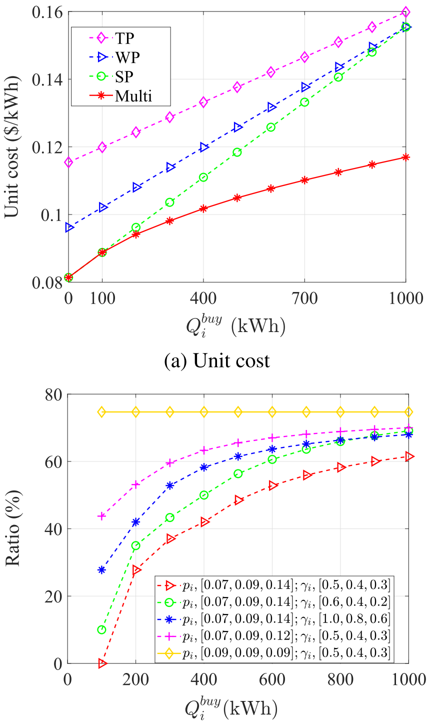
To verify our model, we use Matlab to solve problem $\mathcal{P}$ by our proposed SCP algorithm and give the performance analysis focusing on four aspects, including (i) the impact of PIF, (ii) the behavior of storage scheduling, (iii) the analysis of task dispatching, and (iv) the reduction of power cost and the improvement of clean power usage. Table 5 gives default settings of some parameters. Unless specified otherwise, we configure the parameters of all of our experiments according to Table 5. For example, the benchmarks of the electricity price used in the following experiments correspond to the default electricity price shown in Table 5. Specifically, we assume that each DC can buy power from $N_i = 3$ ECs, whose power for sale mainly originates from thermal power (TP), solar power (SP) and wind power (WP), respectively. The corresponding electricity prices[4] always change at different hours and in different areas.

TABLE 5. Default parameter Settings.

Parameters	Values
Number of DCs I	4
Number of EGs J	50
Number of servers M_i	200
Number of ECs for each DC N_i	3
Capacity of storage, C_i	$40\% P_i^{max} - 60\% P_i^{max}$
CPU-cycle frequency, s_i	0.8–2.8 GHz
Index of CPU frequency, α	3
Basic power, β_i	40 kW – 60 kW
Fraction of cooling to servers' power, φ_i	0.8
Processing speed per server, $\hat{\mu}_i$	20 requests per second (at $s_i = 2$ GHz)
Pollution factor of TP, $\gamma_{i,1}$	0.5
Pollution factor of SP, $\gamma_{i,2}$	0.4
Pollution factor of WP, $\gamma_{i,3}$	0.3
Electricity price of TP, $p_{i,1}$	0.04 – 0.12 \$/kWh
Electricity price of SP, $p_{i,2}$	0.10 – 0.18 \$/kWh
Electricity price of WP, $p_{i,3}$	0.08 – 0.14 \$/kWh
Length of large slot	1 hour
Length of sub-slots	1–10 minutes

A. EFFECT OF THE POLLUTION INDICATOR FUNCTION

First, we aim to verify the effect of PIF on multi-source power buying. Fig. 5a shows the unit power costs for a 1-MW data center when buying power from TP, SP, WP alone and from multiple ECs in one time slot, respectively. It is observed that the unit cost increases with the growth of Q_i^{buy}. This means that the more the power we buy, the higher the unit cost we will pay, which can lead to an increasingly fast growth of total power costs and encourage power savings in order to reduce costs. In addition, we can see that the unit costs in the cases of buying power from TP, SP or WP alone are always higher than those when buying multi-source power, which is caused by the quadratic term in PIF. Therefore, our proposed PIF can encourage users to buy power from multiple suppliers rather than single suppliers.



(a) Unit cost

(b) Ratio of clean power

FIGURE 5. Effects of PIF on the unit cost of power and the usage ratio of clean power.

Fig. 5b shows the ratio of clean power, including SP and WP, to the total bought power Q_i^{buy}. It can be observed that the fraction of clean power will increase with the growth of Q_i^{buy} when the electricity prices $p_{i,n}, \forall n \in \mathcal{N}_i$ are different, which means that the more the power we buy, the higher the fraction of clean power we will use, because the unit cost of clean power increases more slowly than that of brown power due to the smaller pollution factor $\gamma_{i,n}$, although the price $p_{i,n}$ of clean power is higher in general. However, when $p_{i,n}$, $\forall n \in \mathcal{N}_i$ are identical, the ratio of clean power is a constant and is independent of Q_i^{buy}. In this case, the numerators of $q_{i,n}^{*L}$ $\forall n \in \mathcal{N}_i$ in Eq. (23) are equal, thus the ratio $q_{i,1}^{*L} : \cdots : q_{i,N_i}^{*L}$ can be calculated as $\frac{1}{a_{i,1}} : \cdots : \frac{1}{a_{i,N_i}}$, which is constant. Furthermore, we can see that the fraction of clean power is sensitive to the configuration of the electricity price and the pollution factor $\gamma_{i,n}$. The fraction will be higher when the differences among $\gamma_{i,n}$ increase, or the differences among $p_{i,n}$ decrease, $\forall n \in \mathcal{N}_i$.

We can conclude from Fig. 5 that our proposed PIF can encourage power savings, power buying from multiple sources and usage of clean power. The more the power we buy, the higher the usage ratio of clean power. If we set the unit pollution cost to a constant as in the previous works, then the total pollution cost will be a linear function rather than a non-linear function as PIF. In the linear case, users will always choose the EC with lowest unit cost, the sum of electricity price and unit pollution cost, and abandon the powers from other ECs that are a little more expensive. Different from the linear case, the PIF can motivate users to buy power from multiple ECs and the ratio of clean power will increase with

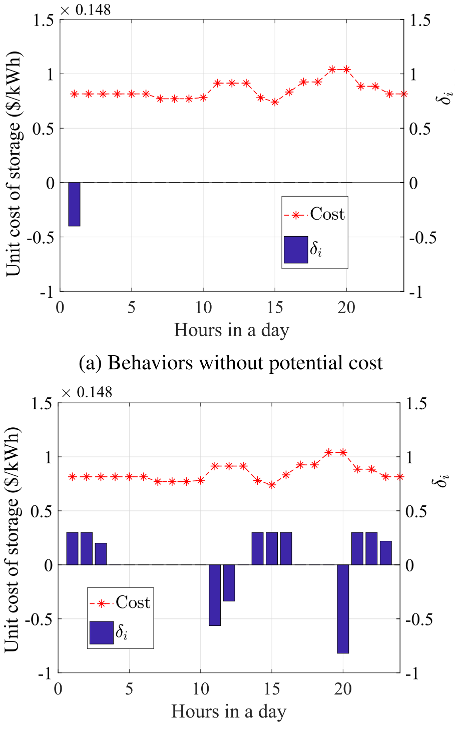
[4] All electricity prices used in this paper are set according to those in China [33], [34], and converted from ¥ to \$ at an exchange rate of 0.148.

the growth of the total bought power. In addition, the fraction of clean power is sensitive to several parameters, such as electricity price and pollution factor $\gamma_{i,n}$.

B. ANALYSIS OF THE STORAGE SCHEDULING

We first fit the curve of η_i' shown in Fig. 2b with many different power and exponential functions. We find that the cubic function $0.873\delta^3 + 1.830\delta^2 + 1.495\delta + 1.038$ where $\delta \in [-1, 0.3]$ is the simplest one that meets the condition in Proposition 1 to guarantee the convexity of problem $\mathcal{P}1$ with negligible fitting errors, so that we adopt the above cubic function in this paper.

Second, we aim to verify the impact of potential cost on storage scheduling and power cost. Fig. 6a shows the optimal decisions on charge and discharge without considering potential cost, in which case batteries discharge much in the first time slot and do not charge or discharge in the remaining slots in order to obtain the maximum present interests while ignoring the future interests. Fig. 6b shows the optimal decisions on charge and discharge considering potential cost, where batteries always tend to discharge in slots with higher unit cost and charge in slots with lower unit cost, where h is the index of a future slot. This reveals that the long-term storage scheduling according to the varying unit-cost of storage can only be optimized when the potential cost is taken into consideration. Additionally, we note that the potential cost can actually influence the storage scheduling in a duration longer than hours. It is observed from our many experiments that the storage scheduling controlled by potential cost generally exhibits a periodicity over days, following the corresponding electricity price. To clearly show how the fluctuations of the electricity price influence (dis)charging actions, we just show the storage scheduling during a day.

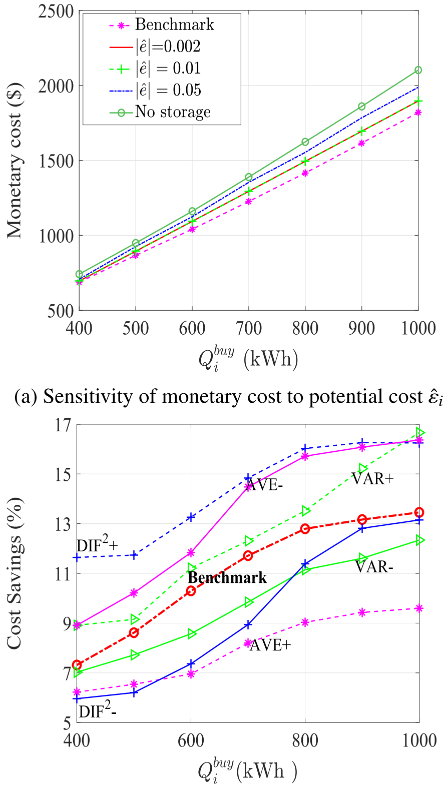


(a) Behaviors without potential cost

(b) Behaviors with potential cost

FIGURE 6. Impact of potential cost on storage scheduling and power cost.

Fig. 7a shows the daily monetary costs for a 1-MW DC with different accuracy of $\hat{\varepsilon}_i$ and that without $\hat{\varepsilon}_i$, where $|\hat{e}|$ represents the prediction errors for $\hat{\varepsilon}_i$ measured as the differences between the actual values and the predicted values. The referred benchmark corresponds to the realization of $\hat{\varepsilon}_i$ where $|\hat{e}| = 0$. It is observed that introducing potential cost can improve the power savings, but a small prediction error $\hat{e}$ for $\hat{\varepsilon}_i$ can cause a severe performance decline of storage scheduling. This means that the performance of storage scheduling is very sensitive to the parameter $\hat{\varepsilon}_i$, and an accurate prediction method of the unit cost of storage in the future slots is needed, which will be further studied in our future work.



(a) Sensitivity of monetary cost to potential cost $\hat{\varepsilon}_i$

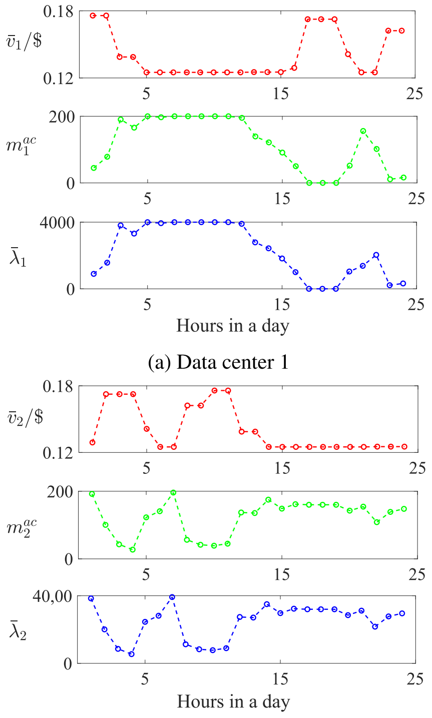
(b) Sensitivity of cost savings to electricity price

FIGURE 7. Sensitivity analysis of parameters for storage scheduling.

Third, we aim to illustrate how the electricity price influences the power savings of storage scheduling, as shown in Fig. 7b. We adopt the default electricity price as a benchmark, and change the average values 'AVE' and the variance 'VAR' of the electricity price, and the square sum of differences between prices in adjacent time slots 'DIF^2', respectively, where '+' represents increase and '−' represents decrease. It is found that the ratios of monetary costs saved by storage scheduling to the total monetary costs will increase when 'VAR' or 'DIF^2' grows. This means that the higher the differences among electricity prices in different slots, the more the monetary costs saved by storage scheduling. However, the ratio of cost savings decreases when 'AVE' increases, because when the electricity prices in different slots are equally increased, the differences among them do not change and the monetary costs with and without storage scheduling increase equally.

C. BEHAVIOR OF TASK SCHEDULING

Fig. 8 shows the correlations among the average unit cost of electricity $\bar{v}_i = F_i^{PC}/Q_i^{cons}$, the number of active servers m_i^{act} and the average request rate to DCs $\bar{\lambda}_i = \frac{1}{HT}\sum_{h=1}^{H}\sum_{j=1}^{J}\lambda_{ji}^{h}$. It can be seen that m_i^{act} increases with the decrease of $\bar{v}_i$, which means that more servers will be activated in areas with lower electricity cost in order to reduce power costs. It is observed that $\bar{\lambda}_i$ is proportional to m_i^{act}. Given the selfishness of players in distributed mechanisms, EGs always prefer low-latency generating DCs, so that the average request rate to DCs in a long duration can be balanced.



(a) Data center 1

(b) Data center 2

FIGURE 8. Diagram of daily $\bar{v}_i$, $\bar{\lambda}_i$, m_i^{act} for each data center.

However, if we observe in real time, periodical fluctuations and oscillations will be found, which is also caused by the selfishness of EGs along with the lack of cooperation among EGs. Benefiting from the form of squared costs in Lyapunov optimization techniques, the Price of Anarchy (PoA) of our proposed distributed task dispatching mechanism is maintained within a reasonable range, as shown in Fig. 9. PoA is a widely used scalar measure of the system performance degradation caused by the anarchy of players. In this paper, we define PoA as the ratio of task latencies derived from our proposed distributed mechanism and from the centralized mechanism. Besides, we can also see from Fig. 9 that the PoA increases with the growth of the number of EGs.

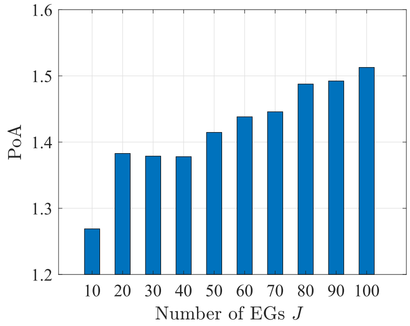


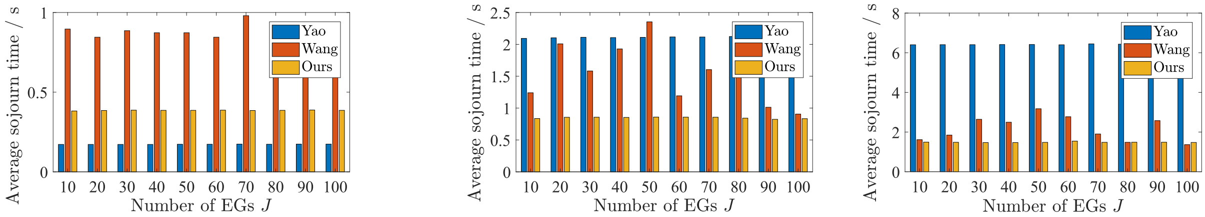
FIGURE 9. Price of Anarchy of the distributed task dispatching mechanism based on the Lyapunov optimization framework.

Table 6 shows the monetary savings of the whole DC system in one time slot achieved by workload scheduling with different configurations of electricity price $p_{i,n}$, number of DCs I and total request rate L_σ, where the default parameters shown in Table 5 are used in the benchmark case and L_{max} is the maximum request rate that the system can deal with to avoid overload. It is observed that the larger the geographical differences in the electricity price, measured by the variances, the more the monetary savings. However, when the prices in different areas increase equally, in which case only the means increase while the variances remain constant, monetary savings will not increase. That is to say the geographical difference in the electricity price is the main contributor to the power savings gained by workload scheduling. In addition, monetary savings will also increase with the growth of I and L_σ, which means that the effect of workload scheduling on power cost savings will be improved when the number of geographical DCs or the total request rate increases.

TABLE 6. Monetary costs saved by workload scheduling with different parameters.

<table>
<tr><td colspan="6">Monetary cost savings with different electricity price</td></tr>
<tr><td>Mean</td><td>Benchmark</td><td colspan="2">Benchmark</td><td>Higher</td><td>Lower</td></tr>
<tr><td>Var</td><td>Benchmark</td><td>Higher</td><td>Lower</td><td colspan="2">Benchmark</td></tr>
<tr><td>Save($)</td><td>436</td><td>967</td><td>176</td><td>429</td><td>432</td></tr>
<tr><td colspan="6">Monetary cost savings with different number of data centers</td></tr>
<tr><td>I</td><td>2</td><td>4</td><td>6</td><td>10</td><td>14</td></tr>
<tr><td>Save($)</td><td>37</td><td>153</td><td>311</td><td>555</td><td>822</td></tr>
<tr><td colspan="6">Monetary cost savings with different total amount of requests</td></tr>
<tr><td>$\frac{L_\sigma}{L_{max}}$</td><td>5%</td><td>20%</td><td>40%</td><td>60%</td><td>80%</td></tr>
<tr><td>Save($)</td><td>65</td><td>186</td><td>340</td><td>436</td><td>323</td></tr>
</table>

Table 7 shows the average request delay of the whole DC system in the case of different numbers of DCs I and different weight parameters $\theta_{1,i}$, where the upper-bound queuing delay D is set in terms of the web requests as 2 s. It can be observed that our formulation will never make the task latency approach or exceed the upper bound that users can tolerate, which is different from the approaches presented in [10]–[25]. That is to say our proposed method can realize the trade-off of power cost and delay cost by a flexible

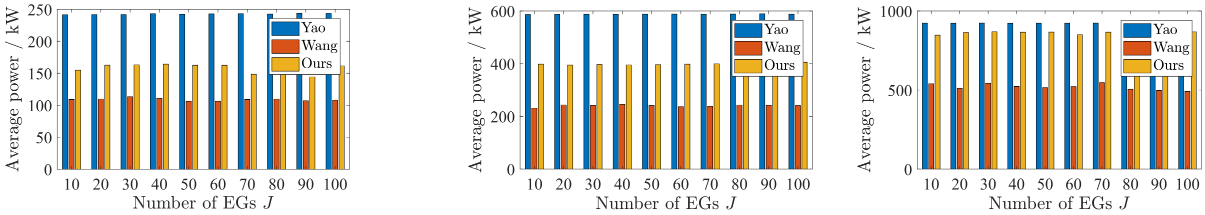


(a) Low workload density (20%)

(b) Medium workload density (50%)

(c) High workload density (80%)

FIGURE 10. **Average sojourn time for different workload densities.**



(a) Low workload density (20%)

(b) Medium workload density (50%)

(c) High workload density (80%)

FIGURE 11. **Average power of DCs for different workload densities.**

TABLE 7. **The average value of request delay ($\theta_{2,i} = 1$).**

$\theta_{1,i}$ \ I	2	4	8
1	112 ms	112 ms	135 ms
100	29 ms	28 ms	27 ms
1000	17 ms	17 ms	17 ms

weight parameter because the request delay is a subpart of the objective function.

Fig. 10 and Fig. 11 show comparison experiments of our proposed scheme to another two-timescale scheme investigated by Yao [24], [25] and to a typical single-timescale scheme investigated by Wang [18]. Specifically, we make observations of the average sojourn time experienced by tasks in DCs and average power of DCs in different situations of workload density, measured by the fraction of the actual request rate and by the system capacity. We can see that Yao's scheme has better latency performance than ours when the workload density is low, but its latency performance exhibits a rapid degradation with the increase of the workload density. On the other hand, our scheme exhibits a slower growth of task latency and is much better than Yao's scheme in situations of medium and high workload densities. This phenomenon mainly benefits from our diverse-timescale decisions on the number of active servers and on task routing, which can better adapt to the uncertainty of task arrivals than Yao's scheme. Additionally, it is observed that the latency performance of Wang's scheme exhibits obvious fluctuations. This is because the task routing strategy and the CPU-cycle frequency can not be adjusted in time to adapt to the time-varying workload density. We note that dynamical CPU-frequency strategies are conducted in small timescales in both Yao's scheme and our scheme. Moreover, the power consumption performance of our scheme is between those of Yao's and Wang's.

TABLE 8. **Monetary cost savings with workload scheduling and storage scheduling.**

$\frac{L_\sigma}{L_{max}}$	Benchmark price			Price with larger VAR		
	$I = 4$	$I = 6$	$I = 8$	$I = 4$	$I = 6$	$I = 8$
0.4	481 15.1%	800 17.0%	1027 17.5%	592 21.1%	1089 26.1%	1390 25.2%
0.6	708 14.5%	1099 15.0%	1529 15.7%	798 18.2%	1323 20.4%	1745 20.1%
0.8	930 13.8%	1454 14.4%	2016 14.9%	1280 20.7%	1941 21.1%	2598 21.2%

D. EVALUATION OF THE POWER COST REDUCTION

Table 8 shows the monetary cost savings obtained by the joint optimization of workload scheduling and power storage scheduling in problem $\mathcal{P}$. It is observed that the monetary cost savings obtained by the joint optimization are larger than those obtained by workload scheduling in Table 6 or power storage scheduling in Fig. 7b alone, and will increase with the growth of I, L_σ and the variance of the electricity price. This means that (i) the joint scheduling of workload and power storage can further improve the power cost savings, and (ii) the amount of cost savings is directly proportional to the number of DCs, the request rate, and the spatial and temporal change rate of electricity price. Specifically, we find that cost savings of up to 10%–30% of the total monetary costs can be obtained by using workload and storage scheduling. However, this kind of power cost reduction depends on the consideration of spatial and temporal differences of unit power cost, which has nearly no social benefits if the unit cost is decoupled from clean power generation [13]. Therefore, we will next check the efficiency of our proposed scheme from the point of view of clean power usage.

Fig. 12 shows the monetary costs and the total power costs in the case of different pollution factors $\gamma_{i,n}$ and that without $\gamma_{i,n}$, where '1×' and '2×' represent $[\gamma_{i,TP}, \gamma_{i,SP}, \gamma_{i,WP}]$

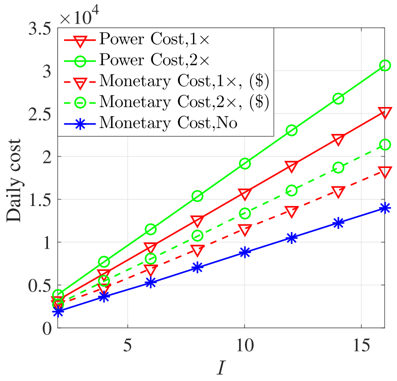


FIGURE 12. The Power costs and monetary costs obtained by the joint optimization.

TABLE 9. The fraction of clean power to the total power bought.

$\gamma_{i,n}$	TP	SP	WP
No	81.34%	18.66%	0%
×1	53.63%	36.68%	9.68%
×2	39.49%	42.73%	17.78%

equal to $[0.5, 0.4, 0.3] \times 10^{-3}$ and $[1, 0.8, 0.6] \times 10^{-3}$, respectively. The corresponding pollution costs can be obtained by deducting the monetary costs from the total power costs. On the other hand, Table 9 shows the corresponding fractions of bought clean power. It is observed that when introducing the pollution factors $\gamma_{i,n}$ or increasing the differences among the $\gamma_{i,n}$ of different ECs, the monetary costs grow, while the fraction of clean power is improved. This means that our proposed method can realize a trade-off between monetary costs and pollution costs, by which the fraction of clean power can be improved to 50%–60% of the total bought power.

V. CONCLUSION AND FUTURE WORK

In this paper, we focused on the eco-friendly and delay-aware power cost minimization of geo-distributed DCs in a deregulated electricity market. We formulated this problem as problem $\mathcal{P}$ and proposed the SCP algorithm to obtain the globally optimal non-integer solution. Based on this, we obtained the quasi-optimal mixed-integer solution of problem $\mathcal{P}$. Simulation results revealed that our method could effectively cut down the total power cost and encourage an eco-friendly use of power, as well as reduce the delay of requests, achieving monetary cost savings of up to 10%–30%. More importantly, our proposed PIF-based power cost model can greatly improve the use of cleaner power and encourage power saving. For example, for a 1-MW DC, the amount of clean power it uses can reach 60% of the total power it buys.

Some interesting improvements toward the adjustment strategies of the integer-constrained variables, not reported here due to space constraints, can be found in [35]. As part of our future work, we will explore better methods to predict the future unit cost of power storage and distributed algorithms to realize workload scheduling under transmission delay considerations.

APPENDIX A
PROOFS ABOUT THE OPTIMALITY OF THE SCP ALGORITHM

In Proposition 1, we prove that the optimal solution for those $q_{i,n}$ with $q^{*L}_{i,n} < 0$ in problem (7) is zero. Based on this, we prove that our proposed SCP algorithm can obtain the globally optimal solution of problem $\mathcal{P}'$ via Proposition 2 and Proposition 3.

Proposition 1: Ignoring Δ_i, $\forall q_{i,n}$ in problem (7), if the optimal Lagrange solution $q^{*L}_{i,n}$ obtained by Eq. (23) is less than zero, then its optimal solution $q^{*}_{i,n}$ of problem (7) with the constraint $q_{i,n} \geq 0$ is zero, where $i \in \mathcal{I}$, $n \in \mathcal{N}_i$.

Proof: First, we assume that there is only one element of the optimal Lagrange solutions obtained by Eq. (23) that is less than zero, whose index is denoted by (i, m), $m \in \mathcal{N}_i$, $i \in \mathcal{I}$, i.e., $q_{i,m}$ is the only variable with $q^{*L}_{i,m} < 0$. Then it is true that $F^{PC}_i(\mathbf{q}^{*L}_i) < F^{PC}_i(q_{i,m}, \mathbf{q}^{*L}_i(q_{i,m}))$, where $\mathbf{q}^{*L}_i(q_{i,m})$ is the optimal Lagrange solution, obtained by Eq. (23) after removing m from $\mathcal{N}_i$, with the constraint $\sum_{n\in\mathcal{N}_i\&n\neq m} q_{i,n} = Q^{cons}_i + Q^{bat}_i - q_{i,m}$.

Secondly, we denote $f_n(q_{i,n}) = a_{i,n}q^2_{i,n} + p_{i,n}q_{i,n}$, and rewrite (7) as $F^{PC}_i(\mathbf{q}_i) = \sum_n a_{i,n}q^2_{i,n} + p_{i,n}q_{i,n} = \sum_n f_n(q_{i,n})$. Then (A-1) can be obtained as

$$\begin{aligned}
&F^{PC}_i(\mathbf{q}^{*L}_i) < F^{PC}_i(q_{i,m}, \mathbf{q}^{*L}_i(q_{i,m}))\\
&\Rightarrow -\sum\nolimits_{n\neq m} \frac{\left\{f_n(q^{*L}_{i,n}(q_{i,m})) - f_n(q^{*L}_{i,n})\right\}}{q_{i,m} - q^{*L}_{i,m}}\\
&\quad< \frac{f_m(q_{i,m}) - f_m(q^{*L}_{i,m})}{q_{i,m} - q^{*L}_{i,m}}\\
&\Rightarrow \sum\nolimits_{n\neq m} \frac{\left\{f_n(q^{*L}_{i,n}(q_{i,m})) - f_n(q^{*L}_{i,n})\right\}}{\alpha_{i,n}\left\{q^{*L}_{i,n}(q_{i,m}) - q^{*L}_{i,n}\right\}}\\
&\quad< \frac{f_m(q_{i,m}) - f_m(q^{*L}_{i,m})}{q_{i,m} - q^{*L}_{i,m}}\\
&\Rightarrow \sum\nolimits_{n\neq m} \frac{\left\{f_n(q^{*L}_{i,n}(0)) - f_n(q^{*L}_{i,n})\right\}}{\alpha_{i,n}\left\{q^{*L}_{i,n}(0) - q^{*L}_{i,n}\right\}} < \frac{f_m(0) - f_m(q^{*L}_{i,m})}{0 - q^{*L}_{i,m}}.
\end{aligned} \tag{A-1}$$

Here, $q^{*L}_{i,n}(0)$ is the $q^{*L}_{i,n}$ obtained by Eq. (23) when $q_{i,m} = 0$, and $\alpha_{i,n}$ is equal to $a_{i,n}X_{i,m}$ according to (23), where $X_{i,m}$ is X_i after eliminating m from $\mathcal{N}_i$.

Thirdly, because all $f_n(\cdot)$ are strictly increasing and convex, and $q^{*L}_{i,m} < 0 < q^{+}_{i,m}$ where $q^{+}_{i,m}$ represents $q_{i,m} > 0$, we can obtain (A-2) as follows.

$$\frac{f_m(q^{+}_{i,m}) - f_m(0)}{q^{+}_{i,m} - 0} > \left.\frac{\mathrm{d}f_m(x)}{\mathrm{d}x}\right|_{x=0} > \frac{f_m(0) - f_m(q^{*L}_{i,m})}{0 - q^{*L}_{i,m}} \tag{A-2}$$

In addition, it can be proved by contradiction that $q^{*L}_{i,n}(q^{+}_{i,m}) < q^{*L}_{i,n}(0) < q^{*L}_{i,n}$. Thus we can obtain (A-3) as below.

$$\frac{f_n(q_{i,n}^{*L}(q_{i,m}^{+})) - f_n(q_{i,n}^{*L}(0))}{q_{i,n}^{*L}(q_{i,m}^{+}) - q_{i,n}^{*L}(0)} < \left.\frac{\mathrm{d}f_n}{\mathrm{d}x}\right|_{x=q_{i,n}^{*L}(0)} < \frac{f_n(q_{i,n}^{*L}(0)) - f_n(q_{i,n}^{*L})}{q_{i,n}^{*L}(0) - q_{i,n}^{*L}} \quad \text{(A-3)}$$

Finally, based on(A-1), (A-2) and (A-3), we can obtain (A-4) as

$$\begin{aligned} &\sum\nolimits_{n \neq m} \frac{\left\{f_n(q_{i,n}^{*L}(q_{i,m}^{+})) - f_n(q_{i,n}^{*L}(0))\right\}}{\alpha_{i,n}\left\{q_{i,n}^{*L}(q_{i,m}^{+}) - q_{i,n}^{*L}(0)\right\}} < \frac{f_m(q_{i,m}^{+}) - f_m(0)}{q_{i,m}^{+} - 0} \\ &\Rightarrow \sum\nolimits_{n \neq m}\left\{f_n(q_{i,n}^{*L}(q_{i,m}^{+})) - f_n(q_{i,n}^{*L}(0))\right\} < f_m(q_{i,m}^{+}) - f_m(0) \\ &\Rightarrow F_i^{PC}(0, \mathbf{q}_i^{*L}(0)) < F_i^{PC}(q_{i,m}^{+}, \mathbf{q}_i^{*L}(q_{i,m}^{+}))\,. \qquad \text{(A-4)} \end{aligned}$$

This means that the optimal solution for $q_{i,m}$ in (7) whose $q_{i,m}^{*L} < 0$ with the constraint $q_{i,m} \geq 0$ is zero. Based on the above, the case that there is only one $q_{i,n}^{L*} < 0$ has been proved. Then we can similarly prove the case with two, three or more $q_{i,n}^{L*} < 0$. □

Proposition 2: The inner iteration procedure of the SCP algorithm shown in lines 7-9 can obtain the optimal solution of problem (7) for the ith data center.

Proof: First, in the t-th, $t \in \{1, \cdots, T\}$ iteration of Algorithm 1, where T is the total number of iterations, the optimal solution of any $q_{i,n}$ which has $q_{i,n}^{L*} < 0$ is equal to zero according to Proposition 1.

Secondly, $\forall q_{i,n}, n \in \mathcal{N}_i$ whose $q_{i,n}^{L*} \geq 0$, the constraint $q_{i,n} \geq 0$ is a slack constraint [32], so that its optimal solution in (7) is the $q_{i,n}^{L*}$ obtained by Eq. (23).

Thirdly, we need to prove that zero is the final optimal solution of the $q_{i,n}$ set to zero in the t-th iteration, although it is optimal in the t-th iteration. When $t = T - 1$, the optimal solutions of the remaining $q_{i,n}$ can be obtained in the Tth iteration, all of which are no less than zero, and there are no changes of the $q_{i,n}$ set to zero in the $(T-1)$-th iteration. When $1 \leq t < T - 1$, there is at least one new n added to $\mathcal{N}_i^-$ in the $(t+1)$-th iteration. We denote q_{i,n_1} as the one set to zero in the t-th iteration, and then denote $q_{i,n_1}^{*L,t}$ and $\nu_i^{*L,t}$ as the optimal Lagrange solution of q_{i,n_1} and the corresponding marginal cost obtained by (25) before setting q_{i,n_1} to zero in the t-th iteration. Considering $q_{i,n}^{*L}(q_{i,m}^{+}) < q_{i,n}^{*L}(0) < q_{i,n}^{*L}$ as stated in the proof of Proposition 1, when we set $q_{i,n_1} = 0$ in the t-th iteration, the optimal Lagrange solutions of the remaining $q_{i,n}$ obtained in the $(t + 1)$-th iteration are all less than those obtained in the t-th iteration. Thus $\nu_i^{*L,t} > \nu_i^{*L,t+1}$, which can be further extended to $\nu_i^{*L,t_1} > \nu_i^{*L,t_2}$, $t_1 < t_2$. The marginal cost of $q_{i,n_1}^{*L,t}$ is $\nu_i^{*L,t}$. If $q_{i,n_1} = 0$ is not the finally optimal solution, then it would be recalculated as in (23) in the $(t+m)$-th iteration, $m = \{2, \cdots, T - t - 1\}$, which must be larger than zero, and the corresponding marginal cost is $\nu_i^{*L,t+m}$. Then $\nu_i^{*L,t+m} > \nu_i^{*L,t}$, because $q_{i,n_1}^{*L,t+m} > 0 > q_{i,n_1}^{*L,t}$, which is a contradiction. Therefore, we can conclude that zero is the final optimal solution of the $q_{i,n}$ set to zero in the t-th iteration.

Based on the above, the complete $\mathcal{N}_i^-$ can be obtained in the $(T - 1)$-th iteration, in which the optimal solutions of all $q_{i,n}$ are zero, and the optimal solutions of all $q_{i,n} \in \widetilde{\mathcal{N}}_i^-$ can be solved in the T-th iteration. In addition, it can be easily proved that $T \leq \max\{N_1, \cdots, N_I\} + 1$, so that it is true that the inner iteration procedure of the SCP algorithm shown in lines 7-9 can obtain the optimal solution of problem (7) for the ith data center with finite repeated rounds. □

Proposition 3: Algorithm 1 is guaranteed to obtain the optimal non-integer solution of problem $\mathcal{P}$ shown in (11)

Proof: Denote the continuity-relaxed problem $\mathcal{P}$ and problem $\mathcal{P}1$ as problem $\mathcal{P}'$ and problem $\mathcal{P}1'$, respectively. Assume that problem $\mathcal{P}1'$ is convex, which will be proved in Appendix B. If the optimal values of all $q_{i,n}$ substituted into problem $\mathcal{P}1'$ satisfy the non-negativity constraints, then problem $\mathcal{P}1'$ is equal to problem $\mathcal{P}'$ and can be solved by the standard SQP algorithm. Then we will prove that all optimal $q_{i,n} \geq 0$ can be obtained by Algorithm 1 as follows.

Different from the individual power cost problem for a single DC in problem (7), for problem $\mathcal{P}'$, the final optimal solution of the $q_{i,n}$ set to zero in the t-th iteration may be larger than zero. Thus we add lines 10–12 into Algorithm 1 to move the n whose $q_{i,n}^{L*,t} < 0$ but $q_{i,n}^{*} > 0$ back to $\widetilde{\mathcal{N}}_i^-$. We denote $q_{i,m}$ as the one whose $q_{i,n}^{L*,t} < 0$ but $q_{i,n}^{*} > 0$, denote $\epsilon^{+} > 0$ as an infinitesimal positive number, and denote $F_i^{PC,t+m}$ as the optimal power cost for the i-th data center in the $(t + m)$-th iteration, where $m = \{1, \cdots, T - t\}$. The marginal cost of $q_{i,m} = 0$ is $p_{i,m}$ and those of all $q_{i,n}$ solved by Lagrange method in the $(t + m)$-th iteration are $\nu_i^{*L,t+m}$. When keeping the total amount of power $\sum\limits_{n \in \mathcal{N}_i \& n \neq m} q_{i,n} + q_{i,m}$ unchanged, $\lim\limits_{\epsilon^{+} \to 0} F_i^{PC,t+m}(\epsilon^{+}) = F_i^{PC,t+m}(0) + p_i^m \cdot \epsilon^{+} - \nu_i^{*L,t+m} \cdot \epsilon^{+}$, where $F_i^{PC,t+m}(*)$ represents the optimal $F_i^{PC,t+m}$ when $q_{i,m} = *$. If $p_{i,m} < \nu_i^{*L,t+m}$, then $F_i^{PC,t+m}(0) > F_i^{PC,t+m}(\epsilon^{+})$, which means $q_{i,m} = 0$ is not the optimum and m needs to be moved back to $\widetilde{\mathcal{N}}_i^-$. Otherwise, $F_i^{PC,t+m}(0) < F_i^{PC,t+m}(\epsilon^{+})$, which means $q_{i,m} = 0$ is still the optimum in the $(t + m)$-th iteration, and is the final optimum when $m = T - t$.

In addition, it can be easily inferred that $T \leq 2 * (\sum\limits_{i=1}^{I} N_i)$. There are two steps leading to the iterations in the SCP algorithm, including (i) setting those $q_{i,n} = 0$ whose optimal Lagrange solution $q_{i,n}^{*L} < 0$ temporarily, (ii) recalculating their optimal values if zeros are not the optimal solutions. In the worst case, all $q_{i,n}^{*L}$ are less than zero, which causes $\sum\limits_{i=1}^{I} N_i$ iterations. Meanwhile, those final optimal solutions are all greater than zero instead of equal to zero, which causes another $\sum\limits_{i=1}^{I} N_i$ iterations, so that the algorithm can converge with no more than $2 * (\sum\limits_{i=1}^{I} N_i)$ iterations.

In conclusion, Algorithm 1 is guaranteed to obtain the optimal solutions of all $q_{i,n}$ limited by $q_{i,n} \geq 0$, so that it is sure to obtain the optimal non-integer solution of problem $\mathcal{P}'$. □

APPENDIX B
CONVEXITY CONDITION FOR PROBLEM $\mathcal{P}1$

In this section, we will give proofs about the condition that $\eta_i'(\delta_i)$, or η_i' for short, needs to meet to guarantee the convexity of the continuity-relaxed problem $\mathcal{P}1$ defined on the convex set $\mathbb{C}$. Here, $\mathbb{C}$ is a convex set composed of all constraints in (28) except the integer restrictions for m_i^{act}.

Proposition 4: For any function of $\eta_i'(\delta_i)$ whose first and second partial derivatives are all existent and continuous, as long as the condition $2 \cdot \frac{\mathrm{d}\eta_i'}{\mathrm{d}\Delta_i} + \frac{\mathrm{d}^2\eta_i'}{\mathrm{d}(\Delta_i)^2} \cdot \Delta_i \geq 0$ is met in the convex set $\mathbb{C}$, where $\delta_i = \frac{\Delta_i}{THC_i}$ and $\forall i \in \mathcal{I}$, then problem $\mathcal{P}1$ is a convex programming problem.

Here, T and H are the length and the number of sub-slots, respectively. *Proof:* [Proof] For any function of $\eta_i'(\delta_i)$ whose first and second partial derivatives are all existent and continuous, $\forall i \in \mathcal{I}$, we can obtain the first and second partial derivatives of Φ_i. Based on this, the Hessian Matrix of Φ_i in problem $\mathcal{P}1$ can be easily obtained, and the Leading Principal Minors $\Gamma_{i,1}$, $\Gamma_{i,2}$ and $\Gamma_{i,3}$ are given as follows.

$$\Gamma_{i,1} = \frac{2\theta_{1,i}}{(m_i^{act}\hat{u}_i - \hat{\lambda}_i)^3}$$

$$\Gamma_{i,2} = \frac{4\theta_{1,i}\theta_{2,i}(s_i^{\alpha})^2}{X_i(m_i^{act}\hat{u}_i - \hat{\lambda}_i)^3}$$

$$\Gamma_{i,3} = \frac{8\theta_{1,i}\theta_{2,i}^2(s_i^{\alpha})^2(Q_i^{cons} + \eta_i'\Delta_i + 0.5X_iY_i)}{(X_i)^2(m_i^{act}\hat{u}_i - \hat{\lambda}_i)^3} \cdot \frac{\mathrm{d}^2(\eta_i'\Delta_i)}{\mathrm{d}(\Delta_i)^2}$$

As we have stated that $m_i^{act}\hat{u}_i - \hat{\lambda}_i > 0$, $Q_i^{cons} + \eta_i'\Delta_i \geq 0$, $X_i > 0$, $Y_i > 0$, etc. above, it can be concluded that both $\Lambda_{i,1} > 0$ and $\Lambda_{i,2} > 0$ are true in the convex set $\mathbb{C}$. $\Lambda_{i,3} \geq 0$ is also true in the case of $\frac{\mathrm{d}^2(\eta_i'\Delta_i)}{\mathrm{d}(\Delta_i)^2} = 2 \cdot \frac{\mathrm{d}\eta_i'}{\mathrm{d}\Delta_i} + \frac{\mathrm{d}^2\eta_i'}{\mathrm{d}(\Delta_i)^2} \cdot \Delta_i \geq 0$. Therefore, if $2 \cdot \frac{\mathrm{d}\eta_i'}{\mathrm{d}\Delta_i} + \frac{\mathrm{d}^2\eta_i'}{\mathrm{d}(\Delta_i)^2} \cdot \Delta_i \geq 0$, the Hessian Matrix of Φ_i is positive semidefinite in the convex set $\mathbb{C}$.

Furthermore, because a sum of convex functions is also a convex function according to [32], Φ is convex in the convex set $\mathbb{C}$ and the continuity-relaxed problem $\mathcal{P}1$ is a convex programming problem when $2 \cdot \frac{\mathrm{d}\eta_i'}{\mathrm{d}\Delta_i} + \frac{\mathrm{d}^2\eta_i'}{\mathrm{d}(\Delta_i)^2} \cdot \Delta_i \geq 0$, $\forall i \in \mathcal{I}$ for any $\eta_i'(\delta_i)$ whose first and second partial derivatives are all existent and continuous. □

REFERENCES

[1] Brian Barrett. (2013). *Google's Insane Number of Servers Visualized*. Accessed: Apr. 14, 2020. [Online]. Available: http://www.gizmodo.com/5517041/

[2] X. Zhiku. (2019). *The Crisis of Data Centers in China: 160 Million MW.H in a Year!* Accessed: Apr. 14, 2020. [Online]. Available: https://mc.manuscriptcentral.com/t-its

[3] C. Li, R. Wang, N. Goswami, X. Li, T. Li, and D. Qian, "Chameleon: Adapting throughput server to time-varying green power budget using online learning," in *Proc. IEEE Int. Symp. Low Power Electron. Design*, Sep. 2013, pp. 100–105.

[4] (2017). *BJX Shoudian*. Accessed: Apr. 14, 2020. [Online]. Available: http://shoudian.bjx.com.cn/news/20171212/867092.shtml

[5] Wikipedia. (2016). *Deregulation of the Texas Electricity Market*. Accessed: Apr. 14, 2020. [Online]. Available: https://en.wikipedia.org/wiki/Deregulation_of_the_Texas_electricity_market

[6] NordREG. (2019). *Nordic Energy Regulators: Power Through Cooperation*. Accessed: Apr. 14, 2020. [Online]. Available: http://www.nordicenergyregulators.org/

[7] L. Rao, X. Liu, L. Xie, and W. Liu, "Coordinated energy cost management of distributed Internet data centers in smart grid," *IEEE Trans. Smart Grid*, vol. 3, no. 1, pp. 50–58, Mar. 2012.

[8] J. Luo, L. Rao, and X. Liu, "Temporal load balancing with service delay guarantees for data center energy cost optimization," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 3, pp. 775–784, Mar. 2014.

[9] J. Luo, L. Rao, and X. Liu, "Spatio-temporal load balancing for energy cost optimization in distributed Internet data centers," *IEEE Trans. Cloud Comput.*, vol. 3, no. 3, pp. 387–397, Jul. 2015.

[10] H. Shao, L. Rao, Z. Wang, X. Liu, Z. Wang, and K. Ren, "Optimal load balancing and energy cost management for Internet data centers in deregulated electricity markets," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 10, pp. 2659–2669, Oct. 2014.

[11] Y. Yang, S. Zhao, W. Zhang, Y. Chen, X. Luo, and J. Wang, "DEBTS: Delay energy balanced task scheduling in homogeneous fog networks," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 2094–2106, Jun. 2018.

[12] X. Wang, K. Wang, S. Wu, S. Di, H. Jin, K. Yang, and S. Ou, "Dynamic resource scheduling in mobile edge cloud with cloud radio access network," *IEEE Trans. Parallel Distrib. Syst.*, vol. 29, no. 11, pp. 2429–2445, Nov. 2018.

[13] Z. Liu, M. Lin, A. Wierman, S. Low, and L. L. H. Andrew, "Greening geographical load balancing," *IEEE/ACM Trans. Netw.*, vol. 23, no. 2, pp. 657–671, Apr. 2015.

[14] M. A. Islam, H. Mahmud, S. Ren, and X. Wang, "A carbon-aware incentive mechanism for greening colocation data centers," *IEEE Trans. Cloud Comput.*, vol. 8, no. 1, pp. 4–16, Jan. 2020.

[15] X. Hu, P. Li, K. Wang, Y. Sun, D. Zeng, X. Wang, and S. Guo, "Joint workload scheduling and energy management for green data centers powered by fuel cells," *IEEE Trans. Green Commun. Netw.*, vol. 3, no. 2, pp. 397–406, Jun. 2019.

[16] H. Dou, Y. Qi, W. Wei, and H. Song, "Carbon-aware electricity cost minimization for sustainable data centers," *IEEE Trans. Sustain. Comput.*, vol. 2, no. 2, pp. 211–223, Apr. 2017.

[17] L. Yu, T. Jiang, and Y. Cao, "Energy cost minimization for distributed Internet data centers in smart microgrids considering power outages," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 1, pp. 120–130, Jan. 2015.

[18] R. Wang, Y. Lu, K. Zhu, J. Hao, P. Wang, and Y. Cao, "An optimal task placement strategy in geo-distributed data centers involving renewable energy," *IEEE Access*, vol. 6, pp. 61948–61958, 2018.

[19] L. Yu, T. Jiang, Y. Cao, and Q. Qi, "Joint workload and battery scheduling with heterogeneous service delay guaranteesfor data center energy cost minimization," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 7, pp. 1045–9219, Jan. 2015.

[20] R. Tripathi, S. Vignesh, and V. Tamarapalli, "Optimizing green energy, cost, and availability in distributed data centers," *IEEE Commun. Lett.*, vol. 21, no. 3, pp. 500–503, Mar. 2017.

[21] S. Bahrami, V. W. S. Wong, and J. Huang, "Data center demand response in deregulated electricity markets," *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 2820–2832, May 2019.

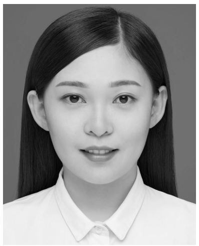
[22] Z. Zhou, F. Liu, Y. Xu, R. Zou, H. Xu, J. C. S. Lui, and H. Jin, "Carbon-aware load balancing for geo-distributed cloud services," in *Proc. IEEE 21st Int. Symp. Modeling, Anal. Simulation Comput. Telecommun. Syst.*, Aug. 2013, pp. 232–241.

[23] R. Tripathi, S. Vignesh, V. Tamarapalli, and D. Medhi, "Cost efficient design of fault tolerant geo-distributed data centers," *IEEE Trans. Netw. Service Manage.*, vol. 14, no. 2, pp. 289–301, Jun. 2017.

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

[25] Y. Yao, L. Huang, A. B. Sharma, L. Golubchik, and M. J. Neely, "Power cost reduction in distributed data centers: A two-time-scale approach for delay tolerant workloads," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 1, pp. 200–211, Jan. 2014.

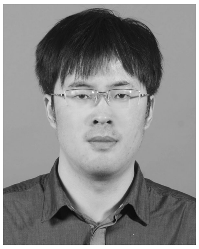
[26] X. Zhu, J. Wang, H. Guo, D. Zhu, L. T. Yang, and L. Liu, "Fault-tolerant scheduling for real-time scientific workflows with elastic resource provisioning in virtualized clouds," *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 12, pp. 3501–3517, Dec. 2016.
[27] J. Cao, K. Li, and I. Stojmenovic, "Optimal power allocation and load distribution for multiple heterogeneous multicore server processors across clouds and data centers," *IEEE Trans. Comput.*, vol. 63, no. 1, pp. 45–58, Jan. 2014.
[28] T. Cui, Y. Wang, S. Chen, and Q. Zhu, "Optimal control of PEVs for energy cost minimization and frequency regulation in the smart grid accounting for battery state-of-health degradation," in *Proc. 52nd ACM/EDAC/IEEE Design Autom. Conf. (DAC)*, pp. 1–6, Jul. 2015.
[29] (2011). *The Charging and Discharging Equations of Batteries*. Accessed: Apr. 14, 2020. [Online]. Available: https://wenku.baidu.com/view/4762a6fdaef8941ea76e056d.html
[30] (Mar. 2004). *Advanced Batteries for Electric-Drive Vehicles: A Technology and Cost Effectiveness Assessment for Battery Electric, Power Assist Hybrid Electric, and Plug-in Hybrid Electric Vehicles*. EPRI. Accessed: Apr. 14, 2020. [Online]. Available: http://gemcardoors.com/EPRI_AdvBatEV.pdf
[31] M. J. Neely, "Stochastic network optimization with application to communication and queueing systems," *Synth. Lectures Commun. Netw.*, vol. 3, no. 1, pp. 1–211, Jan. 2010.
[32] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[33] Y. Yang. (2019). *Decrease the Electricity Price of Solar Power, and Replace Benchmark Price With Guiding Price and Market Bidding*. Accessed: Apr. 14, 2020. [Online]. Available: https://www.thepaper.cn/newsDetail_forward_3389966
[34] (2018). *Benchmark Price of Wind Power in China*. Accessed: Apr. 14, 2020. [Online]. Available: http://www.gov.cn/xinwen/2016-12/28/5153820/files/2c6dd0004bb54a91a6f87628062819ba.pdf
[35] C. Sun, X. Wen, Z. Lu, W. Jing, and M. Zorzi, "Eco-friendly power cost minimization for geo-distributed data centers considering workload scheduling," Nov. 2018, *arXiv:1811.10738*. [Online]. Available: http://arxiv.org/abs/1811.10738



CHUNLEI SUN received the bachelor's degree from Shandong University, in 2014. She is currently pursuing the Ph.D. degree with the Beijing University of Posts and Telecommunications. From 2018 to 2019, she was a Visiting Ph.D. Student at Arizona State University. Her research interests include optimization and scheduling in communication networks, demand side management in energy internet, and wireless sensor networks.

XIANGMING WEN (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in information and communication engineering from the Beijing University of Posts and Telecommunications. He is currently the Director of the Beijing Key Laboratory of Network System Architecture and Convergence, where he has managed several projects related to open wireless networking. He is also the Vice President of the Beijing University of Posts and Telecommunications. His current research interests include radio resource and mobility management, software defined wireless networks, and broadband multimedia transmission technology.

ZHAOMING LU (Member, IEEE) received the Ph.D. degree from the Beijing University of Posts and Telecommunications, in 2012. He joined the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, in 2012. His research interests include open wireless networks, QoE management in wireless networks, software defined wireless networks, and cross-layer design for mobile video applications.



WENPENG JING (Member, IEEE) received the B.S. degree in communication engineering from Shandong University, in 2012, and the Ph.D. degree from the Beijing University of Posts and Telecommunications, in 2017. He currently holds a postdoctoral position at the Beijing University of Posts and Telecommunications. His research interests include green communication, radio resource allocation, and energy efficiency optimization in heterogeneous networks.

MICHELE ZORZI (Fellow, IEEE) received the Laurea and Ph.D. degrees in electrical engineering from the University of Padova, in 1990 and 1994, respectively. From 1992 to 1993, he was on leave from the University of California at San Diego (UCSD). After being affiliated with the Dipartimento di Elettronica e Informazione, Politecnico di Milano, the Center for Wireless Communications, UCSD, and the University of Ferrara, he joined the Faculty of the Information Engineering Department, University of Padova, Italy, in 2003, where he is currently a Professor. His current research interests include performance evaluation in mobile communications systems, random access in mobile radio networks, ad hoc and sensor networks, energy-constrained communications protocols, 5G millimeter-wave cellular systems, and underwater communications and networking. He served for the IEEE Communications Society as a Member-at-Large in the Board of Governors, from 2009 to 2011, as the Director of the Education and Training, from 2014 to 2015, and as the Director of the Journals, from 2020 to 2021. He received many awards from the IEEE Communications Society, including the Best Tutorial Paper Award in 2008 and 2019, the Education Award in 2016, and the S.O. Rice Best Paper Award in 2018. He was the Editor-in-Chief of the IEEE Wireless Communications, from 2003 to 2005, and the IEEE Transactions on Communications, from 2008 to 2011, and the Founding Editor-in-Chief of the IEEE Transactions on Cognitive Communications and Networks, from 2014 to 2018. He was a Guest Editor for several special issues of the IEEE Personal Communications, the IEEE Wireless Communications, the *IEEE Network*, and the IEEE Journal on Selected Areas in Communications.