Sustainable Computing: Informatics and Systems 22 (2019) 13–25

Contents lists available at ScienceDirect

Sustainable Computing: Informatics and Systems

journal homepage: www.elsevier.com/locate/suscom

A market-oriented incentive mechanism for emergency demand response in colocation data centers☆

Youshi Wang[a,b], Fa Zhang[a], Ce Chi[a,b], Shaolei Ren[c], Fangming Liu[d], Rui Wang[e], Zhiyong Liu[a,*]

[a] High Performance Computer Research Center, Institute of Computing Technology, Chinese Academy of Sciences, China
[b] University of Chinese Academy of Sciences, China
[c] University of California, Riverside, USA
[d] Huazhong University of Science and Technology, China
[e] Microsoft, Beijing, China

ARTICLE INFO

Article history:
Received 19 March 2018
Received in revised form 5 October 2018
Accepted 31 January 2019
Available online 6 February 2019

Keywords:
Approximation algorithm
Colocation data center
Energy efficiency
Incentive mechanism
Truthfulness

ABSTRACT

Rapidly developing colocation data centers (or colocations, for short) have become important participants in emergency demand response (EDR) programs. Different from traditional data centers, in colocations, tenants control their own servers; thus, they may not coordinate to reduce their power consumption. In this paper, to encourage tenants to join EDR programs, we propose a market-oriented incentive mechanism, *MicDR*, which can effectively reduce energy costs. *MicDR* includes a local incentive mechanism (*LiMec*), a global incentive mechanism (*GiMec*) and a server-sharing incentive mechanism (*SiMec*). *LiMec* motivates tenants to improve their energy efficiency locally. *GiMec* encourages tenants to improve their energy efficiency by requesting public server resources. To support the requests sparked by *GiMec*, *SiMec* encourages tenants to share idle server resources. A $(1+\epsilon)$-approximation algorithm is proposed to achieve an asymptotic optimal energy-saving scheme. The performance of the proposed algorithm is evaluated, and trace-driven simulations verify the effectiveness and feasibility of *MicDR*.

1. Introduction

Large-scale data centers are power-hungry, but their power demands are flexible [2]. Thus, data centers can participate in demand response (DR) programs, especially in emergency demand response (EDR) programs [3]. EDR is a widely adopted approach to improve the fragile power infrastructure. When emergency events (e.g., extreme weather) occur, EDR providers inform all participants, providing them with a fixed energy-saving target [4]. Then, the participants should reduce their energy consumption to achieve the energy-saving target.

In recent years, one important type of data center, called a colocation data center (or colocation, for short), has developed rapidly. Colocations currently account for approximately 37.3% of all data centers [5]. Colocations help tenants build private data centers by providing professional infrastructure and services, and they are often located in metropolitan areas [6]. Due to the high population densities in metropolitan areas, colocations incur high energy demands, and the available energy is frequently insufficient. Thus, it is necessary for colocations to participate in EDR programs to avoid energy shortages and improve power grid stability [7].

To achieve EDR in these colocations, we focus on ways to improve the energy efficiency of colocations. Energy efficiency technologies have been widely investigated for traditional data centers (e.g., server resource virtualization [8], traffic engineering [9] and energy-efficient data center networks (DCNs) [10]). However, in these works, all the facilities, e.g., servers and infrastructure, were fully controlled by the data center operators. Fig. 1 shows the structure of the colocation data center. The colocation consists of two parts (i.e., the infrastructure and the IT equipment). The infrastructure is managed by the colocation operator. The operator is also responsible for the routine maintenance of the colocation. In contrast, the IT equipment, such as servers in colocations, are fully

Abbreviations: EDR, emergency demand response; *MicDR*, a market-oriented incentive mechanism; *LiMec*, a local incentive mechanism; *GiMec*, a global incentive mechanism; *SiMec*, a server-sharing incentive mechanism; *LG-Mec*, a joint mechanism including *LiMec* and *GiMec*; EC2, elastic compute cloud; PUE, power usage effectiveness; PJM, a regional transmission organization in the United States; MSR, Microsoft Research.

☆ Part of this work was presented at the IEEE 8th Green and Sustainable Computing Conference (IGSC), Orlando, USA, 2017 [1].

* Corresponding author.
E-mail address: zyliu@ict.ac.cn (Z. Liu).

https://doi.org/10.1016/j.suscom.2019.01.020
2210-5379/

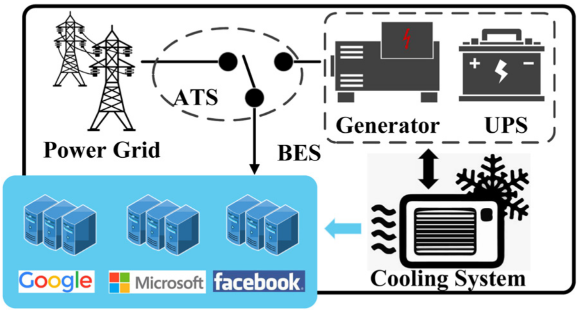


Fig. 1. Structure of a colocation data center.

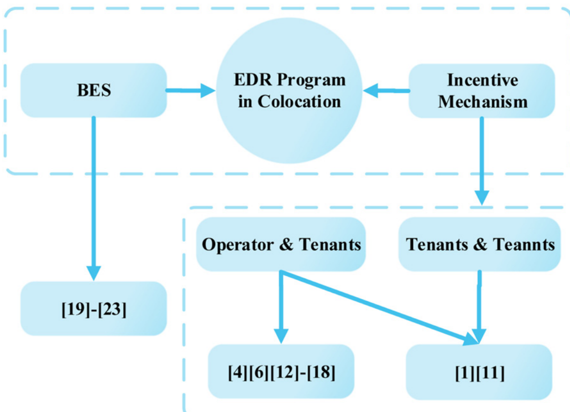


Fig. 2. The relationship among related works.

controlled by tenants. Thus, due to the lack of control of the IT equipment, approaches that focus on improving the energy efficiency of traditional data centers are not feasible for colocations.

The special management pattern of colocations was considered in [11][12] and termed the "uncoordinated relationship" issue, which includes two aspects. First, tenants lack coordination with the colocation operator for energy saving purposes. Some retail tenants pay for energy in advance based on their peak demand. Thus, saving energy is of no benefit to the tenants, and they have no incentive to reduce their energy consumption. Others are wholesale tenants who are charged for energy based on their consumption. This approach ensures that tenants prefer to minimize the energy they use to save costs. However, such tenants may not reduce their energy consumption when an EDR situation occurs. Thus, designing an incentive mechanism to encourage tenants to coordinate with the colocation operator to save energy is a critical problem. Second, tenants lack coordination with each other. By incentivizing the coordination among tenants, the colocation operator can make global optimization decisions rather than relying on the local optimization of each tenant. Thus, designing an incentive mechanism to encourage tenants to coordinate with each other is also a crucial problem.

Many efforts have been made to address the "uncoordinated relationship" between colocation operators and tenants [4,6,12–18]; however, these studies have ignored the problem of coordination among tenants and thus cannot yield good energy efficiency. In this work, we jointly consider coordination between tenants and operators and coordination among tenants.

1.1. Related work

As shown in Fig. 2, there are two main approaches to achieve an EDR program in colocations. One is to use a backup energy storage (BES) system, and another is to design incentive mechanisms. In the mechanism design, two situations of "uncoordinated relationship" issues are considered. One is the "uncoordinated relationship" between the colocation operator and tenants, and the other is the "uncoordinated relationship" among tenants.

Due to the isolation of the colocation operator and tenants, the most popular solution of the EDR program for colocations is to replace the power grid with their own BES system. As shown in Fig. 1, a BES system includes uninterrupted power supply (UPS) and diesel generators. Several prior studies have focused on how to take advantage of BES systems to optimize the total cost of data centers [19–23]. In [19], for saving electricity costs, battery management technology was jointly considered with center-level load balancing as well as the server-level configuration. An optimization framework was proposed in [20] that leverages BES systems in data centers to jointly optimize both peaking shaving and regulation market participation for reducing electricity costs. In [21], a BES system was used to reduce data centers' expenses, considering leakage losses, conversion losses and charging/discharging rates. By leveraging BES systems, the colocation operators can reduce the power demand from the power grid, thus achieving the EDR target. Although this solution bypasses the "uncoordinated relationship" issue, it is far from a good solution due to the high cost and/or high pollution of BES systems.

Some works have considered the energy efficiency issue of colocations. The "split incentive"[1] issue in colocations was first considered in [6], and an incentive mechanism, iCODE, was proposed to incentivize tenants to join the EDR programs. Some subsequent works focused on how to improve the incentive mechanism (e.g., Truth-DR [12]). A joint DR that included economic DR and emergency DR was discussed in [4]. A novel thermal-aware and cost-efficient mechanism called TECH was proposed in [13]. A contract-based mechanism, Contract-DR, was proposed in [14], and different types of tenant costs that applied complete tenants' information and incomplete tenants' information were discussed. In [15], RECO, which provided financial rewards to improve tenants' power management, considered three challenges: the time-varying operation environment, a peak power demand charge and the tenants' unknown responses to the offered rewards. In [16], the demand response provider (DRP) was considered and an incentive mechanism called R2R was proposed to show the interaction between compensation from the DRP and rewards paid to tenants. Some other works exist that consider the incentivization issue in colocations [17,18]. A common feature of the above works is that they all focused on how to incentivize tenants to coordinate with the colocation operator. However, in these studies, the tenants work independently without coordination, which is not conducive to good resource utilization and energy efficiency.

To solve the coordination issue among tenants, a novel framework that incentivizes tenants to reduce their energy consumption via public resources was proposed in [11]. However, three key problems remained unsolved in this framework. First, it did not consider how to ensure the authenticity of tenants' declared cost[2] when they requested public server resources to migrate their workloads. However, guaranteeing truthfulness is a key feature in the design of such a mechanism. Second, the study assumed that the available public server resources could satisfy all the requests. Third, it considered using only some cloud resources and some standby servers to build the public resources. However, it is more cost-efficient to additionally consider tenants' idle servers.

[1] A "split incentive" denotes that while the colocation operator is requested to respond to the energy reduction target from the power grid, the tenants may have no incentive to comply.

[2] "Declared cost" denotes the evaluated energy-saving loss by tenants based on their energy reduction target.

Table 1
Notation description.

$\mathcal{N}$	Set of tenants	M_i	Number of tenant i's servers
β	PUE of the colocation	h_i^{vm}	Number of VM instances held by tenant i's server
$\rho_{i,j}$	Average utilization of tenant i at the time slot j	$C_{i,j}$	Upper bound of tenant i's available servers at the time slot j
E	The whole energy reduction target for the colocation in the EDR program	g_e	Power supply from the BES
τ	BES unit cost	ψ_i^p	Number of needed servers in tenant i's available servers
λ_i, μ_i	Parameters of tenant i's utility function	ψ_i^s	Number of shared servers in tenant i's available servers
ψ_i^{s*}	Optimal solution of ψ_i^s for tenant i'	γ_i^{cost}	Unit cost of shared servers for tenant i
γ^{pay}	Unit payment for a shared server in the colocation	e_i	Energy reduction target of tenant i in *LiMec*
d_i	Declared cost of tenant i in *LiMec*	$\mathcal{B}$	Bid set in *LiMec*
b_i^l	Tenant i's bid in *LiMec*	s_i	Energy reduction target of tenant i in *GiMec*
c_i	Declared cost of tenant i in *GiMec*	g_i	VM instances demand of tenant i in *GiMec*
$\mathcal{B}'$	Bid set in *GiMec*	b_i^g	Tenant i's bid in *GiMec*
D^c	Number of cloud VM instances from cloud providers	δ	Unit price of a VM instance for cloud providers
G	Number of VM instances in the shared cloud	ϵ	The approximation ratio parameter for Algorithm 3
T_{opt}	The theoretical optimal cost of problem (P_1)	T_l	The lower bound of T_{opt}
T_u	The upper bound of T_{opt}	K	A normalized parameter
$F()$	The cost function of building shared cloud		

1.2. Contributions

In this paper, we design a market-oriented incentive mechanism (*MicDR*) that not only encourages coordination between tenants and the colocation operator but also encourages coordination among tenants to further improve resource utilization and reduce energy consumption. To achieve the coordination among tenants, *MicDR* allows tenants to request server resources from a shared cloud to further integrate tasks. In the shared cloud, the colocation operator can improve resource utilization by integrating heterogeneous tasks and resources based on centralized control. Then, to support tenants' requests and reduce the cost for public server resources, tenants are motivated to share their idle server resources. The main contributions of this paper are summarized as follows.

- We propose a novel mechanism, *MicDR*, which not only encourages tenants to optimize their local servers, but also incentivizes tenants to improve their energy efficiency based on a shared cloud. Meanwhile, based on a Stackelberg game, *MicDR* is also designed to incentivize tenants to share their idle servers to supplement the shared cloud.
- *MicDR* is formulated as a mixed-integer nonlinear programming (MINLP) problem, and we develop a $(1+\epsilon)$-approximation algorithm to solve it. For the developed algorithm, we discuss its time complexity and prove that it can satisfy the truthfulness requirement of *MicDR*.
- Based on the developed algorithm, we provide detailed proofs that *MicDR* is a truthful and feasible mechanism. We also demonstrate that *MicDR* can achieve the Nash equilibrium when tenants share their idle servers to supplement the shared cloud.
- We validate the efficiency of the proposed mechanism and algorithm though simulations based on real workload traces and show that *MicDR* can achieve significant energy efficiency improvements in colocations.

1.3. Organization

The rest of this paper is organized as follows. In Section 2, we introduce the colocation model, the EDR model and the tenant server price model. In Section 3, a novel mechanism is proposed to solve the "uncoordinated relationship" issue, and we formulate the cost minimization problem for the proposed mechanism. We solve the problem though an effective algorithm developed in Section 4. Then, the truthfulness and feasibility of the proposed mechanism are proved in Section 5. In Section 6, we show the simulation results and verify the practical performance of our mechanism. Finally, we conclude this paper in Section 7.

2. System model

In this section, we introduce the colocation model and the server price model. The parameters used in this paper are listed and described in Table 1.

2.1. Colocation and EDR model

We consider a colocation data center with n tenants denoted as $\mathcal{N}=\{1, 2, \ldots, n\}$. For tenant $i \in \mathcal{N}$, we use M_i to denote the number of its servers. The Power Usage Effectiveness (PUE) is used to describe the ratio of a colocation's total energy consumption to its IT energy consumption, denoted as β. The computing capacity of a server is considered to be that of an m4.large virtual machine (VM) instance, which is the latest generation of general purpose instances in the Amazon Elastic Compute Cloud (EC2). We assume that all the tenants' servers are homogeneous and that each server can hold h_i^{vm} VM instances. Then, we divide one day into twenty-four time slots and measure the average utilization of tenant i during time slot j, denoted as $\rho_{i,j}$. We use $C_{i,j}$ to denote the upper bound of the available servers for tenant i at time slot j, where $C_{i,j}=\lfloor(1-\rho_{i,j})\cdot M_i\rfloor$.

When the colocation is notified of an EDR program, the colocation operator is requested to reduce the energy demand on the power grid, and the reduction target is denoted as E, which is a fixed value for the colocation operator. A traditional method adopted by the colocation operator is to run its own BES system. The BES's power supply is denoted as g_e, where τ is the unit cost of power generation.

2.2. Tenant server price model

In this subsection, a model is introduced to measure the utility of tenants' available servers. For tenant $i \in \mathcal{N}$, we use $u_i(\psi_i^p, C_{i,j})$ to denote its utility when the tenant is using ψ_i^p servers for its own needs. Based on the law of diminishing returns, a logarithmic function is used to evaluate the utility $u_i(\psi_i^p, C_{i,j})$ [24][25], given as

$$u_i(\psi_i^p, C_{i,j}) = \lambda_i \ln(\mu_i \frac{\psi_i^p}{C_{i,j}}),$$

where λ_i and μ_i are coefficients of the utility function. We assume that tenants can share some servers for public use. For tenant i, the number of shared servers is denoted as ψ_i^s with the unit cost γ_i^{cost};

thus, the total cost for sharing servers can be expressed as $\psi_i^s \cdot \gamma_i^{cost}$. In addition, tenant *i* can also earn some income from the colocation operator. Let γ^{pay} denote the unit payment for a shared server. By sharing servers, tenant *i* can obtain as the following: $\psi_i^s \cdot \gamma^{pay}$.

3. Incentive mechanism design

For solving the "uncoordinated relationship" issue, we propose a joint incentive mechanism called *MicDR* to solve the "uncoordinated relationship" issue and satisfy the energy reduction target of an EDR program. In the following, we first introduce the implementation details and mathematical formulations for these three sub-mechanisms of *MicDR*. Then, considering the framework of *MicDR*, the cost minimization problem of colocations is formulated.

3.1. Designs of the sub-mechanisms

3.1.1. LiMec

The "uncoordinated relationship" issue between tenants and the colocation operator stems from a lack of incentives: tenants have no incentive to optimize their own servers if they do not profit by doing so. An intuitive approach is to provide some economic benefits to incentivize tenants to improve their energy efficiency. Thus, we adopt *LiMec*, a local incentive mechanism, to incentivize tenants to cooperate with the colocation operator to save energy during an EDR program.

In *LiMec*, tenants can be regarded as the sellers and the operator as the buyer. Thus, it forms a typical auction pattern called a "reverse auction" [6]. For tenant $i \in \mathcal{N}$, let e_i and d_i denote the energy-saving target and the declared cost, respectively. In *LiMec*, tenant *i*'s bid can be expressed as $b_i^l = (e_i, d_i)$. Then, we use $\mathcal{B} = \{b_1^l, b_2^l, \ldots, b_n^l\}$ to denote the set of bids.

3.1.2. GiMec

Beyond the lack of incentives, another important issue is that each tenant is relatively independent in the colocation. Thus, the "uncoordinated relationship" issue also exists among tenants. We discuss two situations in which tenants gain by cooperating with each other. First, tenant cooperation can help reduce energy waste by maintaining high server usage rates. For example, for each tenant, when a task executes on and requires only one VM instance, keeping a server running or turning it on may waste energy. However, if the tenants were to cooperate by merging tasks, a higher level of energy efficiency could be maintained. Second, when tenants cooperate, tasks can be integrated more efficiently. Tasks can be divided into several types (e.g., CPU-bound and I/O-bound). Each tenant's tasks may belong to a single type. Assuming that most tasks are the I/O-bound type, optimally integrating these tasks will free many CPU resources and cause idle servers. In contrast, when tenants cooperate with each other, all the resources can be better allocated by integrating different types of tasks to achieve higher utilization and thus better energy efficiency. Therefore, in addition to *LiMec*, we also design *GiMec*, a global incentive mechanism, which helps tenants cooperate in further optimizing colocation utilization and energy efficiency.

Unlike in *LiMec*, it is infeasible to pay economic benefits directly to tenants to incentivize them to cooperate with each other. Thus, we propose to provide a shared cloud so that tenants can integrate different types of tasks. The minimum unit in the shared cloud is a VM instance. In *GiMec*, we collect tenants' information in the form of bids. For tenant $i \in \mathcal{N}$, g_i denotes the number of required VM instances, and s_i and c_i denote tenant *i*'s energy-saving target and declared cost, respectively. Therefore, tenant *i*'s bid can be expressed as $b_i^g = (s_i, c_i, g_i)$. Then, we use $\mathcal{B}' = \{b_1^g, b_2^g, \ldots, b_n^g\}$ to denote the set of bids.

3.1.3. SiMec

In *GiMec*, an important issue is how to build the shared cloud. First, because colocations are typically built in downtown areas close to customers, cloud providers such as Amazon and Google have increasingly deployed part of their servers in these colocations. Thus, to satisfy tenants' VM instance demands, the colocation operator can rent cloud VM instances. We assume that the colocation operator rents D^c VM instances at a unit price of δ from the cloud providers.

Considering that tenants in the colocation usually have some idle servers, it is feasible to incentivize tenants to share their idle servers to build the shared cloud. Thus, we design *SiMec*, which is a server-sharing incentive mechanism, to build a supply-and-demand relationship between the colocation operator and the tenants based on a Stackelberg game which includes a leader and multiple followers. Specifically, the leader moves first, and the followers then react sequentially [26]. In *SiMec*, the colocation operator (who is responsible for building the shared cloud for *GiMec*) acts as the leader, and the tenants (who share some servers to maximize their profits) are regarded as the followers. The colocation operator first issues the unit price for shared servers; then the tenants react independently and selfishly to the unit price to decide how many servers they will share [27]. The shared cloud is composed of both the cloud resources and the servers shared by tenants.

For tenant $i \in \mathcal{N}$, the total profit U_i is composed of three parts: the utility of the servers it needs, the server-sharing cost and the payment amount from the colocation operator. Then, we have

$$U_i(\psi_i^s) = u_i(C_{i,j} - \psi_i^s, C_{i,j}) - \gamma_i^{cost}\psi_i^s + \gamma^{pay}\psi_i^s. \tag{1}$$

To maximize its own profit, tenant *i* can play a subgame, given as

$$\psi_i^{s*} = arg \max_{\psi_i^s \in [0, C_{i,j}]} U_i(\psi_i^s), \tag{2}$$

where ψ_i^{s*} is the optimal solution of ψ_i^s for tenant *i* to achieve the maximum revenue. By calculating the stationary point, the optimal number of shared servers for tenant *i* can be expressed as follows:

$$\psi_i^{s*} = \begin{cases} \left\lfloor C_{i,j} - \dfrac{\lambda_i}{\gamma^{pay} - \gamma_i^{cost}} \right\rfloor & if \quad \gamma^{pay} > \gamma_i^{cost} \\ 0 & if \quad \gamma^{pay} \le \gamma_i^{cost}. \end{cases} \tag{3}$$

We assume that the parameters $C_{i,j}$, λ_i and γ_i^{cost} can be estimated by a machine-learning algorithm. Thus, the colocation operator can predict the reaction of tenant $i \in \mathcal{N}$ for any unit payment γ^{pay}.

3.2. A market-oriented incentive mechanism: MicDR

The framework of *MicDR* is shown in Fig. 3, and the detailed steps are as follows: (1) *MicDR* begins when an EDR program notification reaches the colocation operator and specifies the energy reduction target. (2) The colocation operator announces the beginning of *MicDR*, and tenants can decide whether to submit bids to improve their own energy efficiency. This is a reverse auction process. (3) Tenants make their bids based on their own traffic loads and task types. Two sub-mechanisms can be chosen independently. Accordingly, tenant *i* can determine the bids b_i^l and b_i^g in the sub-mechanisms *LiMec* and *GiMec*, respectively. Then, all bids are collected and submitted to the colocation operator. (4) Based on all collected bids, the colocation operator makes the optimal decision to achieve the overall energy-saving target with the minimal cost. However, to satisfy the capacity of the shared cloud, sub-mechanism *SiMec* builds a server sharing market between the colocation operator and tenants based on the Stackelberg game theory. Thus, in the fourth step, the colocation operator sets the market price γ^{pay} of shared servers and notifies all tenants. (5) Tenants

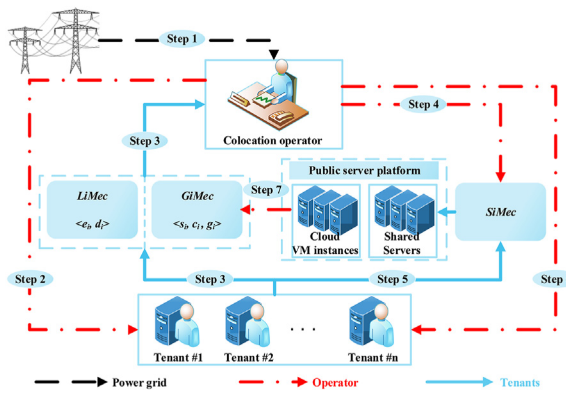


Fig. 3. Framework of *MicDR*. Step 1: Receiving the energy reduction requirement; Step 2: Launching the incentive mechanism *MicDR*; Step 3: Submitting bids; Step 4: Issuing the unit price of shared servers; Step 5: Reacting to the unit price; Step 6: Selecting bids and payment; Step 7: Resources supply and energy reduction.

receive the market price γ^{pay}, and the optimal number of shared servers is computed based on Eq. (3). Then, to obtain the benefits, tenants share $\psi_i^{s*}(i \in [1, \mathcal{N}])$ servers under sub-mechanism *SiMec*. (6) The colocation operator pays for all selected bids in *LiMec* and *GiMec*. Based on γ^{pay} and $\psi_i^{s*}(i \in [1, \mathcal{N}])$, the colocation operator also pays for tenants' earnings of sharing idle servers. (7) The shared cloud provides server resources based on the bids $b_i^g(i \in [1, \mathcal{N}])$, and the energy efficiency optimization officially launches in the colocation.

In *MicDR*, we assume that two bids in *LiMec* and *GiMec* from the same tenant can be selected independently. The VM instances in the shared cloud involve two factors. First, the colocation operator can rent D^c VM instances from the cloud providers. Second, based on *SiMec*, tenant $i \in \mathcal{N}$ can share ψ_i^s servers. Then, the cost to the colocation operator includes the payments for the selected bids in *LiMec* and *GiMec*, the cost of using its own BES system and the payment for the VM instances in the shared cloud.

Thus, the cost minimization problem for the colocation operator can be formulated as problem (P_1), as follows:

$$(P_1) \quad \min \delta D^c + \tau g_e + \sum_{i \in \mathcal{N}} \left(\gamma^{pay} \cdot h_{vm} \cdot \psi_i^{s*} + d_i x_i + c_i y_i \right), \tag{4}$$

$$s.t. \quad g_e + \beta \sum_{i \in \mathcal{N}} (e_i x_i + s_i y_i) \geq E, \tag{4a}$$

$$\sum_{i \in \mathcal{N}} g_i y_i \leq D^c + h_{vm} \cdot \sum_{i \in \mathcal{N}} \psi_i^{s*}, \tag{4b}$$

$$\gamma^{pay}, \psi_i^{s*} \in \{0, \mathcal{Q}^+\}, \tag{4c}$$

$$D^c, g_e \in \mathcal{N}, \tag{4d}$$

$$x_i, y_i \in \left\{0, 1\right\}, \quad i \in \mathcal{N}, \tag{4e}$$

where x_i denotes whether tenant i's bid is selected in *LiMec* and y_i denotes whether tenant i's bid is selected in *GiMec*. Constraint ((4)a) guarantees the energy reduction target, and Constraint ((4)b) implies that the required VM instance in *GiMec* is less or equal to that in the shared cloud.

4. Algorithm design and analysis

In this section, we first discuss how to obtain a feasible approximate solution to problem (P_1) and then analyze the developed algorithms' features and theoretical performances.

(P_1) is a mixed-integer nonlinear programming (MINLP) problem, and it is NP-hard in general [28]. Compared with the problem formulated in Truth-DR [12], an extra constraint, Constraint ((4)b), is added in our problem, which is about the capacity constraint of shared cloud. To our knowledge, considering the new constraint, no feasible algorithm exists to solve the problem. Moreover, there are two requirements for the algorithm design. First, the developed algorithms should solve problem (P_1) with a reasonable time complexity. Second, the truthfulness of *MicDR* must be guaranteed by the design of the algorithm. Thus, we first divided and reformulated problem (P_1) and developed algorithms for the resulting problems. Then, by combining all the developed algorithms, we can obtain a $(1+\epsilon)$-approximation solution for (P_1).

We use T_{opt} to denote the theoretical optimal cost of (P_1) and assume that T_l and T_u can satisfy $T_l \leq T_{opt} \leq T_u$. Let $(x_i', y_i')(i \in \mathcal{N})$ denote a feasible solution vector for (P_1); thus, T_u can be expressed as $T_u = \sum_{i \in \mathcal{N}} (d_i x_i' + c_i y_i')$. Note that if it is difficult to find a feasible solution vector, we let $T_u = \min\{\tau E, \sum_{i \in \mathcal{N}} (d_i + c_i) + \tau \cdot \max\{(E - \sum_{i \in \mathcal{N}} (e_i + s_i)), 0\}\}$. Then, T_l is expressed as $T_l = \min_{i \in \mathcal{N}} \{\tau E, \eta e_i, \eta s_i\}$, where η is a constant and denotes the lower bound of the smallest energy-saving unit price. We define

$$K = \frac{\varepsilon \cdot T_l}{2n},$$

where ε is a parameter related to the approximate ratio of Algorithm 3. K is a scaling parameter that is used to solve the dual problem of (P_1). It is also important to guarantee the approximation ratio of Algorithm 3. We will show how to obtain the value of K in Lemma 2.

Let $d_i' = \lceil \frac{d_i}{K} \rceil$ and $c_i' = \lceil \frac{c_i}{K} \rceil$. Then, we use $F(G)$ to denote the cost of building a shared cloud with at least G VM instances. Based on *SiMec*, the colocation operator can obtain $\left(D^c + h_{vm} \cdot \sum_{i \in \mathcal{N}} \psi_i^s\right)$ VM instances. Thus, we can formulate the cost minimization problem for $F(G)$:

$$F(G) = \min \delta D^c + \gamma^{pay} \cdot h_{vm} \cdot \sum_{i \in \mathcal{N}} \psi_i^{s*},$$

$$s.t. \quad D^c + h_{vm} \cdot \sum_{i \in \mathcal{N}} \psi_i^{s*} \geq G.$$

Accordingly, we can transform (P_1) to (P_2):

$$(P_2) \quad \min F(G) + \tau g_e + K \sum_{i \in \mathcal{N}} \left(d_i' x_i + c_i' y_i \right), \tag{5}$$

$$s.t. \quad g_e + \beta \sum_{i \in \mathcal{N}} e_i x_i + s_i y_i \geq E, \tag{5a}$$

$$\sum_{i \in \mathcal{N}} g_i y_i \leq G, \tag{5b}$$

$$x_i, y_i \in \{0, 1\}, \quad i \in \mathcal{N}. \tag{5c}$$

To simplify (P_2), we first consider the case with fixed G and g_e, denoted as G_f and $g_{e,f}$, respectively. The minimum cost of building a shared cloud with at least G_f VM instances can be denoted as $F(G_f)$. Then, problem (P_2) can be simplified as follows:

$$(P_3) \quad \min F(G_f) + \tau g_{e,f} + K \sum_{i \in \mathcal{N}} \left(d_i' x_i + c_i' y_i \right), \tag{6}$$

$$s.t. \quad \beta \sum_{i \in \mathcal{N}} e_i x_i + s_i y_i \geq E', \tag{6a}$$

$$\sum_{i \in \mathcal{N}} g_i y_i \leq G_f, \tag{6b}$$

$$x_i, y_i \in \{0, 1\}, \quad i \in \mathcal{N}. \tag{6c}$$

where $E' = E - g_{e,f}$.

Because $F(G_f)$ and $\tau g_{e,f}$ are both constants in problem (P_3), we can ignore them when calculating the optimal solution vector of (P_3). Then, based on (P_3), we can formulate the dual problem (P_3'):

$$(P_3') \quad \max \sum_{i\in\mathcal{N}} e_i x_i + s_i y_i, \tag{7}$$

$$s.t. \quad \sum_{i\in\mathcal{N}} d_i' x_i + c_i' y_i \le T', \tag{7a}$$

$$\sum_{i\in\mathcal{N}} g_i y_i \le G_f, \tag{7b}$$

$$x_i, y_i = \{0, 1\}, \quad i \in \mathcal{N}, \tag{7c}$$

where $T' \in \mathcal{T}$(i.e. $\mathcal{T} = \{\lceil \frac{T_l}{K} \rceil, \lceil \frac{T_l}{K} \rceil + 1, \ldots, 2n + \sum_{i=1}^{n}(d_i' x_i' + c_i' y_i')\}$).

To solve the dual problem (P_3'), we adopt the dynamic programming (DP) approach. Let (k, t, G_f') denote a state of DP where $k \in \{0, 1, \ldots, 2n\}$, $t \in \{0, 1, \ldots, T'\}$ and $G_f' \in \{0, 1, \ldots, G_f\}$. Thus, for the state (k, t, G_f'), a sub-problem of (P_3') is to maximize the energy saving for the first k bids, where the payment for the selected bids is no greater than t and the required VM instance is less or equal to G_f', given as

$$(P_3'') \quad \max \sum_{1\le i\le \min\{n,k\}} e_i x_i + \sum_{1\le i\le k-n} s_i y_i, \tag{8}$$

$$s.t. \quad \sum_{1\le i\le \min\{n,k\}} d_i' x_i + \sum_{1\le i\le k-n} c_i' y_i \le t, \tag{8a}$$

$$\sum_{1\le i\le k-n} g_i y_i \le G_f', \tag{8b}$$

$$x_i, y_i = \{0, 1\}, \quad i \in \mathcal{N}. \tag{8c}$$

The optimal solution of the state (k, t, G_f') is denoted as $OPT'(k, t, G_f')$. The corresponding optimal decision variables are denoted as $\mathbf{x}'(k, t, G_f')$ and $\mathbf{y}'(k, t, G_f')$, where $\mathbf{x}'(k, t, G_f') = \{x_i'(k, t, G_f') \mid i \in \mathcal{N}\}$ and $\mathbf{y}'(k, t, G_f') = \{y_i'(k, t, G_f') \mid i \in \mathcal{N}\}$. For the DP process, the initial state is set to

$$\begin{cases} OPT'\left(k = 0, t \ge 0, G_f' \ge 0\right) = 0 \\ OPT'\left(t < 0 || G_f' < 0\right) = -INF \end{cases}$$

We consider two different cases to design the state transition equations of the DP process. First, when $k \le n$, we have

$$\begin{aligned} OPT'\left(k, t, G_f'\right) = \quad & \max\{OPT'\left(k-1, t-d_k', G_f'\right) \\ & + e_k, OPT'\left(k-1, t, G_f'\right)\}. \end{aligned}$$

Then, when $n < k \le 2n$, we have

$$\begin{aligned} OPT'\left(k, t, G_f'\right) = \quad & \max\{OPT'(k-1, t-c_{k-n}', G_f' \\ & -g_{k-n}) + s_{k-n}, OPT'\left(k-1, t, G_f'\right)\}. \end{aligned}$$

Based on Algorithm 2, we can obtain the optimal solution set $\{OPT'(2n, t, G_f') \mid t \in \{0, 1, \ldots, 2n + \sum_{i=1}^{n}(d_i' x_i' + c_i' y_i')\}$ && $G_f' \in \{0, 1, \ldots, G_f\}\}$ for the dual problem (P_3'). For each $OPT'(2n, t, G_f')$, the optimal solution vectors are $\mathbf{x}'(2n, t, G_f')$ and $\mathbf{y}'(2n, t, G_f')$.

Because (P_3) is a special case of (P_2), we directly develop a general algorithm to solve (P_2). For (P_2), G and g_e are the design variables. Let $G_{\max} = \sum_{i\in\mathcal{N}} g_i$ denote the upper bound of G, given as $G_f' \in \{0, 1, \ldots, G_{max}\}$. Based on Algorithm 2, we can calculate the optimal set as $\{OPT'(2n, t, G_f') \mid t \in \{0, 1, \ldots, 2n + \sum_{i=1}^{n}(d_i' x_i' + c_i' y_i')\}$ && $G_f' \in \{0, 1, \ldots, G_{\max}\}\}$. Then, Lemma 1 helps in developing an algorithm to solve (P_2) based on the solution to (P_3').

Lemma 1. *When and only when $\beta \cdot OPT'(2n, t, G_f') \ge E - g_{e,f}$, the corresponding solution vectors of (P_3') are feasible solution vectors for (P_2).*

From Algorithm 3, we get (P_2)'s optimal solution $t_{\min}^{P_2}$ and the corresponding vector $\{(x_i(\min\_G_f', \min\_g_e), y_i(\min\_G_f', \min\_g_e) \mid i \in \mathcal{N}\}$. Specifically, we consider the value of each possible pair of G and g_e (i.e., G_f and $g_{e,f}$), and find the optimal solution vector that satisfies $\beta \cdot OPT'(2n, t, G_f') \ge E - g_{e,f}$ with the minimum t. Then, we obtain the optimal solution of problem (P_2) by finding the minimum cost among all possible values of G_f and g_e as shown in lines 14–15 of Algorithm 3. Let $t_{\min}^{P_1}$ denote an approximate optimal solution to problem (P_1). Lemma 2 proves the approximate ratio of Algorithm 3 for the original problem (P_1).

Lemma 2. *The solution vector $\{(x_i(\min\_G_f', \min\_g_e), y_i(\min\_G_f', \min\_g_e) \mid i \in \mathcal{N}\}$ of (P_2) is the $(1+\epsilon)$-approximation solution vector of (P_1).*

Above all, based on Lemma 2, we can find that Algorithm 3 is a $(1+\epsilon)$-approximation algorithm for (P_1).

Finally, we analyze the complexity of the algorithms. First, we consider that Algorithm 1 is an initialization process, so its time complexity is $O(1)$. The time complexity of Algorithm 2 can be expressed as $((2n + \sum_{i=1}^{n}(d_i' x_i' + c_i' y_i')) \cdot 2n \cdot G_f \cdot n)$. Because $d_i' = \lceil \frac{d_i}{K} \rceil$ and $c_i' = \lceil \frac{c_i}{K} \rceil$, we can get $\sum_{i\in\mathcal{N}}(d_i' x_i' + c_i' y_i') \le \lceil \frac{T_u}{K} \rceil + 2n = \lceil \frac{2nT_u}{\varepsilon T_l} \rceil + 2n$. Thus, the time complexity of Algorithm 2 can be simplified as $O(n^3 \lceil \frac{T_u}{\varepsilon T_l} \rceil G_f)$. Algorithm 3 finds an optimal solution from the optimal solution set $\{OPT'(2n, t, G_f') \mid t \in \{0, 1, \ldots, T'\}$ && $G_f' \in \{0, 1, \ldots, G_{max}\}\}$; therefore, Algorithm 2 can be considered as a part of Algorithm 3. Then, considering the variable $g_e \in [0, E]$, the time complexity of Algorithm 3 can be expressed as $O(n^3 \lceil \frac{T_u}{\varepsilon T_l} \rceil G_{\max} + n \lceil \frac{T_u}{\varepsilon T_l} \rceil E G_{\max})$, which can be simplified to $O(n^3 \lceil \frac{T_u}{\varepsilon T_l} \rceil G_{\max})$.

5. Truthfulness and feasibility

In this section, we show that *MicDR* is a truthful and feasible mechanism. We will discuss how to guarantee the authenticity of the energy-saving costs declared by tenants. Moreover, we also show that tenants will always obtain positive utility when their bids are selected, which means that tenants will tend to participate in our mechanisms. Besides, we will also explain how a Nash equilibrium can be achieved in *SiMec*. Before this analysis, we provide three hypotheses as preconditions:

- Tenants are rational people who know their own preferences and have a clear understanding of their goals. Furthermore, they can make choices independently—meaning that they are not influenced by others during their bidding processes.
- Tenants always make rational choices, which means that random or experiential decisions do not exist when tenants make decisions during the auction.
- Tenants embody the principle of self-interest, which means that they participate in the auction to obtain maximal profit, and they do not pay attention to others.

First, we guarantee the truthfulness of *MicDR* based on the Vickrey–Clarke–Groves (VCG) theory, which is a truthful auction theory that can achieve a socially optimal solution [29–31].

Let $\mathcal{D}$ denote a set of energy-saving bids, expressed as $\mathcal{D} = \{b_1, \ldots, b_{2n}\}$, where $b_i = (m_i, h_i, \tilde{g}_i)$. Tenant $i \in \mathcal{N}$ has two bids included in $\mathcal{D}$, bids b_i and b_{i+n} respectively. In set $\mathcal{D}$, when $i \in [1, n]$, let $m_i = e_i$, $h_i = d_i$ and $\tilde{g}_i = 0$. Then, when $i \in [n+1, 2n]$, let $m_i = s_i$, $h_i = c_i$ and $\tilde{g}_i = g_i$. To better explain the truthfulness, we use a new variable, $V_{\mathcal{D}}^{E}$, to denote $t_{\min}^{P_2}$, where $\mathcal{D}$ is the bid set and E is the energy saving target. Similarly, we use $G_{\mathcal{D}}^{E}$ to denote min_G_f'. Let $\mathcal{D}\backslash\{b_i\}$ $(i \in [1, 2n])$ denote that bid b_i is deleted from set $\mathcal{D}$, which can be written as $\mathcal{D}\backslash\{b_i\} = \{b_1, \ldots, b_{i-1}, b_{i+1}, \ldots, b_{2n}\}$. In set $\mathcal{D}\backslash\{b_i\}$ $(i \in [1, 2n])$, the approximate optimal solution of (P_1) can be denoted as $V_{\mathcal{D}\backslash\{b_i\}}^{E}$. Let $p_i^1 = V_{\mathcal{D}\backslash\{b_i\}}^{E}$ and $p_i^2 = V_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i}$. Then, the market price[3] p_i of bid $b_i \in \mathcal{D}$ can be written as follows:

$$p_i = p_i^1 - p_i^2 = V_{\mathcal{D}\backslash\{b_i\}}^{E} - V_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i}. \tag{9}$$

Let h_i^{true} denote the authentic cost of the bid $b_i \in \mathcal{D}$, and let u_i denote bid b_i's utility. To guarantee the authenticity of bids, Lemma 3 shows that it is impossible for any tenant to obtain higher utility by declaring a false cost.

Lemma 3. *(**Truthfulness**) If a bid $b_i \in \mathcal{D}$ declares a false cost, h_i^{false}, its utility does not increase.*

To guarantee the feasibility of our pricing strategy, Lemma 4 shows that when a bid is selected, its utility is nonnegative.

Lemma 4. *(**Feasibility**) If bid $b_i \in \mathcal{D}$ is selected, its utility is nonnegative.*

Finally, we explain how Nash equilibrium can be achieved based on *SiMec*. *SiMec* is designed based on a Stackelberg game. We use the utility function in Eq. (1) to describe the total profits when tenant i shares ψ_i^s servers. The utility function in Eq. (1) builds a supply-and-demand relationship between the colocation operator and the tenants, but assumes that the tenants operate independently, which means that each tenant also makes decisions independently. Then, tenant $i \in \mathcal{N}$ can calculate the optimal number of shared servers ψ_i^{s*} to maximize its total profits, as shown in Eq. (3). Moreover, because tenants are independent, whether tenant i changes the number of shared servers ψ_i^{s*} has no influence on the decisions of the other tenants. Thus, for tenant $i \in \mathcal{N}$, ψ_i^{s*} can be regarded as its Nash equilibrium point. Accordingly, we find that *SiMec* can achieve Nash equilibrium.

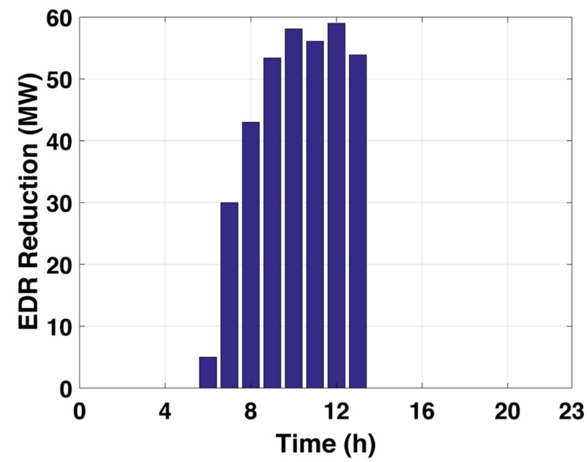
Most importantly, we have shown that *MicDR* can guarantee the authenticity of tenants' bids based on the VCG theory and have also proved that tenants always obtain positive utility when their bids are selected. In addition, we explained how *SiMec* can build a supply-and-demand relationship between the colocation operator and the tenants and achieve the Nash equilibrium. Thus, we can conclude that *MicDR* is a truthful and feasible mechanism.

6. Performance evaluation

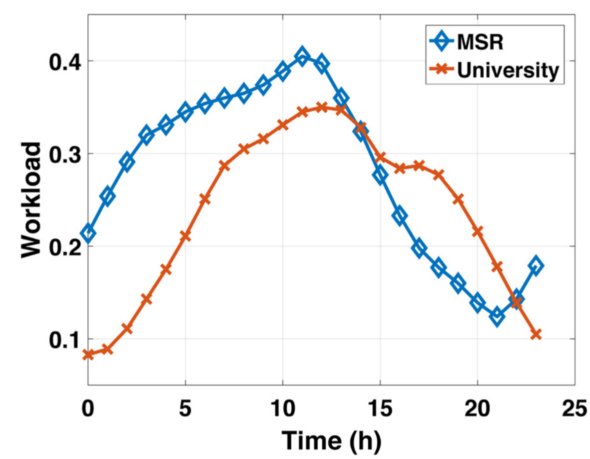
In this section, we present simulations conducted to evaluate the performance of the proposed *MicDR* mechanism. We first introduce the simulation settings, which are based on both widely used parameters [6,12,24,32,33] and real traces [2,34]. Then, we validate the performance of the proposed approximation algorithm. Finally, the simulation results verify the effectiveness and feasibility of *MicDR*.

6.1. Settings

Colocation data center: Assume that there are six tenants (denoted as Tenant #1, Tenant #2, ..., Tenant #6) in the colocation, and each tenant has 10,000 servers. We assume that all the servers in the colocation are homogeneous Dell PowerEdge R730s. We also adopt the m4.large VM instance from the Amazon EC2 to measure the capacity of each server. Under these conditions, each server can hold five m4.large VM instances, i.e., $h_{vm} = 5$. Moreover, we can obtain the price of an m4.large VM instance from an Amazon EC2 spot instance [35]. Specifically, we set δ to 1.55 cents/h based on the price of an m4.large VM instance on October 25, 2017 in Ohio, USA. We assume that the static power P^s and the dynamic power P^d of a server are 0.15 kW and 0.1 kW, respectively [6][33]. A reasonable PUE for the colocation is set to 1.6 [12]. Accordingly, we can obtain the peak power of the colocation, which is 24 MW. Then, based on [12][36], the unit cost of a BES system ranges from 150 $/MWh to 350 $/MWh, where $ 350/MWh is the cost of a typical diesel generator [37].



(a)



(b)

Fig. 4. (a) EDR energy reduction. (b) Workload traces.

Energy reduction target and workload: The energy reduction targets were sourced from PJM's EDR on April 22, 2015 [34], and the data are scaled down to 15% of the colocation's peak power to avoid affecting normal operations [32]. The energy reduction targets are shown in Fig. 4(a). Eight events occurred from 6 am to 13 pm, and each event lasted for one hour. The workload traces were obtained from "MSR" and "Florida International University" [2], as shown in Fig. 4(b).

Tenants' energy reduction and costs: In *MicDR*, we consider that tenants optimize energy efficiency by turning off idle servers [38]. In *SiMec*, these idle servers may be shared to build the shared cloud, which is managed by the colocation operator in a unified manner for achieving higher energy efficiency. From a global view, sharing servers is helpful to further optimize the energy efficiency. In *LiMec*, let $n_{i,l}$ denote the number of turned-off servers for tenant i. Then, we

[3] The market price is the real value of goods in the market.

Table 2
Theoretical performance comparisons among different mechanisms.

	Problem	Approximate ratio	Time complexity
MicDR	**MINLP**	$1+\varepsilon$	$O(n^3\lceil \frac{T_u}{\varepsilon T_l}\rceil G_{\max})$
Truth-DR [12]	MILP	2	n^2
LG-Mec	ILP	$1+\varepsilon$	$O(n^3\lceil \frac{T_u}{\varepsilon T_l}\rceil)$
Branch-bound	INLP	**1**	$n2^n$

find that $e_i = n_{i,l} \cdot P^s \cdot T$, where $T = 1$ *hour* is one EDR period. Next, let $n_{i,g}$ denote the number of servers joining the *GiMec* mechanism for tenant *i*. The utilization of each server is denoted as ρ_k, where $k \in [1, n_{i,g}]$. Thus, we obtain $s_i = T \cdot \sum_{k \in [1,n_{i,g}]}(1-\rho_k)P^s$. We assume that each tenant has its own bid parameter, θ_i, which is used to distinguish the tenants' different expected costs. From [12], we know that the tenants' costs obey a uniform distribution between 6.7 and 13.3 cents/kWh. Thus, based on the tenants' energy reductions, tenant *i*'s costs b_i and c_i can be obtained.

Server price model: The available servers for tenant *i* in time slot *j* is denoted by $C_{i,j} = \lfloor (1-\rho_{i,j}) \cdot M_i \rfloor$, where $M_i = 10,000$ and $\rho_{i,j}$ can be obtained from Fig. 4(b). Then, the parameters γ^{cost} and $\mu_i, i \in \mathcal{N}$, are set to 0 and 100, respectively [24]. Thus, the parameter α_i can be given as $\alpha_i = \frac{\delta C_{i,j}}{\ln 100}$.

6.2. Results and analysis

MicDR is a market-oriented incentive mechanism composed of three sub-mechanisms: *LiMec*, *GiMec* and *SiMec*. In Section 1, we mentioned that most existing works focused on incentivizing tenants to reduce energy consumption as does *LiMec* (e.g., iCODE [6] and Truth-DR [12]). Thus, we choose one typical mechanism, Truth-DR, as a control group. To evaluate the effects of *SiMec*, we also consider a mechanism that includes *LiMec* and *GiMec* but utilizes only cloud VM instances in *GiMec*, termed *LG-Mec* in the following results.

6.2.1. Theoretical performance

We compare the theoretical performance of different mechanisms in Table 2 using the branch-and-bound strategy as a baseline. Compared with *LG-Mec*, *MicDR* involves a more complicated mathematical problem. Moreover, *MicDR* maintains both the same approximate ratio and a similar time complexity as *LG-Mec*. Compared with Truth-DR, *MicDR* can solve more complex problems with a better approximate ratio. The response time of an EDR program is always within several hours. Thus, considering the scale of colocations, the time complexity of *MicDR* is acceptable and can satisfy the time limitation of an EDR program.

6.2.2. Simulation results

In this paper, we first need to validate that *MicDR* always achieves the energy reduction target of EDR program at all time slots, which is shown in Fig. 4(a). As shown in Fig. 5, *MicDR* always satisfies the energy reduction target; consequently, we can conclude that *MicDR* is effective for meeting the EDR's goals. We can also find that *MicDR* maintains a small difference with the EDR target at each time slot. Consider that *MicDR* is designed to satisfy the requirement of an EDR program with minimum costs, this result shows that *MicDR* can always find a cost-efficient solution to avoid higher costs. Moreover, considering the robustness of *MicDR*, which will be discussed in Section 6.3, the power supply g_e from the BES system is retained. Thus, to verify whether *MicDR* can incentivize tenants to save energy, we show the sources of energy reduction in Fig. 6. For each time slot, *MicDR* results in a slight energy reduction from the BES system, which means that the overall energy reduction from tenants alone nearly satisfies the EDR target. Thus, *MicDR*

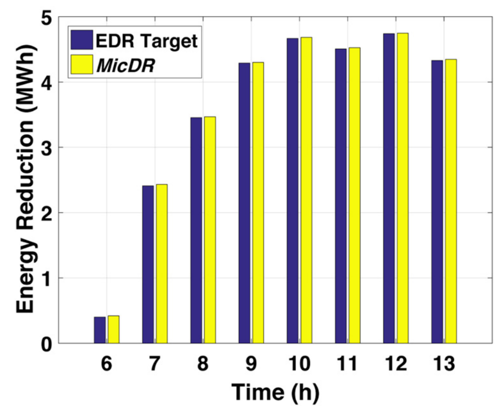


Fig. 5. Comparison between energy reduction targets and total energy reductions using *MicDR*.

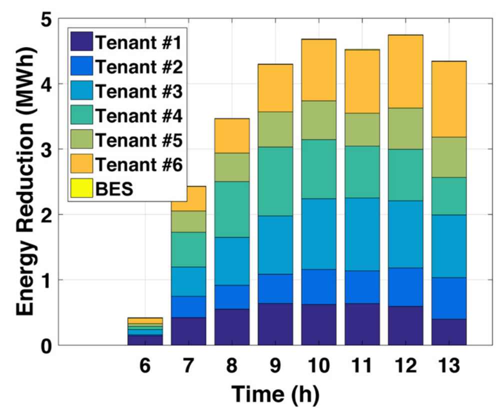
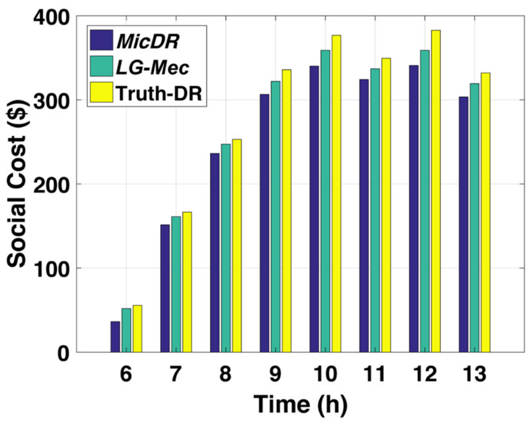


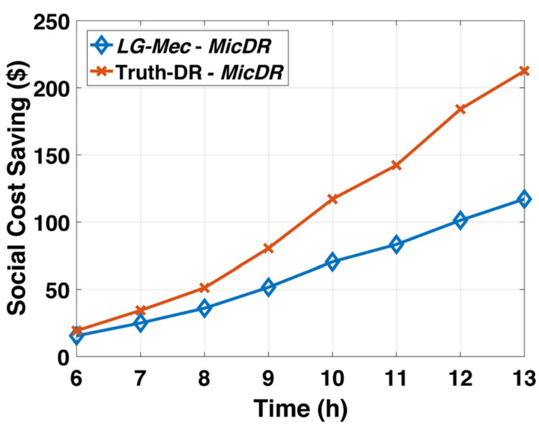
Fig. 6. Comparison of energy reduction by tenants and a BES system.

is a green mechanism, which incentivizes tenants to reduce energy consumption by improving energy efficiency rather than replacing the power grid with a BES system.

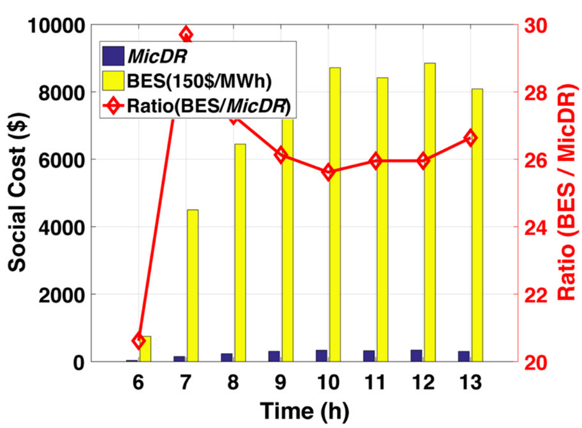
We use social costs to denote the total payments of the colocation operator to achieve the energy reduction target of an EDR program, and it is the main index to measure the mechanism performance in this work. Fig. 7 shows the social costs of different mechanisms. For all tenants, server utilization is calculated based on a normal distribution, where the expectation is the tenant workload. Thus, we obtain the social cost from the average of 150 experiments. We first compare the social cost of different incentive mechanisms in each time slot in Fig. 7(a); then, we show the overall social cost savings between *MicDR* and the other two mechanisms during the EDR program period in Fig. 7(b). It was shown in [12] that Truth-DR achieves a close-to-optimal performance when tenants are incentivized to reduce energy consumption by improving the energy efficiency of their local servers. By introducing the global incentive mechanism *GiMec*, *LG-Mec* obtained an even lower social cost because global resource management and the integration of multiple task types helps further optimize the energy efficiency of colocations. Compared with *LG-Mec*, *MicDR* reduces the cost of building a shared cloud using the server-sharing incentive mechanism *SiMec*. Thus, *MicDR* achieves a lower social cost than *LG-Mec*. Furthermore, in addition to the comparison among different mechanisms, we also compare the social cost between *MicDR* and a BES system in Fig. 7(c). Although using a BES system to achieve an EDR program is not influenced by the "uncoordinated relationship" issue, this approach uses extra power to replace the power from the grid rather than reducing energy consumption. Thus, the energy cost still accounts for a large proportion of the social cost for



(a)



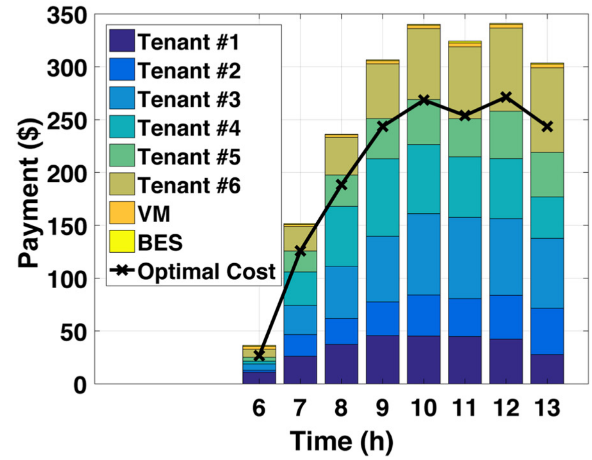
(b)



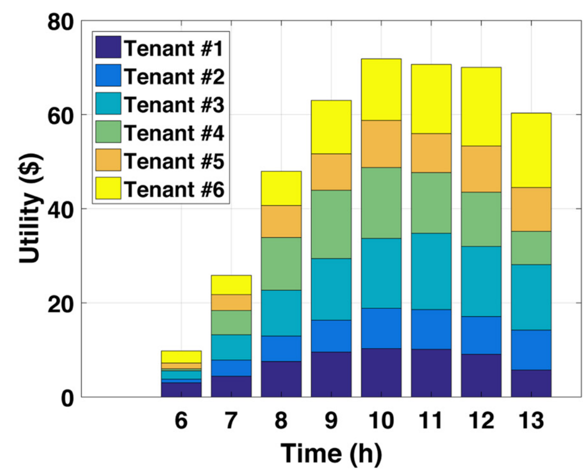
(c)

Fig. 7. Comparison of social costs among different approaches.

the BES system. We use the lower bound of the unit cost to measure the social cost of BES systems, which is 150 $/MWh. However, *MicDR* can effectively reduce much of the energy consumption to satisfy the energy reduction target of EDR programs. As shown in Fig. 7(c), *MicDR* has a lower cost than BES systems, where the social cost of a BES system is more than twenty times that of *MicDR*. The results also show that although *MicDR* needs to pay for incentivizing tenants, it is a more cost-effective approach than replacing energy sources with a BES system.



(a)



(b)

Fig. 8. Average payments and utilities for tenants under *MicDR*.

To verify the features of *MicDR* proposed in Lemma 3 and Lemma 4, we show the average payments of the colocation operator for different tenants and resources in Fig. 8(a). The optimal cost is also added for comparison. By analyzing the social cost components, we find that only a small payment is paid for the power supply from a BES system as well as building the shared cloud. To echo the energy reduction shown in Fig. 6, *MicDR* can achieve a large energy reduction by incentivizing, and it only needs a small amount of power from a BES system. Moreover, by introducing the mechanism *SiMec* to incentivize tenants to share their idle servers to the shared cloud, the colocation operator can use idle servers for the shared cloud with lower costs. Accordingly, *MicDR* successfully achieves a lower social cost by reducing expenses related to extra energy reduction. Based on the average payments to tenants, we calculate the tenants' average utilities and show them in Fig. 8(b). For 150 experiments, the utility of each tenant is always nonnegative. Thus, based on the payments calculated in *MicDR*, tenants do not lose anything when their bids are selected; thus, Lemma 4 is verified. To show the results more intuitively, we select an individual experiment from the 150 experiments randomly in Fig. 9, which shows that not all tenants' bids are selected in one EDR program. By comparing Fig. 9(a) and (b), we find that tenants can obtain utilities only when tenants' bids are selected in *MicDR*. Furthermore, Lemma 4 is

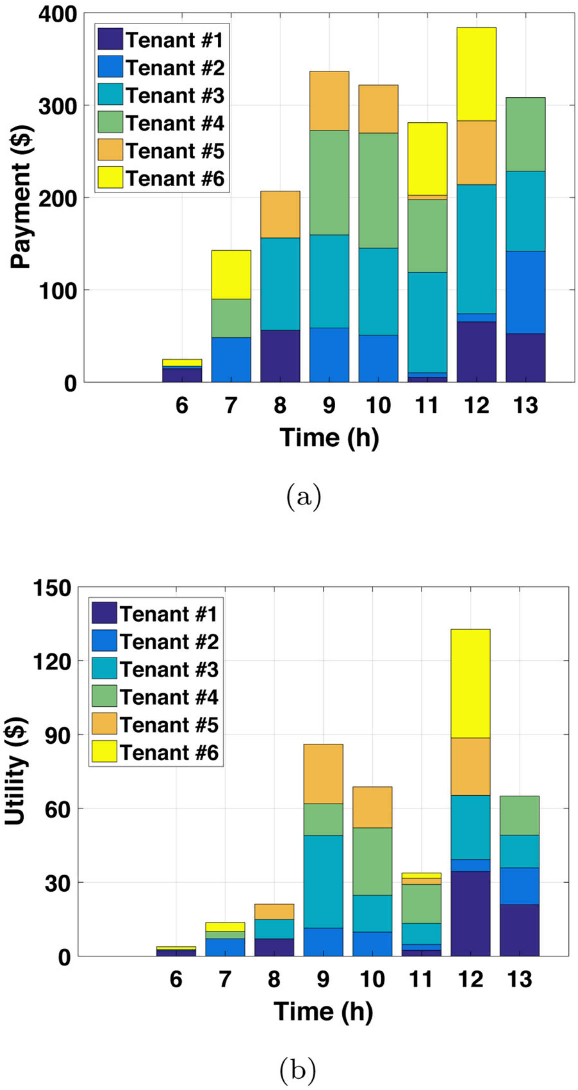


Fig. 9. Payment and utility for tenants based on a randomly selected experiment.

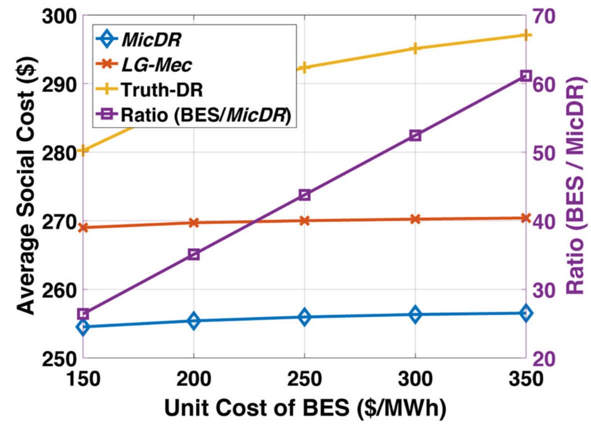


Fig. 10. Comparison of the average social cost for different BES unit prices.

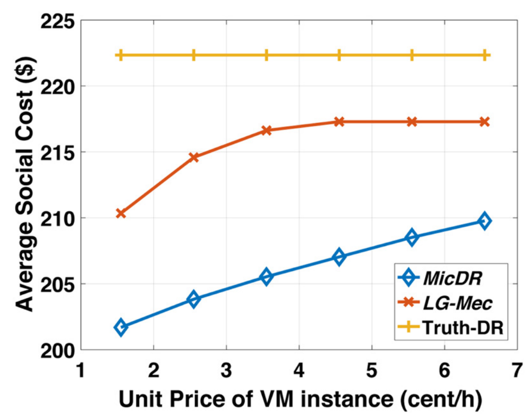


Fig. 11. Comparison of the average social cost for different VM instance prices.

verified again that tenants never receive negative utility when they join the EDR program based on *MicDR*.

Finally, we analyze the influence of three important parameters for the performance of *MicDR*, including the unit cost of a BES system τ, the unit price of a VM instance δ and the static server power P^s. For each parameter, we calculate the average social cost of each mechanism during the EDR program.

- Fig. 10 shows the average social cost comparison among different mechanisms and approaches when the unit cost τ of the BES system changes from 150 $/MWh to 350 $/MWh. We first compare *MicDR* and *LG-Mec*. Because they only need a small power supply from a BES system, the change in τ has a negligible effect on their average social cost. Thus, the social cost difference between *MicDR* and *LG-Mec* remains stable for different τ. However, because Truth-DR uses more BES power compared with *MicDR*, when τ increases, the difference between *MicDR* and Truth-DR also increases. However, the rising tendency of the difference decreases, which indicates that Truth-DR can also reduce the power demand from a BES system for increasing τ. We also show the social cost ratio between a BES system and *MicDR*. Because the average social cost of *MicDR* is approximately constant as τ changes, the ratio has an approximately linear correlation with τ. This finding means that *MicDR* can reduce costs even more for increasing τ. Therefore, we can conclude that (1) both *MicDR* and *LG-Mec* have more stable performance than Truth-DR as well as BES systems when τ changes, and (2) although both have good performance, *MicDR* is better than *LG-Mec*.
- Fig. 11 shows a comparison of the average social cost of the three mechanisms when the unit price of a VM instance δ changes from 1.55 cents/h to 6.55 cents/h. The average social cost of *MicDR* increases almost linearly as δ increases. For *LG-Mec*, as δ changes from 1.55 to 4.55, the speed of increase in its average social cost diminishes, and when $\delta > 4.55$, its average social cost does not change. For Truth-DR, because it does not consider using public resources to improve the energy efficiency, when δ changes, its average social cost remains constant. We can reach two conclusions from Fig. 11. First, compared with Truth-DR, when δ decreases, *MicDR* and *LG-Mec* achieve better performances. In addition, no matter how much δ increases, *MicDR* and *LG-Mec* always perform better than Truth-DR. Second, compared with *LG-Mec*, the average social cost savings of *MicDR* initially increases and then decreases, and it reaches its maximum when $\delta = 3.55$. When $\delta < 3.55$, as δ increases, the influence of the total VM instance cost on the social cost becomes greater. Thus, the average social cost savings of *MicDR* increases as δ rises when $\delta < 3.55$. However, when $\delta > 3.55$, an increase in δ (and the consequent higher VM instance cost) causes fewer global bids to be selected. Thus, the average social cost savings of *MicDR* decreases as δ goes up after $\delta > 3.55$.
- Fig. 12 shows the average social cost comparison of three mechanisms as the static server power P^s changes from 0.15 kW to 0.4 kW. When P^s increases, the average social cost of the three mechanisms increases almost linearly; however, the average

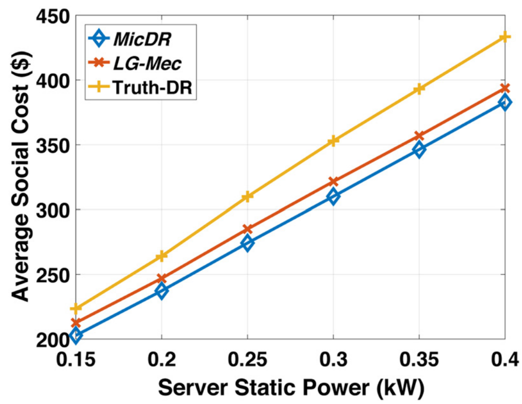


Fig. 12. Comparison of the average social costs at different static server power values.

social cost of Truth-DR increases faster than that of either *MicDR* or *LG-Mec*. Thus, as P^s increases, *MicDR* and *LG-Mec* achieve better performances than does Truth-DR.

6.3. Discussion

Based on the theoretical analysis and the simulation verification, we have shown that *MicDR* can achieve energy reduction targets of EDR programs in colocations with the lowest social cost compared with other mechanisms and approaches. During the mechanism design, the robustness of *MicDR* is also considered to handle special cases. First, if tenants have sensitive data, it may not be acceptable to migrate these data to a shared cloud, which may make the sub-mechanism *GiMec* infeasible. To avoid mutual influence of multiple sub-mechanisms, in *MicDR*, all bids of each tenant are independent and are submitted to different sub-mechanisms. Due to the independence of the bids, tenants can freely make decision about their bids based on their current intention. Thus, *MicDR* is a non-coercive mechanism. However, although some benefits are provided, only a few tenants submit bids to join in *MicDR*. This is the reason why the power supply g_e from a BES system is retained. Based on g_e, although no tenants submit bids, the EDR program in colocations can also be solved via *MicDR*.

Although *MicDR* is a highly cost-effective approach, it also has some limitations, which should be researched in future work. First, as mentioned above, when tenants' data are sensitive, although *MicDR* still works, it may have some performance loss. In the future work, we intend to improve the sub-mechanism *GiMec* to avoid the sensitive data issue and achieve better cooperation among tenants. Moreover, except for obtaining benefits, the quality of service (QoS) loss is also an important factor that influences the decisions of tenants. The QoS loss issue may exhibit two components in *MicDR*: (1) considering the queue model, turning off servers may cause local service delay increases in *LiMec*, and (2) the transmission delay when tenants migrate tasks to a shared cloud in *GiMec* is ignored. *MicDR* is not optimized for delay sensitive situations; thus, the performance of *MicDR* may decline in such situations. In future work, we will consider a biobjective optimization problem, including the social cost as well as the delay.

7. Conclusions

Due to their high energy consumption, colocations play an important role in EDR programs. By analyzing the special management pattern of colocations, we showed that solving the "uncoordinated relationship" issue is the key to improving energy efficiency at colocations. In this paper, we proposed a market-oriented incentive mechanism called *MicDR*, which is composed of three sub-mechanisms (*LiMec*, *GiMec* and *SiMec*) that improve energy efficiency and resource utilization in colocations. *LiMec* and *GiMec* incentivize tenants to reduce their energy consumption, while *SiMec* is designed to create a server-sharing market to provide idle resources to *GiMec* based on a Stackelberg game. Then, for *MicDR*, we formulated a MINLP cost minimization problem and developed a $(1+\epsilon)$-approximation algorithm to solve it. In addition, we also proved that *MicDR* is both a truthful and feasible mechanism. We analyzed the theoretical performance of *MicDR* and compared it with existing works. Our simulations in this study were based on widely used settings and real traces. By comparing *MicDR* with different mechanisms, we showed that it incurs lower social costs for the same energy reduction target, thus validating its effectiveness. By analyzing the relationship between the energy reduction and the utility for each tenant, we also verified the lemmas proposed in this paper.

Acknowledgements

This work is supported partially by the National Natural Science Foundation of China under the grants 61520106005, 61761136014, and the National Key Research and Development Program of China under the grant 2017YFB1010001.

Appendix A. Proof of Lemma 1

For $\beta \cdot OPT'(2n, t, G_f{}') \geq E'(t \in \{0, 1, \ldots, 2n + \sum_{i=1}^{n}(d_i{}'x_i{}' + c_i{}'y_i{}')\}\&\&\ G_f{}' \in \{0, 1, \ldots, G_{\max}\})$, we use $\mathbf{x}'(2n, t, G_f{}')_{\geq E'}$ ($\mathbf{x}'_{\geq E'}$ for short) and $\mathbf{y}'(2n, t, G_f{}')_{\geq E'}$ ($\mathbf{y}'_{\geq E'}$ for short) to denote the corresponding solution vectors. Based on $\mathbf{x}'_{\geq E'}$ and $\mathbf{y}'_{\geq E'}$, for problem $(P_3{}')$, it is easy to obtain $g_{e,f} + \beta(e_i\mathbf{x}'_{\geq E'} + s_i\mathbf{y}'_{\geq E'}) \geq E$ and $g_i\mathbf{y}'_{\geq E'} \leq G_f$, where all the constraints of (P_2) are satisfied. Then, $\mathbf{x}'_{\geq E'}$ and $\mathbf{y}'_{\geq E'}$ are the feasible solution vectors for problem (P_2). When $\beta \cdot OPT'(2n, t, G_f{}') < E - g_{e,f}$, constraint (5)a cannot be satisfied. Then, $\mathbf{x}'_{\geq E'}$ and $\mathbf{y}'_{\geq E'}$ are not feasible solution vectors for problem (P_2). Thus, Lemma 1 has been proved.

Appendix B. Proof of Lemma 2

Let $\{(x_i^{opt}, y_i^{opt}) \mid i \in \mathcal{N}\}$ denote the optimal solution vector for (P_1). The corresponding g_e and $G_f{}'$ are denoted as g_e^{opt} and G^{opt}, respectively. Because the constraints for (P_1) and (P_2) are the same, $\{(x_i^{opt}, y_i^{opt}) \mid i \in \mathcal{N}\}$ is a feasible solution vector for (P_2), and $\{(x_i(\min\_G_f{}', \min\_g_e), y_i(\min\_G_f{}', \min\_g_e) \mid i \in \mathcal{N}\}$ is a feasible solution for (P_1). Thus, we can conclude the following:

$$
\begin{aligned}
T_{opt} &= F(G^{opt}) + \tau g_e^{opt} + \sum_{i \in \mathcal{N}}(d_i x_i^{opt} + c_i y_i^{opt}) \\
&\geq F(G^{opt}) + \tau g_e^{opt} + \sum_{i \in \mathcal{N}}(K(d_i{}' - 1)x_i^{opt} \\
&+K(c_i{}' - 1)y_i^{opt}) \\
&= F(G^{opt}) + \tau g_e^{opt} + K\sum_{i \in \mathcal{N}}(d_i{}'x_i^{opt} + c_i{}'y_i^{opt}) \\
&-2Kn \\
&\geq F(\min\_G_f{}') + \tau \min\_g_e \\
&+K(\sum_{i \in \mathcal{N}}(d_i{}'x_i(\min\_G_f{}', \min\_g_e) \\
&+c_i{}'y_i(\min\_G_f{}', \min\_g_e))) - 2Kn \\
&\geq \left(F(\min\_G_f{}') + \tau \min\_g_e\right) \\
&+(\sum_{i \in \mathcal{N}}(d_i x_i(\min\_G_f{}', \min\_g_e) \qquad +c_i y_i(\min\_G_f{}', \min\_g_e))) - 2Kn \\
&= t_{\min}^{P_1} - 2Kn.
\end{aligned}
$$

Because $K = \frac{\epsilon \cdot T_l}{2n}$, we can obtain

$$t_{\min}^{P_1} \leq T_{opt} + \epsilon \cdot T_l \leq (1+\epsilon) T_{opt}.$$

Thus, we have proved that $t_{\min}^{P_1}$ is the $(1+\epsilon)$-approximation solution of (P_1).

Appendix C. Proof of Lemma 3

The utility of bid $b_i \in \mathcal{D}$ can be expressed as the difference between its market price and its cost. We use u_i^{true} to denote the utility when b_i declares the authentic cost, h_i^{true}, and u_i^{false} to denote the utility when b_i declares a false cost, h_i^{false}. Let $\Delta u_i = u_i^{false} - u_i^{true}$ denote the difference between u_i^{false} and u_i^{true}. Based on the self-interest principle, the false cost must be greater than the authentic cost, i.e., $h_i^{true} < h_i^{false}$.

We consider two cases. First, b_i is not selected when it declares a false cost. Thus, $u_i^{false} = 0$. Because $u_i^{true} \geq 0$, $\Delta u_i = u_i^{false} - u_i^{true} \leq 0$. Accordingly, in this case, Lemma 3 is true. Second, consider the situation that b_i is selected when it declares a false cost. Thus, the cost when b_i is selected is no more than the cost when b_i is not selected, i.e., $V_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i} + F\left(G_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i} + \tilde{g}_i\right) - F\left(G_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i}\right) + h_i^{false} \leq V_{\mathcal{D}\backslash\{b_i\}}^{E}$. Then, when b_i declares a authentic cost, the cost when selecting b_i is less than the cost when b_i is not selected, given as

$$\begin{aligned} & V_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i} + F\left(G_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i} + \tilde{g}_i\right) - F\left(G_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i}\right) \\ & + h_i^{true} \\ & < V_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i} + F\left(G_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i} + \tilde{g}_i\right) \\ & - F\left(G_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i}\right) + h_i^{false} \\ & \leq V_{\mathcal{D}\backslash\{b_i\}}^{E}. \end{aligned}$$

Thus, b_i is selected when it declares the authentic cost. Then, based on Eq. (9), we can get $u_i^{true} = V_{\mathcal{D}\backslash\{b_i\}}^{E} - V_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i} - h_i^{true}$ and $u_i^{false} = V_{\mathcal{D}\backslash\{b_i\}}^{E} - V_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i} - h_i^{true}$. Accordingly, Δu_i is

$$\Delta u_i = u_i^{false} - u_i^{true} = 0. \tag{10}$$

Based on Eq. (10), Δu_i is zero. Thus, for bid $b_i \in \mathcal{D}$, a tenant cannot obtain higher utility by declaring a false cost.

Most importantly, Lemma 3 is proved.

Appendix D. Proof of Lemma 4

Based on Lemma 3, we know that to obtain higher utility, all tenants will declare their costs authentically (i.e., $h_i = h_i^{true}$). The utility of bid $b_i \in \mathcal{D}$ can be expressed as $u_i(b_i)$, given by

$$u_i(b_i) = V_{\mathcal{D}\backslash\{b_i\}}^{E} - V_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i} - h_i.$$

When b_i is selected, based on Algorithm 3, we can obtain

$$V_{\mathcal{D}}^{E} \leq V_{\mathcal{D}\backslash\{b_i\}}^{E},$$

and we can also obtain

$$V_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i} \leq V_{\mathcal{D}}^{E} - F(G_{\mathcal{D}}^{E}) + F(G_{\mathcal{D}}^{E} - \tilde{g}_i) - h_i.$$

Because $F(G_{\mathcal{D}}^{E}) \geq F(G_{\mathcal{D}}^{E} - \tilde{g}_i)$, we find that

$$V_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i} + h_i \leq V_{\mathcal{D}}^{E}.$$

Thus, we obtain

$$V_{\mathcal{D}\backslash\{b_i\}}^{E} \geq V_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i} + h_i,$$

which means that

$$u_i(b_i) = V_{\mathcal{D}\backslash\{b_i\}}^{E} - V_{\mathcal{D}\backslash\{b_i\}}^{E-\beta m_i} - h_i \geq 0.$$

Therefore, Lemma 4 is proved.

Appendix E. Algorithm 1

Algorithm 1. State initialization for (P_3').

Initialize
$T_u = \min\{\tau E', \sum_{i\in\mathcal{N}}(d_i + c_i) + \tau \cdot \max\left\{(E' - \sum_{i\in\mathcal{N}}(e_i + s_i)), 0\right\}\}$,
$T_l = \min_{i\in\mathcal{N}}\{\tau E, \eta e_i, \eta s_i\}$ and $K = \frac{\varepsilon \cdot T_l}{2n}$.
For $i \in \mathcal{N}$, let $d_i' = \lceil \frac{d_i}{K} \rceil$ and $c_i' = \lceil \frac{c_i}{K} \rceil$.
Let $\mathcal{T} = \{\lceil \frac{T_l}{K} \rceil, \lceil \frac{T_l}{K} \rceil + 1, \ldots, 2n + \sum_{i=1}^{n}\left(d_i' x_i' + c_i' y_i'\right)\}$.
Initialize $OPT'\left(k, t, G_f'\right)$. $OPT'(k=0, t \geq 0, G_f' \geq 0) = 0$,
$OPT'\left(t < 0 || G_f' < 0\right) = -INF$, $x_i'\left(k = 0, t \geq 0, G_f' \geq 0\right) = 0$, and
$y_i'\left(k = 0, t \geq 0, G_f' \geq 0\right) = 0$.

Appendix F. Algorithm 2

Algorithm 2. Optimal solution set for (P_3')

for $t = 0$ to $2n + \sum_{i=1}^{n}(d_i' x_i' + c_i' y_i')$ **do**
for $k = 1$ to $2n$ **do**
for $G_f' = 0$ to G_f **do**
if $k \leq n$ **then**
if $OPT'\left(k-1, t-d_k', G_f'\right) + e_k > OPT'\left(k-1, t, G_f'\right)$ **then**
$OPT'\left(k, t, G_f'\right) = OPT'(k-1, t-d_k', G_f') + e_k$,
$x_i'\left(k, t, G_f'\right) = x_i'(k-1, t-d_k', G_f')$, $1 \leq i < k$, and $x_k'\left(k, t, G_f'\right) = 1$
else
$OPT'(k, t, G_f') = OPT'(k-1, t, G_f')$
$x_i'\left(k, t, G_f'\right) = x_i'\left(k-1, t, G_f'\right)$, $1 \leq i < k$, $x_k'\left(k, t, G_f'\right) = 0$
end if
end if
if $k > n$ **then**
if $OPT'\left(k-1, t-c_{k-n}', G_f' - g_{k-n}\right) + s_{k-n} > OPT'\left(k-1, t, G_f'\right)$ **then**
$OPT'\left(k, t, G_f'\right) = OPT'(k-1, t-c_{k-n}', G_f' - g_{k-n}) + s_{k-n}$,
$x_i'\left(k, t, G_f'\right) = x_i'(k-1, t-c_{k-n}', G_f' - g_{k-n})$, $1 \leq i \leq n$,
$y_i'\left(k, t, G_f'\right) = y_i'(k-1, t-c_{k-n}', G_f' - g_{k-n})$, $1 \leq i < k-n$,
$y_{k-n}'\left(k, t, G_f'\right) = 1$
else
$OPT'\left(k, t, G_f'\right) = OPT'\left(k-1, t, G_f'\right)$
$x_i'\left(k, t, G_f'\right) = x_i'\left(k-1, t, G_f'\right)$, $1 \leq i \leq n$,
$y_i'\left(k, t, G_f'\right) = y_i'\left(k-1, t, G_f'\right)$, $1 \leq i < k-n$, $y_{k-n}'\left(k, t, G_f'\right) = 0$
end if
end if
end for
end for
end for
Return $\{OPT'\left(2n, t, G_f'\right) \mid t \in \left\{0, 1, \ldots, 2n + \sum_{i=1}^{n}(d_i' x_i' + c_i' y_i')\right\}$ && $G_f' \in \left\{0, 1, \ldots, G_f\right\}\}$.

Appendix G. Algorithm 3

Algorithm 3. Optimal solution for (P_2)

Input: $\{OPT'\left(2n, t, G_f'\right) \mid t \in \{0, 1, \ldots, 2n + \sum_{i=1}^{n}(d_i' x_i' + c_i' y_i')\}$ && $G_f' \in \left\{0, 1, \ldots, G_{\max}\right\}\}$
Output: $t_{\min}^{P_2}$ and $\{(x_i\left(\min\_G_f', \min\_g_e\right), y_i\left(\min\_G_f', \min\_g_e\right)) \mid i \in \mathcal{N}\}$
for ALL$G_f' \in \left\{0, 1, \ldots, G_{\max}\right\}$ **do**
$g_e = E$
for $t = \lceil \frac{T_l}{K} \rceil$ to $2n + \sum_{i\in\mathcal{N}}\left(d_i' x_i' + c_i' y_i'\right)$ **do**
while $g_e \geq 0$ and $\beta \cdot OPT'\left(2n, t, G_f'\right) \geq E - g_e$ **do**

$x_i\left(G_f{}', g_e\right) = x_i{}'\left(2n, t, G_f{}'\right)$
$y_i\left(G_f{}', g_e\right) = y_i{}'\left(2n, t, G_f{}'\right)$
$g_e = g_e - 1$
end while
if $g_e < 0$ **then**
Break
end if
end for
end for
$(\min\_G_f{}', \min\_g_e) = \mathrm{argmin}_{G_f{}', g_e}\{F(G_f{}') + \tau \cdot g_e + K\sum_{i\in\mathcal{N}}(d_i{}'x_i(G_f{}', g_e) + c_i{}'y_i(G_f{}', g_e))\}$
$t^{P_2}_{\min} = F(\min\_G_f{}') + \tau \cdot \min\_g_e + K\sum_{i\in\mathcal{N}}(d_i{}'x_i(\min\_G_f{}', \min\_g_e) + c_i{}'y_i(\min\_G_f{}', \min\_g_e))$

References

[1] Y. Wang, F. Zhang, S. Ren, F. Liu, R. Wang, Z. Liu, Energy efficiency in colocation data centers: a joint incentive mechanism approach, in: 2017 International Green and Sustainable Computing Conference (IGSC), IEEE, 2017, pp. 1–8.
[2] N. Chen, X. Ren, S. Ren, A. Wierman, Greening multi-tenant data center demand response, Perform. Eval. 91 (2015) 229–254.
[3] A. Wierman, Z. Liu, I. Liu, H. Mohsenian-Rad, Opportunities and challenges for data center demand response, in: 2014 International Green Computing Conference (IGCC), IEEE, 2014, pp. 1–10.
[4] N.H. Tran, C.T. Do, S. Ren, Z. Han, C.S. Hong, Incentive mechanisms for economic and emergency demand responses of colocation datacenters, IEEE J. Select. Areas Commun. 33 (12) (2015) 2892–2905.
[5] M.A. Islam, X. Ren, S. Ren, A. Wierman, X. Wang, A market approach for handling power emergencies in multi-tenant data center, in: 2016 IEEE International Symposium on High Performance Computer Architecture (HPCA), IEEE, 2016, pp. 432–443.
[6] S. Ren, M.A. Islam, Colocation demand response: why do I turn off my servers? ICAC (2014) 201–208.
[7] Is cloud computing always greener? URL http://www.nrdc.org/energy/files/cloud-computing-efficiency-IB.pdf.
[8] S. Verboven, K. Vanmechelen, J. Broeckhove, Black box scheduling for resource intensive virtual machine workloads with interference models, Future Gener. Comput. Syst. 29 (8) (2013) 1871–1884.
[9] L. Wang, F. Zhang, J.A. Aroca, A.V. Vasilakos, K. Zheng, C. Hou, D. Li, Z. Liu, GreenDCN: a general framework for achieving energy efficiency in data center networks, IEEE J. Select. Areas Commun. 32 (1) (2014) 4–15.
[10] B. Heller, S. Seetharaman, P. Mahadevan, Y. Yiakoumis, P. Sharma, S. Banerjee, N. McKeown, Elastictree: saving energy in data center networks, Nsdi, vol. 10 (2010) 249–264.
[11] Y. Wang, F. Zhang, Z. Liu, Truthful strategy and resource integration for multi-tenant data center demand response, International Workshop on Frontiers in Algorithmics (2015) 259–270, Springer.
[12] L. Zhang, S. Ren, C. Wu, Z. Li, A truthful incentive mechanism for emergency demand response in colocation data centers, in: 2015 IEEE Conference on Computer Communications (INFOCOM), IEEE, 2015, pp. 2632–2640.
[13] Z. Zhao, F. Wu, S. Ren, X. Gao, G. Chen, Y. Cui, Tech: a thermal-aware and cost efficient mechanism for colocation demand response, in: 2016 45th International Conference on Parallel Processing (ICPP), IEEE, 2016, pp. 464–473.
[14] K. Ahmed, M.A. Islam, S. Ren, A contract design approach for colocation data center demand response, in: 2015 IEEE/ACM International Conference on Computer-Aided Design (ICCAD), IEEE, 2015, pp. 635–640.
[15] M.A. Islam, H. Mahmud, S. Ren, X. Wang, Paying to save: reducing cost of colocation data center via rewards, in: 2015 IEEE 21st International Symposium on High Performance Computer Architecture (HPCA), IEEE, 2015, pp. 235–245.
[16] N.H. Tran, T.Z. Oo, S. Ren, Z. Han, E.-N. Huh, C.S. Hong, Reward-to-reduce: an incentive mechanism for economic demand response of colocation datacenters, IEEE J. Select. Areas Commun. 34 (12) (2016) 3941–3953.
[17] L. Niu, Y. Guo, H. Li, M. Pan, A Nash bargaining approach to emergency demand response in colocation data centers, in: 2016 IEEE Global Communications Conference (GLOBECOM), IEEE, 2016, pp. 1–6.
[18] Y. Guo, M. Pan, Coordinated energy management for colocation data centers in smart grids, in: 2015 IEEE International Conference on Smart Grid Communications (SmartGridComm), IEEE, 2015, pp. 840–845.
[19] Y. Guo, Y. Fang, Electricity cost saving strategy in data centers by using energy storage, IEEE Trans. Parallel Distrib. Syst. 24 (6) (2013) 1149–1160.
[20] Y. Shi, B. Xu, B. Zhang, D. Wang, Leveraging energy storage to optimize data center electricity cost in emerging power markets, in: Proceedings of the Seventh International Conference on Future Energy Systems, ACM, 2016, p. 18.
[21] M. Dabbagh, B. Hamdaoui, A. Rayes, M. Guizani, Shaving data center power demand peaks through energy storage and workload shifting control, IEEE Trans. Cloud Comput. (1) (2017) 1.
[22] M. Reyes, O. Martinez, I. Gil, E. Domingez, S. Vazquez, K. McGrath, W. Beez, Flexible and cost effective hybrid energy storage system based on batteries and ultracapacitors, in: 2015 IEEE International Conference on Industrial Technology (ICIT), IEEE, 2015, pp. 1013–1018.
[23] L. Yu, T. Jiang, Y. Cao, Energy cost minimization for distributed internet data centers in smart microgrids considering power outages, IEEE Trans. Parallel Distrib. Syst. 26 (1) (2015) 120–130.
[24] K. Poularakis, G. Iosifidis, I. Pefkianakis, L. Tassiulas, M. May, Mobile data offloading through caching in residential 802.11 wireless networks, IEEE Trans. Netw. Serv. Manage. 13 (1) (2016) 71–84.
[25] C. Courcoubetis, R. Weber, Pricing Communication Networks: Economics, Technology and Modelling, John Wiley & Sons, 2003.
[26] Stackelberg competition. URL https://en.wikipedia.org/wiki/Stackelberg_competition.
[27] T. Roughgarden, Stackelberg scheduling strategies, SIAMJ. Comput. 33 (2) (2004) 332–350.
[28] R.G. Michael, S.J. David, Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman Co., San Francisco, 1979, pp. 245–248.
[29] W. Vickrey, Counterspeculation, auctions, and competitive sealed tenders, J. Finan. 16 (1) (1961) 8–37.
[30] E.H. Clarke, Multipart pricing of public goods, Public Choice 11 (1) (1971) 17–33.
[31] T. Groves, Incentives in teams, Econometrica (1973) 617–631.
[32] G. Ghatikar, V. Ganti, N. Matson, M.A. Piette, Demand Response Opportunities and Enabling Technologies for Data Centers: Findings from Field Studies, Tech. Rep., Lawrence Berkeley National Lab.(LBNL), Berkeley, CA, USA, 2012.
[33] Y. Wang, F. Zhang, R. Wang, Y. Shi, H. Guo, Z. Liu, Real-time task scheduling for joint energy efficiency optimization in data centers, in: 2017 IEEE Symposium on Computers and Communications (ISCC), IEEE, 2017, pp. 838–843.
[34] Demand response activity. URL http://www.pjm.com/~/media/markets-ops/demand-response/pjm-cold-days-reports-for-april-21-22-2015.ashx.
[35] Amazon elastic compute cloud spot instances pricing. URL https://aws.amazon.com/cn/ec2/spot/pricing/.
[36] E. Consulting, Pricing Data Center Co-location Services. URL http://www.enaxisconsulting.com/blog/white-papers/pricing-data-center-co-location-services/.
[37] Wikipedia, Diesel Generator. URL http://en.wikipedia.org/wiki/Dieselgenerator.
[38] M. Lin, A. Wierman, L.L. Andrew, E. Thereska, Dynamic right-sizing for power-proportional data centers, IEEE/ACM Trans. Netw. 21 (5) (2013) 1378–1391.