# Adaptive Capacity Provisioning for Carbon-Aware Data Centers: A Digital Twin-based Approach

Zhiwei Cao*, Ruihang Wang*, Xin Zhou†, Rui Tan*,*Senior Member, IEEE*, Yonggang Wen*,*Fellow, IEEE*, Yuejun Yan ‡, and Zhaoyang Wang ‡
*Nanyang Technological University, Singapore
†Jiangxi Science and Technology Normal University, China
‡Alibaba Group, China
*zhiwei003, ruihang001, tanrui, ygwen@ntu.edu.sg, †1020161201@jxstnu.edu.cn
‡yanyuejun.yyj, chaoyang.wzy@alibaba-inc.com

***Abstract*—This paper considers the carbon-aware data center (DC) capacity provisioning problem under uncertain green energy availability and computing demand. To address it, accurate carbon emissions estimation and robust capacity provisioning are necessary. Existing studies mainly consider the carbon footprint of the computing system and merely consider that of the physical facilities, which also contribute significant carbon emissions. Furthermore, their capacity provisioning is neither uncertainty-aware nor adaptive to the dynamic computing demand. To bridge these gaps, we propose an adaptive capacity provisioning framework based on the physics-informed digital twin. We design the digital twin to holistically capture a DC's operational carbon footprint, including both the computing system and the physical facilities. The digital twin is differentiable and established with a collection of physics-informed learnable models that are learned with online operational data. We further address the challenge of capacity provisioning under uncertainties by designing a shrinking horizon model predictive control. The designed capacity planner updates its estimation of future computing demand based on the observable computing system states. At each capacity provisioning round, we solve the capacity provisioning problem using a gradient-based optimization technique with the gradient provided by the digital twin. We extensively evaluate our approach using *real* operational data from a large-scale production data center. Firstly, our digital twin accurately predicts holistic DC energy usage with a relative absolute error of less than 5%, which is accurate according to the industrial rule of thumb. Second, we show that our solution is comparable to the Oracle solution with perfect knowledge about all uncertainties, outperforming the state-of-the-art Predict-then-Plan approach significantly in terms of SLO violation reduction. Furthermore, our approach reduces carbon footprint by 27% compared with the over-provisioning scheme currently adopted by the industry.**

***Index Terms*—Data center, digital twin, capacity provisioning, model predictive control**

## I. Introduction

Data centers (DCs), as supporting pillars for the digital world and artificial intelligence (AI), have been growing rapidly in scale recently, raising public concern about their sustainability. Currently, DCs consume around 3% global electricity usage and contribute around 1-2% global carbon emissions [1]. This ratio is more significant in tropical countries where DCs are expected to contribute 12% electricity usage by 2030 due to the huge demand for cooling in the tropics [2]. Furthermore, as the energy efficiency improvement is stagnant, DC decarbonization becomes more challenging [3]. In this regard, many cloud providers and DC operators have announced their sustainability goals to reduce or neutralize the carbon footprint of their systems. For example, Google aims to achieve 24/7 carbon-neutral DC operation by 2030 [4]. To achieve this, they propose a carbon-aware computing system to modulate their workload according to the carbon intensity [1] of the power grid [5].

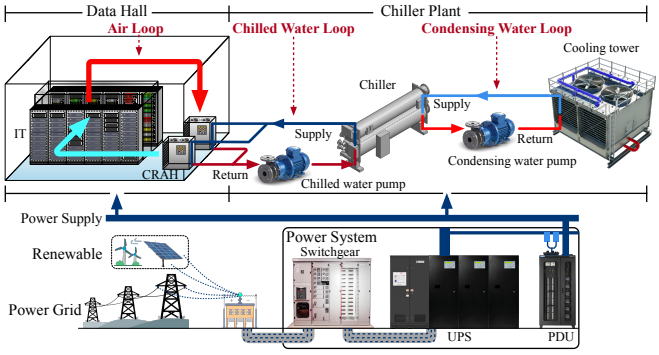


Fig. 1. Illustration of the architecture of a typical DC, which consists of the IT system, the cooling system, and the electrical power supply system.

Accurate DC carbon footprint characterization is challenging due to the complicated architecture of a modern DC that consists of multiple interrelated subsystems. A typical DC comprises three major systems, i.e., the information technology (IT) system, the cooling system, and the electrical power supply system, as shown in fig. 1. The IT system undertakes the computing workloads, the major energy demand source. The IT workload can be roughly categorized into two classes, i.e., the real-time services workload and the batch workload.

This research is supported by the RIE2025 Industry Alignment Fund – Industry Collaboration Projects (IAF-ICP) (Award I2301E0026), administered by A*STAR, as well as supported by Alibaba Group and NTU Singapore. This work was also supported by the National Natural Science Foundation of China (No. 62262026) and the Jiangxi Natural Science Foundation (No. 20232BAB202020).

[1]Carbon intensity is the carbon emissions embodied in unit electricity generation. The unit is $kgCO_2/kWh$.

TABLE I
SUMMARY OF NOTATIONS.

| Sym. | Definition | Sym. | Definition |
|---|---|---|---|
| $D[k]$ | aggregated computing demand at the $k$-th period | $N_{\text{VM}}[k]$ | number of incoming tasks at the $k$-th period |
| $l_i$ | duration of the $i$-th task | $d_i$ | the CPU demand of the $i$-th task |
| $P_i[k]$ | power consumption of the $i$-th server at the $k$-th period | $T_z[k]$ | zone average temperature at the $k$-th period |
| $T^i_{\text{sup}}[k]$ | the $i$-th CRAH unit's supply temperature | $m^i_{\text{sup}}[k]$ | supply air mass flow of the $i$-th CRAH unit |
| $Q[k]$ | total data hall heat load at the $k$-th period | $m_{\text{coil}}[k]$ | cooling coil water mass flow rate at the $k$-th period |
| $Q^i_{\text{coil}}[k]$ | cooling load of the $i$-th cooling coil | $Q^i_{\text{chiller}}$ | cooling load of the $i$-th chiller |
| $f_*$ | parametric device power consumption model | $\rho[k]$ | carbon intensity of the electrical grid at the $k$-th period |
| $c[k]$ | average cluster CPU utilization at the $k$-th period | $r[k]$ | renewable energy generation at the $k$-th period |
| $u[k]$ | capacity provisioning decision at the $k$-th period | $\mathbf{e}[k]$ | collection of the inputs to the DC physical system |
| $\psi$ | workload shifting ratio | $\sigma$ | maximum allowable consecutive capacity change |
| $T$ | capacity provisioning period | $\tau$ | real-time scheduling period |
| $C^a_p$ | specific heat of air | $C^w_p$ | specific heat of water |

The batch workloads, e.g., machine learning model training, and big data analytics, are usually *delay-tolerant*, indicating that they can be scheduled to run at the proper time as long as they can finish before their *deadlines*. The cooling system is designed to extract heat generated in the data hall. For a typical air-cooling DC, the computing room air handlers (CRAHs) supply cold air to the IT system inlet and draw hot air from the IT system outlet. Outside the data hall, there is a chiller plant that supplies chilled water to the cooling coils of the CRAHs. The heat extracted from the data hall is removed by two fluid loops, i.e., *chilled water loop* and *condensing water loop*. The electrical power supply system delivers electrical power to the IT equipment and cooling system of a DC. It integrates multiple power supply sources including grid power, on-site renewable energy generators, etc., through a microgrid. Specifically, the utility grid is connected to the uninterrupted power supply (UPS) which connects the power distribution units (PDUs). Each PDU supplies power to several server racks and cooling devices.

As electricity-related carbon emissions dominate a DC's carbon footprint [4], integrating low-carbon renewable energy into its energy supply is essential for DC decarbonization. Renewable energy can reduce the energy consumption of a DC from the electrical grid, which currently still has higher carbon intensity. However, renewable energy, e.g., solar energy, is often intermittent and uncertain. For example, its supply might not be sufficient during the energy demand peak period. Therefore, uncertainty-aware DC capacity provisioning is important to improve renewable energy utilization and reduce DC carbon footprint. To achieve this, we need to address two main challenges. The first is to holistically capture the DC carbon footprint. The second is to develop an uncertainty-aware framework that is robust to these uncertainties.

To address the first challenge, a common industrial practice is using a fixed Power Usage Effectiveness (PUE) [6], which is the ratio between total DC energy consumption and the total IT energy consumption, to estimate the DC energy consumption from the IT energy consumption. However, operational PUE is hard to estimate and time-varying. Several existing works adopt simplified models (e.g., piece-wise linear model) to estimate DC energy consumption [5]. Though these models provide intuitive insights into the DC energy footprint, they do not consider the detailed DC energy consumption. They may generate poor extrapolation performance when unseen data is not covered during the model fitting process. Some latest works adopt sophisticated physics-based energy simulators (e.g., EnergyPlus [7]) to evaluate DC energy footprint. These models precisely characterize the physical process and energy footprint of a DC. However, building such a model is difficult as it requires detailed information about a DC. Furthermore, these black-box simulators are not differentiable, preventing them from being integrated into an efficient optimization framework.

To address the second challenge, existing works can be categorized into two classes: a) online capacity provisioning without predicting uncertain parameters, and b) predictive capacity provisioning. In [8], [9], the authors adopt the Lyapunov optimization framework [10] to dynamically adjust the DC capacity provisioning with the time-varying renewable energy availability or electricity price. These studies do not require knowledge about uncertain system information. However, their convergence speed is not satisfactory (requiring thousands of iterations), hindering their practical implementation. Other studies leverage the predicted renewable energy availability and computing demand to implement workload management. In [11], the authors implement GreenSlot, a renewable-aware job scheduler that schedules workload with predicted electricity price and renewable energy generation in a small-scale cluster. However, it only considers the computing system's energy usage. Another study proposes a renewable and cooling-aware DC workload management framework with renewable energy and IT workload forecasting [12]. It considers the DC holistic energy footprint with simplified DC facility energy models. However, it does not consider the forecasting uncertainties. Recently, Google proposed a risk-aware DC capacity provisioning framework that considers errors in IT workload forecasting [5]. However, the capacity provisioning framework is not adaptive, as it is not designed to be responsive to prediction errors.

To bridge these gaps, we propose to build the DC *digital twin* and integrate it into an adaptive DC capacity provisioning framework for DC decarbonization. A DC digital twin is

TABLE II
SUMMARY OF EXISTING WORKS ON DC ENERGY MODELING AND CARBON-AWARE WORKLOAD MANAGEMENT

| | Uncertainty Consideration | | | DC Energy Model | Capacity Provisioning Methodology | IT Workload SLO Consideration |
|---|---|---|---|---|---|---|
| | IT Workload Resource Request | IT Workload Runtime | Renewable Energy Availability | | | |
| [11] | ✗ | ✗ | ✓ | N.A. | Greedy heuristic | Deadline |
| [13] | ✓ | ✗ | ✓ | N.A. | Mixed Integer Linear Programming (MILP) | N.A. |
| [8], [9] | ✗ | ✗ | ✗ | Fixed PUE | Lyapunov optimization | QoS |
| [12] | ✓ | ✗ | ✓ | Simplified empirical facility model | Convex optimization | QoS & deadline |
| [5] | ✓ | ✗ | N.A. | Piecewise linear model | Convex Optimization | N.A. |
| [14] | ✗ | ✗ | ✗ | Simplified empirical facility model | Deep Reinforcement Learning (DRL) | Deadline |
| [15] | ✗ | ✗ | ✗ | N.A. | Convex Optimization | Flexible Workload Ratio |
| **Ours** | ✓ | ✓ | ✓ | Physics-informed digital twin | Model Predictive Control | Deadline |

the digital counterpart of a physical DC that holistically models its dynamics [16]. Unlike traditional physics-based simulations, a digital twin can continuously update itself and adapt to the latest DC system state via a *learning process*. To build the digital twin, we adopt a *physics-informed* approach. Specifically, we build a collection of *learnable* DC facility models and follow the DC heat transfer process to establish the digital twin. We can simulate the detailed DC thermodynamics and carbon footprint with the digital twin. Furthermore, we implement the system as a *differentiable* one, making it feasible to be plugged into an effective optimization framework. It also continuously collects online data from the physical system and updates its internal parameters accordingly. To address the risk-aware issue, we integrate the digital twin into an adaptive capacity provisioning framework based on model predictive control (MPC) [17]. The framework is adaptive as it continuously monitors the system state and updates its estimation of uncertain variables. At each capacity provisioning round, an optimization problem is formulated and efficiently solved with a gradient-based optimization method thanks to the *differentiable* property of the digital twin. We evaluate our solution with the operational data collected from a large-scale production DC. The evaluation results show that our digital twin can accurately predict holistic energy usage of the physical DC with around 3% relative mean absolute error (RMAE). In addition, our adaptive capacity provisioning can reduce around 27% carbon emissions compared with existing proposals while incurring negligible service quality degradation of the DC.

Our contributions are listed in the following.

- We develop the *physics-informed digital twin* for a real-world production DC. The digital twin system is based on the physical laws governing the DC physical processes. It encapsulates a learning process and keeps track of the latest state of a physical DC. Furthermore, as we inject physics knowledge into the digital twin, the models in the digital twin system are lightweight and the additional overhead is negligible.
- With the digital twin system, we develop an adaptive capacity provisioning framework based on shrinking horizon MPC. It keeps monitoring the status of the IT system and updates the estimation of future workload computing demand. Optimization is solved to generate the capacity provisioning decision at each provisioning period.
- We evaluate our proposal with the operational data collected from a *real* large-scale co-located DC. We show that our physics-informed digital twin yields less than 5% RMAE compared with the real operation data. We use the *real* DC workload traces to evaluate our adaptive capacity provisioning framework, which approaches the offline optimal solution with perfect knowledge about all uncertainties. We show that an additional 27% carbon footprint is available compared to the current industrial practice without capacity provisioning. Furthermore, we conduct two ablation studies to explore the importance of adaptive computing demand re-estimation and demonstrate the robustness of our design against large forecasting errors.

*Paper Roadmap*: In §II, we present the related work. System architecture overview will be introduced in §III. Subsequently, we present the system models in §IV. Problem formulation and the adaptive capacity provisioning framework will be introduced in §V. Evaluation results can be found in §VI. In what follows, we present a discussion on three future directions that can be explored based on the proposed framework §VII. Lastly, conclusions are drawn in §VIII.

## II. RELATED WORK

In this section, we present existing works on DC energy modeling and carbon-aware DC workload management. We summarize the literature as shown in table II.

### A. DC Energy Modeling

DC energy modeling is the foundation for carbon-aware DC management. Most existing work focuses on the energy modeling of computing devices such as servers and networking devices. Among these models, the linear model [8] [9] and the piece-wise linear models [5] are widely adopted due to

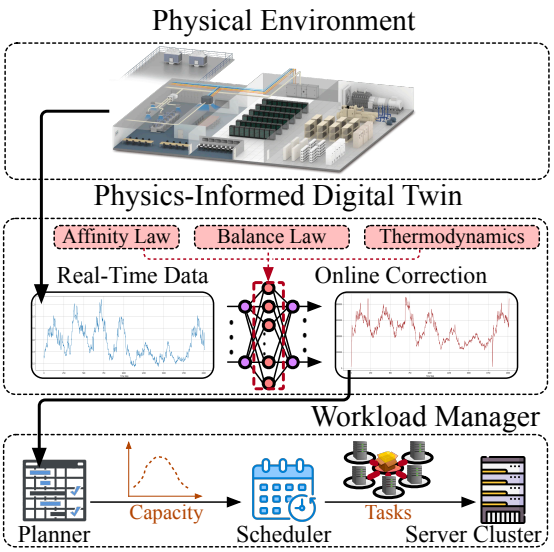


Fig. 2. Architectural overview of the proposed system. The physics-informed digital twin collects online data and updates its parameters with online data. It is integrated into the capacity planner to generate an hourly DC capacity provisioning plan. The real-time scheduler works in a finer time interval and schedules workloads to physical servers under hourly capacity provisioning. The system works hierarchically and follows the modularized design principle.

their simplicity. Some works also consider the impact of inlet temperature on server energy consumption [18] [19]. Although these modeling approaches are validated against experimental data, they only capture the energy consumption from the computing system. To synergistically analyze a DC's energy footprint, some proposals adopt the physics-based energy simulator EnergyPlus [7] to model the holistic DC infrastructure [18] [19] [20]. Although these simulators provide detailed models for the cooling system and the electrical equipment of a DC, they do not support rollout and cannot be integrated into an optimization framework. To address it, some studies propose to develop simplified models for DC facilities [12]. Some other studies, e.g., [21], adopt data-driven approaches, but their models' extrapolation capability is unsatisfactory. The historical data collected from a DC running at a stable operating point are often non-exploratory and centered around a target setpoint under conventional feedback control. The model learned with such data tends to be overfitted and poorly extrapolated to unseen states. Recently, Wang et al. proposed a physics-informed approach to model the DC energy consumption [20]. Such physics-based models capture the governing physical process inside a DC enclosure and have better extrapolation capability than previous data-driven approaches. However, their modeling tailors to a small DC scenario without scalability validation. In this study, we develop a physics-informed approach for *holistic* DC energy modeling and validate our model on a large-scale production DC trace.

### B. Carbon-aware Workload Management

Effective workload management can significantly reduce the energy consumption and carbon emissions of cloud data centers. For example, cloud operators can judiciously manage the VM selection, migration, and consolidation to reduce energy footprint [22]–[26]. However, as the compute demand is escalating, improving the workload energy efficiency alone is not sufficient [4]. Recently, carbon-aware or renewable energy-aware workload management has emerged as a new option for decarbonizing data centers. Due to the high variability of grid carbon intensity, renewable energy availability, and electricity price, the flexibility of DC workloads has been explored to decarbonize both the DC itself [8] [5] [12] [27] and the power grid [28] or reduce the monetary cost [29]. In this section, we focus on the existing literature on carbon-aware workload management. Existing studies on this topic can be categorized into two classes: a) reactive workload management and b) predictive workload management.

*Online* algorithms have been proposed to address the uncertainty of future workload and renewable availability [8], [9], [30]. These algorithms do not require knowledge about future system information, e.g., renewable energy availability, grid carbon intensity, or electricity price. However, they need thousands of interactions with the system to converge, making them infeasible in real-world deployment. GreenDRL [14] is a DC manager based on Deep Reinforcement Learning (DRL). Like the online algorithms, GreenDRL does not require the prediction of uncertain variables. However, the DRL agent requires significant exploratory data to learn an optimal policy. Furthermore, its learning process is not explainable and also lacks SLO satisfaction guarantees.

Another group of works considers *predictive* workload management. In some recent empirical studies [15], [31], the DC capacity provisioning problem is addressed by simplifying or ignoring the uncertainty consideration. They assume perfect knowledge about the uncertainties and aim to demonstrate the effectiveness of carbon-aware DC capacity provisioning. Some studies consider more realistic scenarios. GreenSlot [11] is a parallel batch job scheduler that uses the predicted renewable energy availability to schedule workload with a greedy heuristic to minimize the DC's electricity cost. However, it only considers the energy usage of the computing system. Similar to [11], in the study [13], a mixed integer linear programming (MILP) is formulated for carbon-aware capacity provisioning without consideration of the holistic DC energy usage. Liu et al. [12] present a convex optimization framework that uses the predicted workload and renewable energy availability to generate a fixed capacity provisioning plan for the next day. Their work considers the energy consumption from the cooling facility with a simplified empirical model. However, their study does not consider the uncertainty in the predictions of renewable energy and workload computing demand. In addition, their plan is a time-based job scheduling, which may raise scalability concerns when their solution needs to handle massive job requests in modern DCs. Recently, Google announced its Carbon Intelligent Computing System (CICS), a risk-aware optimization framework for carbon-aware computing [5]. CICS adopts a modularized design by decoupling the capacity planner with the real-time job scheduler. The predicted workload computing demands and the grid carbon intensity are utilized to generate the capacity provisioning plan for the next day. To consider the risk, it uses the historical

prediction error to obtain an over-provisioning ratio, and over-provisions computing resources accordingly. Although CICS considers the forecasting error, its capacity plan is still fixed for the next day, making it infeasible to handle the unexpected workload surge. Furthermore, although some existing works consider the workload deadline [12], [32], they assume that the workload runtime is known, which is impractical. In this study, we further relax this assumption by considering the scenario where only a noisy run estimation can be obtained for each task. Our design is effective and robust in this realistic scenario with extensive evaluation presented in VI.

## III. System Architecture Overview

In this section, we introduce the proposed system's workflow and overall architecture, as illustrated in fig. 2.a We conduct a holistic modeling of a DC, including the computing system and the facility to evaluate its carbon footprint accurately. The system consists of a workload manager and a collection of physics-informed models to calculate its total energy consumption under dynamic workload management. In the rest of this section, we will present the workflow of these two sub-systems.

### A. Workload Manager

The workload manager consists of two modules: a) the capacity planner and b) the real-time workload scheduler. In this study, we pursue a *modularized* design by decoupling the capacity planner and the real-time scheduler to make our design scalable and flexible.

The capacity planner takes charge of adjusting the capacity provisioning under time-varying renewable energy generation to minimize the total DC carbon footprint. It works at a coarse-grained time slot of length $T$, e.g. $T = 1$ hour. At the beginning of each coarse-grained time slot, the capacity planner generates the capacity provisioning budget (in terms of total CPU cores) for the next period according to the system states and the forecasting result.

The real-time workload scheduler selects a candidate task in the task queue, places it on a proper physical machine, and allocates a certain amount of computing resources to the task. It works at a fine-grained time slot of length $\tau$, e.g., $\tau = 5$ minutes. At the beginning of each fine-grained time slot, it receives a bunch of workloads and pushes them into the task queue. We consider a more realistic scenario where the runtime of a task cannot be known exactly, but a noisy estimation is available. Specifically, the scheduler estimates the runtime of each workload through the estimator presented in §IV-D1. Subsequently, it will update the *aggregated computing demand*, as introduced in §IV-A2. In what follows, the earliest due first (EDF) scheduler dispatches the tasks to the servers according to their service level objective (SLO) due time. We use SLO and deadline interchangeably in this paper. If the scheduler finds that all computing resources at this period have been strained, it stops scheduling new tasks. At the end of each fine-grained time slot, it traverses each server and updates its record of the available resources on the server. If a task is completed, the scheduler recycles the allocated computing resources for the task and the available resources on the server will be updated accordingly. Otherwise, the scheduler updates the runtime estimation to this task via the re-sampling technique introduced in §V-B. The purpose of runtime re-estimation is to reduce the uncertainty of the future aggregated computing workload estimation. With more accurate computing demand forecasting, the capacity planner avoids under-provisioning.

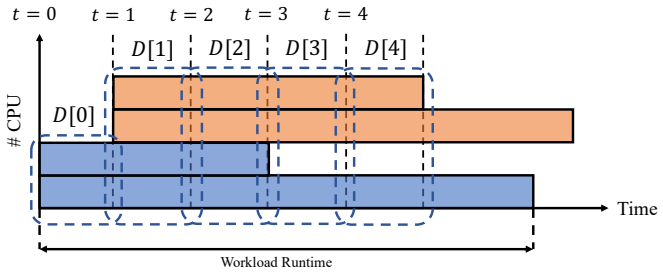


Fig. 3. Illustration of the aggregated computing demand. It will be affected by both the workload computing resource request and its runtime.

### B. Physics-Informed Digital Twin

In our system, a set of DC facility models are built via a *physics-informed* approach. These models can simulate the physical process inside a DC and calculate its carbon footprint. In addition, we feed the latest operational data of a physical DC to the models and keep them synchronized with the physical DC. These models also continuously update themselves with the latest operation states. Thus, we refer to the ensemble of these models as the *digital twin* of a physical DC. We organize the workflow of these models by resembling the physical process in a DC, which will be presented in §IV-B. The physical quantities of the DC facilities are derived according to the first principle laws, while the device performance models are *learned* with online collected data from the physical DC. Physical laws are also applied to simplify the device performance model so that our system can learn the dynamics of a physical DC using the DC operational data with a narrow dynamic range (see §VI-C). Furthermore, we implement our digital twin using Pytorch [33] to make it *differentiable* with respect to its inputs and internal parameters. Hence, it enables us to efficiently optimize the DC carbon footprint with gradient-based optimization techniques.

## IV. System Models

This section presents our system modeling, including the computing system models, the physics-informed digital twin, the carbon footprint model, and the forecaster models.

### A. IT System Models

In this section, we introduce our modeling of the IT workload, the aggregated computing demand as well as the servers in the DC.

*1) Workload Models:* We consider the virtual machine (VM) workload in this paper. At the fine-grained time slot $k$, there are $N_{\text{VM}}[k]$ requests coming to the system. Each workload specifies its CPU core demand and its deadline.[2] Each workload runs for a certain period. In this study, we do not assume that the workload runtime is precisely known in advance. Instead, we only obtain a *noisy estimation* of the runtime for each incoming workload using the runtime estimator presented in §IV-D2. In this paper, we assume the workload is running in a *non-preemptive* manner, indicating that it continuously runs before its termination with a fixed amount of resources. If a workload misses its deadline, the scheduler will either suspend it if it is running or remove it from the task queue.

*2) Aggregated Computing Demand Modeling:* As the capacity planner takes charge of the high-level capacity provisioning, it only considers the *aggregated computing demand*. At the $k$-th time slot, it is defined as:

$$D[k] = \sum_{i=1}^{N_{\text{VM}}[k]} \sum_{k \in [k_i^s, k_i^s + l_i]} d_i, \tag{1}$$

where $N_{\text{VM}}[k]$ is the number of incoming tasks at the $k$-th fine-grained time slot, $k_i^s$ is the submission time slot index of the task, $l_i$ is the runtime of the task and $d_i$ is the resource demand of the task. The aggregated computing demand at the $k$-th time slot is related to both the resource request of the incoming workloads and that from the previously arrived workloads as shown in fig. 3.

*3) Server Models:* We consider a DC with $N$ servers.[3] Each server has a total number of CPU cores $c_i$. The $i$-th server tracks the following two variables at each fine-grained time slot $k$: a) the number of occupied cores $c_i^o[k]$ and b) the utilization $u_i[k] = c_i^o[k]/c_i$. The scheduler will update these variables for each server accordingly when a new task is allocated to a server or a running task in the server is accomplished. In this paper, we focus on the power modeling of the aggregated computing demand similar to the previous work [5], [12]. The linear model is utilized to estimate the *total* IT system power consumption under a given utilization level as it was shown to be accurate in cluster-level power modeling for a large-scale server cluster. Specifically, the power consumption of the IT system is established as $P_{\text{IT}}[k] = P_{\text{static}} + (P_{\text{full}} - P_{\text{static}}) \cdot u[k]$, where $u[k]$ is the cluster *average* CPU utilization, $P_{\text{static}}$ is the static power consumption of the cluster, and the $P_{\text{full}}$ is the rated cluster power consumption when it is fully loaded. The average CPU utilization $u[k]$ is defined as $c[k] = \frac{1}{N \times c_i} \sum_{i=1}^{N} c_i^o[k]$.

## B. Physics-Informed Digital Twin

This section presents the physics-informed holistic modeling approach for DC facilities. We mainly consider the energy consumption from the cooling facilities and account for all electrical loss and the lightning power with a learnable coefficient. The high-level workflow of the model is illustrated in fig. 4.

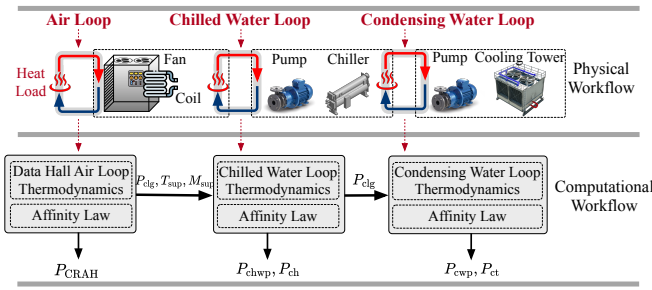


Fig. 4. Illustration of the physics-informed DC facility models. It adopts a stage-by-stage process, inspired by the heat transfer process.

*1) Data Hall Air loop Thermodynamics Model:* Air loop thermodynamics characterizes the heat transfer process inside the data hall. In this study, we consider an air-cooling DC and adopt a nodal model for the heat transfer process. We assume that the data hall temperature is uniformly distributed. Thus, considering the average room temperature is sufficient [7]. Let $N_{\text{CRAH}}$ denote the number of active CRAH units inside a data hall, the supply air temperature and mass flow rate of the $i$-th CRAH unit at time $t$ be $T_{\text{sup}}(t)$ and $m_{\text{sup}}(t)$, respectively. We further consider the UPS electrical efficiency and model the total heat load in the data hall as $Q(t) = P_{\text{IT}}(t)/\alpha$, where $\alpha$ is the UPS electrical efficiency. We continuously update $\alpha$ to be its moving average over the online data. The thermodynamics can be modeled with the following ordinary differential equation (ODE) [34]: $\frac{dT_z}{dt} = \sum_{i=1}^{N_{\text{CRAH}}} C_p^a \cdot m_{\text{sup}}^i(t) \cdot [T_{\text{sup}}^i(t) - T_z(t)] + Q(t)$, where $C_p^a$ is the specific heat of air and $T_z(t)$ is the zone temperature at time $t$. In this study, we consider a discrete-time model:

$$T_z[k+1] = \sum_{i=1}^{N_{\text{CRAH}}} C_p^a \cdot m_{\text{sup}}^i(k) \cdot [T_{\text{sup}}^i[k] - T_z[k]] + Q[k]. \tag{2}$$

Given the data hall heat load and the operational setting of each CRAH unit, we can predict the average room temperature at the next time slot.

*2) Chiller Plant Thermodynamics:* A chiller plant consists of the chilled water loop and condenser water loop as shown in fig. 4. Both loops have the supply-side half-loop and the demand-side half-loop.

The chilled water demand-side half-loop connects the cooling coil of each CRAH unit and the chillers. We should determine the heat load of each cooling coil and its chilled water flow rate. We adopt the uniform load setting to distribute the total cooling load at the $k$-th time slot $Q[k]$ according to our observation from the historical operational data. For the cooling coil of the $i$-th CRAH unit, its cooling load is $Q_{\text{coil}}^i[k] = Q[k]/N_{\text{CRAH}}$. The cooling coil transfers the allocated heat load to the chilled water. To calculate the required chilled water mass flow rate to meet the cooling load, we adopt the NTU-effectiveness approach [35] to model the heat transfer process. Specifically, we choose the chilled water

[2]Although our approach can handle a multi-dimensional resource model, we restrict our focus on CPU-bounded workloads in this paper

[3]Our modeling approach can also be extended to the heterogeneous server configuration.

flow rate as $m^i_{\text{coil}}[k] = \frac{C^a_p}{C^w_p} \cdot m^i_{\text{sup}}[k]$, where $C^w_p$ is the specific heat of water. Under this setting, the heat transfer effectiveness between air and chilled water is 1. Hence, the total chilled water mass flow rate is $m_{\text{chw}}[k] = \sum_{i=1}^{N_{\text{CRAH}}} m^i_{\text{coil}}[k]$.

The chilled water supply-side half-loop contains chillers to provide chilled water to the CRAH units and a mechanical pump to cycle the chilled water around the chilled water loop. We consider there exist $N_{\text{ch}}$ chillers. The total cooling load of the chillers is equal to the total heat load in the data hall. We adopt the uniform load distribution mechanism to distribute the total cooling load to each chiller. The cooling load of the $i$-th chiller is $Q^i_{\text{chiller}}[k] = Q[k]/N_{\text{ch}}$. The part load ratio (PLR) of the $i$-th chiller is defined as $r^i_{\text{chiller}}[k] = Q^i_{\text{chiller}}[k]/C^i_{\text{chiller}}$ where $C^i_{\text{chiller}}$ is the cooling capacity of the $i$-th chiller. The mass flow rate of the pump is the total chilled water mass flow rate of the demand half-loop $m_{\text{chw}}[k]$.

The condenser water loop connects the chillers and the cooling towers through the fluid loop between them. Its architecture is similar to the chilled water loop. The heat load of a chiller will be transferred to the condenser water via its condenser and then exhausted to the environment via the cooling towers. The condenser water pump and cooling towers usually operate at their rated speed. Hence, we set the cooling towers and the condenser water pump to operate at their rated setting according to the specifications.

*3) Facility Power Consumption Models:* After we compute the thermal properties of each cooling device, we calculate its power consumption according to its operating condition. The power usage of each component at the $k$-th period is modeled by $P_{\text{CRAH}}[k] = \sum_{i=1}^{N_{\text{CRAH}}} f^i_{\text{CRAH}}(m_{\text{sup}}[k])$, $P_{\text{chwp}}[k] = f_{\text{chwp}}(m_{\text{chwp}}[k])$, $P_{\text{ch}}[k] = \sum_{i=1}^{N_{\text{ch}}} f^i_{\text{ch}}(T_{\text{chw}}[k], T_{\text{cw}}[k], r^i_{\text{ch}}[k])$, $P_{\text{cwp}}[k] = f_{\text{cwp}}(m_{\text{cwp}}[k])$ and $P_{\text{ct}}[k] = \sum_{i=1}^{N_{\text{ct}}} f^i_{\text{ct}}(m^i_{\text{ct}}[k], T_o[k])$. $T_{\text{chw}}[k]$, $T_{\text{cw}}[k]$ and $T_o[k]$ represent the chilled water supply temperature, condenser water supply temperature and outdoor wet-bulb temperature at the $k$-th period, respectively. The CRAH units and pumps' power consumption is modeled with *cubic* polynomials according to the affinity laws [36]. We fix the chilled water supply temperature $T_{\text{chw}}[k]$, and condenser water supply temperature $T_o[k]$ as per the specifications. As the chiller power consumption mainly comes from the mechanical compressor which also obeys the affinity law, we use *cubic* polynomial to calculate its power usage in terms of its cooling load. For the condenser water pumps and cooling towers, we use their design power usage from the specifications.

*4) Summary:* This section summarizes the key features of the physics-informed digital twin. Firstly, we implement all device models and thermodynamics via Pytorch [33] to obtain gradient information with auto-differentiation. For example, the computing system power usage is related to the provisioned capacity. The heat load is related to the computing system's power usage, indicating that the chiller plant's cooling load is also related to the computing system's power usage, so as the chillers' power usage. With the chain rule, we can obtain the derivative of the chillers' power usage with respect to the resource consumption. Similar analysis can be applied to other physical devices to establish the derivative of their power usage with respect to resource consumption. Second, our design is modularized. More sophisticated device models can be smoothly integrated into our system if they are differentiable. This makes our design flexible and scalable to large-scale DCs.

### C. DC Carbon Footprint Model

In this study, we focus on electricity-rated operational carbon emissions. The total energy usage consists of that from the IT system and the facility. The power consumption of the cooling system comes from the CRAH units, the chillers, the pumps, and the cooling towers. It is a function related to the data hall heat load. We define the collection of all control inputs and weather conditions at the $k$-th period as $\mathbf{e}[k] = [\mathbf{T}_{\text{sup}}[k], \mathbf{m}_{\text{sup}}[k], T_{\text{chw}}[k], T_{\text{cw}}[k], T_o[k]]$. The control inputs to the facility are obtained with the policy described in §VI.[4] We denote the DC cooling facility power usage as $p_{\text{clg}}(Q[k]; \mathbf{e}[k], \theta)$ where $\theta$ is the set of all *learnable* parameters in the digital twin. The total DC power is defined as $P_{\text{dc}}(u[k]; \mathbf{e}[k], \theta) = P_{\text{IT}}(u[k]) + p_{\text{clg}}(Q(u[k]); \mathbf{e}[k], \theta)$, where $u[k]$ is the capacity provisioning plan at the $k$-th period. We denote the renewable supply at the $k$-th time slot as $r[k]$. The total DC operational carbon footprint is modeled as $c[k] = \rho[k] \cdot \Big[P_{\text{dc}}(u[k]; \mathbf{e}[k], \theta) - r[k]\Big]_+$ where $[\cdot]_+$ denotes $\max(0, \cdot)$ and $\rho[k]$ represents the grid carbon intensity at the $k$-th period.

### D. Forecasting Models

Our predictive capacity provisioning relies on knowledge about future computing workloads and renewable energy availability. In this section, we introduce the prediction models.

*1) IT Workload Predictor:* In this study, we obtain the aggregated computing demand forecasting via the state-of-the-art generative cloud workload model [37]. It adopts a three-stage workload generation process. As the computing workload has a strong periodical pattern, the temporal information is used to generate the forecasting. In particular, we use the hour-of-day and day-of-week as our temporal features and adopt one-hot encoding to these features. They are sent to the first-stage *Poisson arrival model* to obtain the number of arrival workloads at a specific time slot. Then, the output of the Poisson arrival model is combined with the temporal features to the *resource model* to predict the resource request of each workload with a long short-term memory (LSTM) model [38]. Finally, the temporal features and the output of the resource model are sent to the *duration model*, which is also an LSTM model, to obtain the runtime estimation of each workload. Using the generated workload trace, we can calculate the aggregated computing demand from the future IT workload with eq. (1).

*2) Workload Runtime Predictor:* In this paper, we reuse the duration LSTM in the workload predictor as our workload runtime predictor. Different from the one used in workload prediction, we use the *real* resource demand of each incoming task as the input to the duration LSTM to obtain the runtime

[4]Note that our digital twin supports arbitrary policies as we view controls as external inputs.

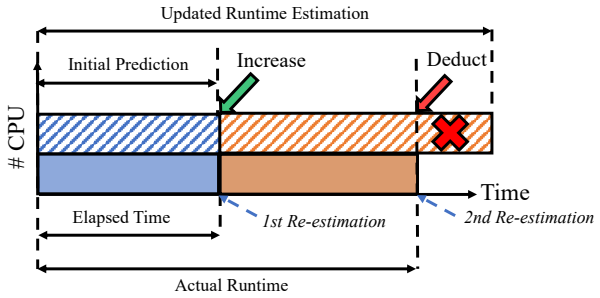


Fig. 5. Illustration of the workload runtime re-estimation process. At the first re-estimation, we calculate a new posterior runtime distribution and draw a new estimation accordingly. At the second re-estimation, we deduct the remaining computing demand from the future aggregated computing demand to avoid over-provisioning.

estimation for each task. The output of the duration LSTM is a probability vector $\mathbf{p}^i_{\text{dur}} \in \mathbb{R}^d$. Here, real-valued runtime is quantized into $d$ bins with the approach in [37]. We view $\mathbf{p}^i_{\text{dur}}$ as the prior knowledge about the runtime of the $i$-th task. The $k$-th element in $\mathbf{p}^i_{\text{dur}}$ is the probability that the $i$-th task's runtime will fall into the bin, i.e., $\Pr(l_i = k)$. To obtain the real-valued runtime, we first sample an index according to the probability vector $\mathbf{p}^i_{\text{dur}}$ of the $i$-the task. Then, we sample a real-valued runtime between the lower and upper bound represented by the index according to the uniform distribution.

*3) Solar Energy Predictor:* In this study, we consider solar energy as the green energy source and leverage the state-of-the-art time series forecasting model DeepAR [39] to forecast solar energy availability. It is an LSTM model working in an autoregressive manner. It consumes a sequence of historical features and outputs the prediction step-by-step. We use the autoregressive model since it is compatible with our shrinking horizon MPC design. Similar to previous studies [11], [12], we view the weather-related variables that directly affect solar energy production, e.g., solar radiance, outdoor dry-bulb temperature, and wind speed as covariates as they can be obtained from weather forecasting. In addition, since solar energy production has a diurnal pattern, we view the temporal features as covariates as well. Specifically, we use a cyclic encoding for the hour-of-day feature. Furthermore, we also augment the input feature with the recent $M$ hours of solar energy production to harness the short-term temporal correlation in the solar energy trace. We select $M = 6$ in our study based on cross-validation with historical data.

## V. Adaptive Capacity Provisioning via Shrinking Horizon Model Predictive Control

In this section, we introduce the proposed carbon-aware adaptive capacity provisioning solution. We first present the problem formulation. In what follows, we introduce the workload runtime re-estimation technique and the computing demand correction mechanism, which helps to improve the estimation accuracy of future computing demands. Based on these techniques, the shrinking horizon MPC reformulation is proposed to conduct the capacity provisioning.

### A. Problem Formulation

In this paper, we consider the carbon-aware DC capacity provisioning problem as shown in the following:

$$\underset{u[1],...,u[H]}{\text{minimize}} \quad \sum_{k=1}^{H} \rho[k] \cdot \Big[P_{\text{dc}}(u[k]; \mathbf{e}[k], \theta) - r[k]\Big]_+, \tag{3a}$$

$$\text{subject to} \quad \sum_{k=1}^{H} u[k] \geq \sum_{k=1}^{H} D[k], \tag{3b}$$

$$\psi \cdot D[k] \leq u[k] \leq C, \tag{3c}$$

$$\Big|u[k+1] - u[k]\Big| \leq \sigma. \tag{3d}$$

where $\psi$ is the load shifting ratio, and $C = N \cdot c_{\text{total}}$ is the total computing resource respectively. The objective is to minimize the DC's operational carbon footprint. The first constraint is the computing resource provisioning conservation constraint. It states that the provisioned computing resources in total $H$ periods should be greater than the total computing demand. This constraint ensures that the task queue backlog does not go to infinity [5], [40]. The second constraint indicates that the provisioned computing resources are bounded. The lower bound indicates the minimal computing demands should be satisfied in each period. Without the knowledge of the uncertain future computing workloads and renewable energy generation, this problem is intractable. In this paper, we use *prediction* to solve this problem via a shrinking horizon MPC. In addition, the problem formulation is quite general, and it is possible to replace the grid carbon intensity signal $\rho[k]$ with a varying electricity price and optimize the data center operation under the uncertain electricity price to minimize the operation cost. It is also interesting to explore joint cost and carbon footprint optimization under the proposed framework.

### B. Re-estimating Task Runtime for Improved Aggregated Computing Demand Forecasting

We design a runtime re-estimation mechanism to progressively obtain more and more accurate estimations of the future aggregated computing demand. From the definition of aggregated computing demand eq. (1), the runtime of existing tasks will affect the future aggregated computing demand. Since only a noisy estimation for the runtime can be obtained when a new task comes to the system, we propose to dynamically adjust our runtime estimation according to the *elapsed time* to improve future aggregated computing demand estimation, as illustrated in fig. 5.

The idea is to use the initial runtime distribution as a prior and derive the *posterior distribution* of the workload runtime given the elapsed time. When we find that the initial runtime prediction is under-estimated at the $m$-th fine-grained time slot (see the first re-estimation in fig. 5), we update our estimation via the posterior probability:

$$\Pr(l_i = k | l_i \geq m) = \frac{\Pr(l_i = k)}{1 - \sum_{n=1}^{m} \Pr(l_i = n)}. \tag{4}$$

Eq. (4) gives the posterior runtime distribution of the task given it has run for $m$ fine-grained time slots. We use it to sample a new runtime for the task. The future aggregated

computing demand is updated accordingly via eq. (1). Similarly, if our runtime estimation of the task is higher than its actual one (see the second re-estimation in fig. 5), we deduct the over-estimated part in the future aggregated computing demand accordingly to avoid over-provisioning.

### C. Shrinking Horizon MPC Reformulation

We first briefly describe the workflow of the shrinking horizon capacity planner at each capacity provisioning round. The first step is using prediction models to obtain the prediction of future system states. Then, an optimizer will be evoked to solve a shrinking-horizon optimization problem. Once solved, we obtain an optimized action trajectory that also satisfies the constraints. Lastly, the *first* action in the trajectory will be applied to the system to make the system robust to long-term forecasting errors. The planner observes the system state at the beginning of each fine-grained time step and then adjusts its prediction accordingly.

We consider the case where at the beginning of the $\tau$-th capacity provisioning round, only $H-\tau$ periods of noisy forecasting about the future aggregated computing demand $\hat{D}[\tau], \hat{D}[\tau+1], ..., \hat{D}[H]$ and renewable energy generation $\hat{r}[\tau], \hat{r}[\tau+1], ..., \hat{r}[H]$ are available. We define a variable $s[\tau] = \sum_{k=1}^{\tau-1} \left[u[k] - \widehat{D}[k]\right]$ as the *cumulative supply deficit* until the $\tau$-th capacity provisioning round. This variable captures how much computing demand is not satisfied in the previous $\tau-1$ capacity provisioning rounds. Our main idea is to compensate for the deficit in the future to ensure that the budget constraint in eq. (3b) is satisfied. Specifically, at the beginning of the $\tau$-th capacity provisioning round, we observe the system and obtain the updated cumulative supply deficit $s[\tau]$ and then solve the following optimization:

$$\underset{u[\tau],...,u[H]}{\text{minimize}} \quad \sum_{k=\tau}^{H} \rho[k] \cdot \left[P_{\text{dc}}(u[k]; \mathbf{e}[k], \theta) - \widehat{r}[k]\right]_+ \tag{5a}$$

$$\text{subject to} \quad s[\tau] + \sum_{k=\tau}^{H} u[k] \geq \sum_{k=\tau}^{H} \widehat{D}[k], \tag{5b}$$

$$\psi \cdot \widehat{D}[k] \leq u[k] \leq C, \tag{5c}$$

$$\left|u[k+1] - u[k]\right| \leq \sigma. \tag{5d}$$

As the DC energy model is *differentiable*, we solve it efficiently via the gradient-based optimization technique [41]. By following the standard MPC setting, we only apply the first optimized action $u^*[\tau]$. Note that we compensate for the cumulative supply deficit in the past $\tau-1$ periods by adding it to the predicted aggregated computing demand of future $K-\tau$ periods. With this treatment, we aim to make the cumulative supply deficit close to zero at the end of the last capacity provisioning period to satisfy the budget constraint eq. (3b).

## VI. Performance Evaluation

In this section, we first present the evaluation setup. Subsequently, we illustrate the digital twin prediction accuracy and compare it with the data-driven energy modeling approach. Next, we present the trade-off between carbon footprint and SLO violation, followed by the ablation studies.

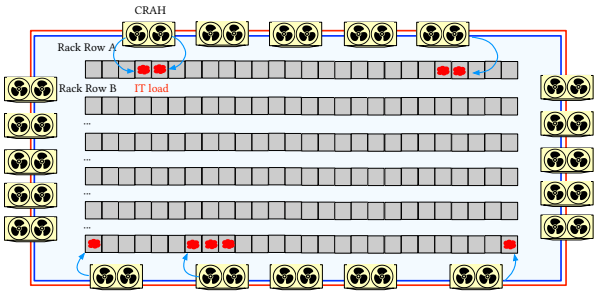


Fig. 6. Illustration of the floor plane of the evaluated DC. There are 16 running CRAH units inside the data hall. The rated IT capacity is 1.5 MW.

### A. Evaluation Setup

In this paper, we refer to the physical configuration of a real large-scale production DC to conduct our evaluation. The floor plan of the DC is illustrated in fig. 6. The DC has a rated IT power of 1.5 MW and its area is around 2, 500 $m^2$. There are 1, 250 16-core servers in the data hall, and the rated power of each server is 1 kW. There are 16 CRAH units in the data hall. The chiller plant with one chiller and one cooling tower produces chilled water for the data hall [5]. The DC is equipped with 2 MW photovoltaic (PV) arrays to provide solar energy. We use one month of data from the DC to build a detailed energy model using EnergyPlus [7]. We implement the PV generator model with EnergyPlus and use real weather data to drive the EnergyPlus simulation. Our physics-informed digital twin interacts with the EnergyPlus simulator to mimic the online data collection and learning process. We set the cooling controls according to the dynamic capacity provisioning result to maintain the data hall temperature around a target setpoint, i.e., 30 °C in operations. Specifically, we set the supply air temperature of each CRAH unit to 22 °C, and calculate the corresponding mass flow rate with eq. (2). The chilled water supply temperature is set to 16 °C. We train the workload predictor and the solar energy predictor using 21 days of historical trace and use the trained model in our *7-day* evaluation. We use publicly available cloud workload trace from Azure [42] in our evaluation. This large-scale workload trace contains over a million workloads and reflects the real-world DC computing system status. The trace contains the CPU and memory request for each workload, as well as the runtime. It is recorded every 5 minutes ($\tau = 5$ minute). The plan period $H$ is 24 hours and capacity provisioning is updated every hour ($T = 1$ hour). As the original trace does not specify the workload deadline, we set it by multiplying the runtime with a *slack ratio*. Similar to [43], we randomly sample the slack ratio for each workload from the interval $[2, 4]$ with a uniform distribution. For the grid carbon intensity, we use the statistics in Singapore, which is 0.40 $kgCO_2/kWh$ [44]. We set the load shifting ratio $\psi$ as $0.6C$ according to the statistics in [15], and the capacity switch threshold $\sigma$ as $0.15C$.

[5]We omit the detailed parameters due to the non-disclosure agreement.

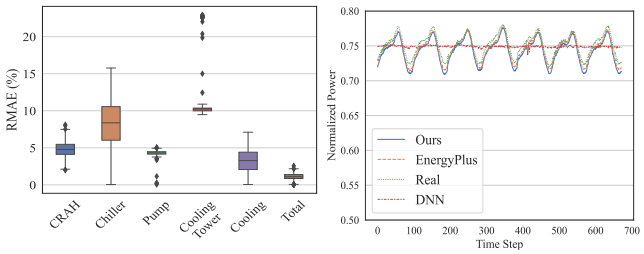


Fig. 7. Illustration of the prediction accuracy of the physics-informed digital twin and the data-driven holistic DC energy model. The prediction error for cooling system energy consumption is lower than 5%, and the holistic DC energy usage prediction error is around 3%. Nonetheless, the DNN-based energy model struggles with the relatively stable DC operation data.

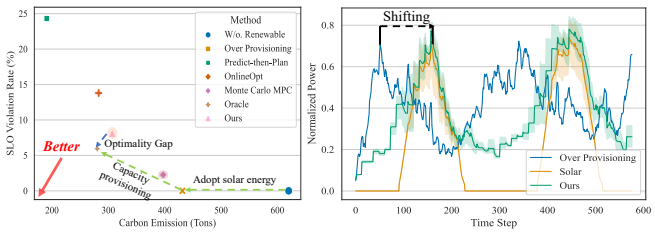


Fig. 8. Illustration of the trade-off between carbon emissions and SLO violation. The left subplot presents the performance of different capacity provisioning strategies, and the right subplot shows the DC power usage before and after provisioning. Our design approaches the performance of the Oracle scheme with perfect knowledge about all uncertainties. Compared with the Over-provisioning scheme, our approach has higher solar energy utilization.

### B. Benchmark Methods

- **Predict-then-Plan** [5], [12], [40]: This approach uses next-day renewable energy and IT workload forecasting to generate the next-day capacity provisioning plan and then follow them strictly.
- **OnlineOpt** [45]: This is a primal-dual optimization approach to solving the original problem with adjustable Lagrangian multipliers. It does not require the knowledge of the future uncertain variables and makes decisions based on the current system states.
- **Monte Carlo MPC** [46]: This approach uses Monte Carlo sampling to obtain a bunch of action trajectories, and then evaluate the objective function to find the best trajectory. We use a random walk-based sampling strategy to enforce the satisfaction of the capacity switching constraint eq. (3d). After sampling, we delete the trajectories that violate the budget constraint eq. (3b).
- **Oracle**: This approach assumes $r[k]$ and $D[k]$ in eq. (3) are known in advance. In addition, the runtime of each arrival workload is known precisely as well. With such perfect information, we directly solve eq. (3) to generate the capacity provisioning for the next day. The solution yields the best trade-off between carbon footprint and SLO satisfaction, as shown in fig. 8.
- **Over-provisioning**: This approach does not conduct capacity provisioning, and keeps it running at full capacity.
- **W/o. Renewable**: This approach does not involve any renewable energy supply and keeps the DC running at full capacity.

### C. Digital Twin Accuracy Validation

In this study, we validate the accuracy of our physics-informed digital twin using *real DC operational data* and demonstrate the superiority of the physics-informed digital twin compared with the widely adopted data-driven approach. The result is illustrated in fig. 7. The left subplot demonstrates the equipment-level energy usage prediction accuracy, which is measured with the relative mean absolute error (RMAE). The right subplot presents the prediction accuracy of different modeling approaches. We choose the Deep Neural Network (DNN), as the alternative DC energy model. Specifically, it is a fully connected network with 3 hidden layers, and each layer has 128, 64, and 32 neurons, respectively. We set the input features of the DNN model the same as those of the digital twin and the physics-based EnergyPlus model for fair comparison.

Firstly, our physics-based digital twin accurately predicts the DC energy usage under time-varying IT system power usage. The overall prediction error is less than 5%, which is acceptable according to the ASHRAE standard [47]. Hence, it can be integrated into the MPC planner by providing the gradient of the DC energy usage concerning the capacity provisioning decision.

Second, a calibrated EnergyPlus model also accurately predicts the DC energy usage, and the physics-informed digital twin yields similar accuracy. However, the EnergyPlus is a black-box simulator and it cannot provide gradient information. Thus, it cannot be integrated into the proposed MPC capacity planner. The digital twin not only leverages the governing physical laws to improve the prediction capability but also is compatible with the capacity planner as it is differentiable.

Lastly, the comparison between the physics-informed digital twin and the DNN model demonstrates the superiority of injecting physics laws into the DC energy model. From fig. 7, we can find that the IT power usage varies in a narrow range. It makes traditional data-driven modeling approaches fail as they require sufficient exploratory data to learn the mapping function. This situation is even worse when we consider a large-scale data center with many physics assets as their operating conditions are also the inputs to the energy model. Without high-quality exploratory operational data, a DNN model with a high dimensional input cannot be sufficiently trained, leading to high extrapolation error. By incorporating physics law into the learning process, the digital twin efficiently learns complicated DC physical dynamics with limited operation data. Furthermore, the digital twin is also more scalable compared with the DNN model as it follows a decentralized learning mechanism according to the physical process, where each physical asset learns its model with locally collected online data. However, the data-driven model needs to be updated when the physical system changes.

### D. Trade-off between Carbon Footprint and SLO Violation

This section explores the trade-off between carbon emissions and SLO violation with different capacity provisioning

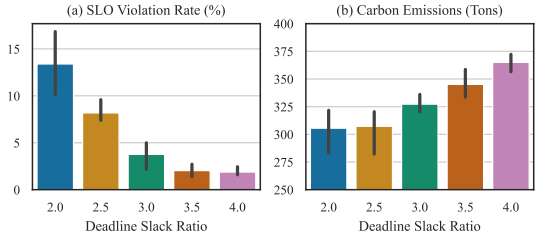


Fig. 9. Illustration of the impact of deadline slack ratio on the overall performance of our approach. Loose SLO constraints indicate a higher potential of temporal shifting for less carbon emissions.

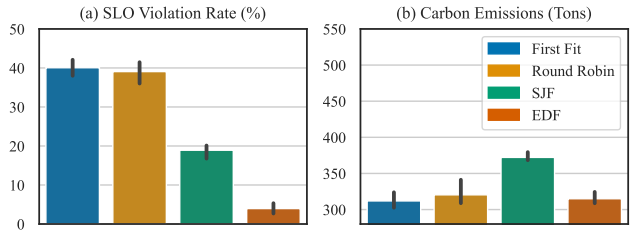


Fig. 10. Illustration of the impact of different task schedulers on the overall performance of our approach. SLO-aware workload scheduler outperforms other schedulers by a large margin.

strategies. Several conclusions can be drawn from Fig. fig. 8.

Our design approaches the performance of the oracle scheme with perfect knowledge about all uncertainties, outperforming the Predict-then-Plan approach significantly. It demonstrates the benefits of adaptive adjustment of capacity provisioning plan according to the system feedback. By refining our prediction about the future computing demand with the latest feedback, our approach mitigates the risk of underprovisioning DC capacity. In addition, the marginal gain from improving the prediction accuracy is also negligible. Hence, we believe the low-overhead LSTM predictors also achieve a good trade-off between prediction accuracy and computational complexity.

DC capacity provisioning benefits from prediction. With future system state prediction, the planner can *plan in advance* to better utilize renewable energy while avoiding underprovisioning. Though the traditional online algorithm without prediction is also adaptive, it can only respond to the system dynamics in a *passive* way. This takes a longer time to converge and is inferior to our approach in the short-horizon scenario.

The differentiable digital twin facilitates the MPC capacity planner, as shown in the comparison between the two MPC schemes. As the digital twin is differentiable, we can adopt an efficient gradient-based optimization method to solve the MPC problem at each time step. Whereas, the Monte Carlo MPC only evaluates the trajectory cost with the digital twin without considering the gradient information, making it less efficient.

Our approach achieves better renewable energy utilization compared to the Over-provisioning approach as shown in the right subplot in fig. 8. The over-provisioning approach always runs the DC at full capacity without considering the potential of temporal workload shifting to leverage renewable energy availability, although it achieves the lowest SLO violation rate. Our approach achieves an additional 27% carbon reduction by running more workloads when solar energy is sufficient without compromising SLO service quality (less than 5% violation). Furthermore, as many cloud workloads can tolerate long waiting (even up to 24 hours [48]), cloud data centers can significantly reduce their carbon footprint by adaptive capacity provisioning without significantly compromising service quality.

### *E. Impacts of Deadline Slack Ratio on SLO Satisfaction*

In this section, we evaluate our approach with different deadline slack ratios to explore its impacts on SLO violation similar to the analysis in [43]. The evaluation result is shown in fig. 9. We empirically find that when the deadline slack ratio is larger than four, the negative impacts of capacity provisioning on SLO satisfaction are negligible. It indicates that long-running tasks with loose SLO, e.g., AI model training tasks, suffer less under carbon-aware capacity provisioning. In addition, a diminishing benefit of loosening SLO is also observed. Regarding the lower carbon emissions with tight deadline slack ratio (see the right sub-figure in fig. 9), it is due to many tasks being unable to be executed as the scheduler detects that their SLO cannot be met by any mean.

### *F. Impacts of Different Task Schedulers on SLO Satisfaction*

In this section, we investigate the impacts of different real-time task schedulers. We compare our default EDF scheduler with three different widely adopted schedulers, i.e., the round-robin (RR) scheduler, the first-fit (FF) scheduler, and the shortest job first (SJF) scheduler. The evaluation result is shown in fig. 10. The schedulers with higher SLO violations have less carbon emissions due to the reasons discussed in the previous section.

Under time-varying capacity, different task schedulers have significantly different performance. When varying DC capacity, the duration-agnostic schedulers, i.e., the RR scheduler and the FF scheduler, incur large SLO violations. We empirically find that when using the two schedulers, many long-running jobs are scheduled to run when the scheduler detects that the current capacity is sufficient to support these workloads. However, they ignore the potential capacity reduction due to the less renewable energy availability in the future. Hence, the capacity is strained with the long-running workload and the schedulers will prevent the future workload from being scheduled to run, leading to SLO violations. Such observation also coincides with the findings in [49], showing the necessity of workload duration-aware scheduler design when the DC capacity is time-varying. The SJF scheduler works better because it considers the task runtime estimation. It prefers tasks with short runtime, which suffer less due to dynamical capacity provisioning. Nonetheless, it is still not deadline-aware, making it inferior to the default EDF scheduler.

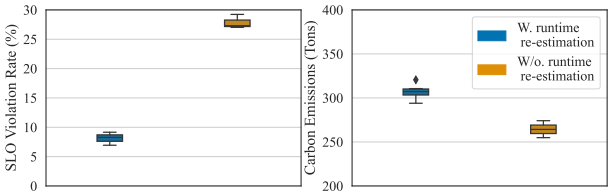


Fig. 11. Comparison between the MPC capacity planner with and without task runtime re-estimation. Performance degradation is severe without re-estimation, showing its necessity in the capacity planner.

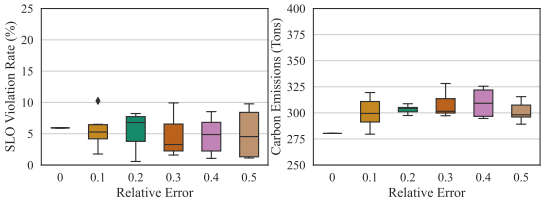


Fig. 12. Performance evaluation of our design under different forecasting error levels. "0" represents the results of the oracle scheme.

### G. Impacts of Adaptive Workload Runtime Re-estimation

This section evaluates our approach with and without runtime re-estimation, as shown in fig. 11. The effectiveness of the adaptive MPC framework relies on the accurate prediction of the future aggregated computing demand. Without the workload runtime re-estimation, the forecasting errors in the initial workload runtime estimation harm the performance of the MPC planner. With the re-estimation, we progressively obtain more accurate workload runtime estimation and the aggregated computing demand estimation. Therefore, the workload runtime re-estimation is critical to the effectiveness of the MPC capacity planner.

### H. Robustness Under Different Forecasting Error Levels

In this section, we evaluate the robustness of the proposed MPC planner under different controlled forecasting error levels. The results are presented in fig. 12. We add controlled noise to both the original IT workload trace and the solar energy trace. We find that our MPC planner is robust under a broad spectrum of forecasting errors and even works well with *very noisy* forecasting (see the results under 50% relative error). The robustness comes from both the runtime re-estimation and the reformulated shrinking-horizon MPC with the supply deficit compensation mechanism.

## VII. Discussions

We now discuss several issues not fully addressed in this paper and shed some light on the potential directions for resolving these issues and further extend this work.

### A. Heterogeneous Workload and Servers

This study focuses on managing the batch workload with a specific SLO. In our future work, we will relax this assumption by considering the capacity provisioning of the heterogeneous workloads, including the real-time service workload, long-running AI training workload, etc. In addition, our framework can also be extended to consider the penalty raised by the workload delay, similar to [40], which is left for future work. Furthermore, the assumption that each server has the same power model can also be relaxed by calibrating a parametric power model for each server or a group of servers with similar configurations if we have the server-level power data. Though such power data is difficult to obtain from the co-located data center operators, we plan to collect them on our on-premise data center testbed and use it to develop power models for different kinds of servers.

### B. Joint IT and Facility Optimization

In this paper, we develop the differentiable digital twin for evaluating the data center's holistic power consumption under dynamic capacity provisioning. Note that the digital twin also captures the sophisticated relationship between the facility power consumption and control inputs to the DC's physical system. It can also be used to optimize the DC workloads and physical infrastructure jointly. This topic has recently been explored under the framework of deep reinforcement learning (DRL) [50], which jointly performs job allocation and data hall CRAH unit control to reduce total DC energy consumption. However, the DRL algorithm requires extensive interaction with the physical system to learn the optimal policy. With the differentiable digital twin, we can add the cooling system control inputs, e.g., data hall CRAHs' supply air temperature, into the problem formulation eq. (3a), and get the optimal control efficiently with a gradient-based optimization technique. Moreover, similar to the previous work [12], [51], the energy storage system can also be integrated into the existing framework to dynamically adjust its charge and discharge to further reduce carbon emissions and improve service quality.

### C. Fair Schedule Issue

In this work, we attempt to schedule the execution of delay-tolerant workloads to match the renewable energy generation to minimize the DC's carbon emissions. Our empirical result shows that the tasks with long runtime are more likely to be delayed to reduce the carbon footprint. This might raise the fairness concern. In our future study, we plan to further consider workload execution fairness by adding fairness as another objective in our original formulation eq. (3a), similar to [40]. By doing so, we can holistically consider sustainability and fairness when designing the capacity planner and the workload scheduler.

### D. Real-time Scheduler for Better SLO Satisfaction

In this work, we focus on the design of the carbon-aware capacity planner, and it is found that some SLO violations are still inevitable even if all uncertainties are eliminated. To improve the service quality for some critical computing tasks, the real-time scheduler should be designed to handle the potential resource deficit. One possible solution is to schedule a high-priority task to run without considering the capacity budget determined by the MPC planner when the real-time scheduler detects that it is likely to violate the deadline. It is also possible to overestimate the job runtime as well as the

future computing demand as the LSTM predictor can produce the posterior distribution of the job runtime. For instance, we can use the 95% percentile instead of the expected value as the job runtime estimation. With the overestimated job runtime, the capacity planner will over-reserve the computing resources and reduce the risk of missing some critical computing tasks.

## VIII. Conclusions

In this paper, a differentiable physics-informed digital twin is developed for holistic DC operational carbon footprint evaluation, and it is integrated into an MPC-based capacity provisioning framework for DC decarbonization. Two techniques are proposed to mitigate the negative impacts caused by forecasting errors. The first technique involves conducting workload runtime re-estimation to compensate for computing demand forecasting errors resulting from noisy runtime estimation. Additionally, the concept of "cumulative supply deficit" is introduced to address computing resource supply deficits through a shrinking-horizon MPC reformulation. The approach is evaluated using real operational data from a large-scale co-located DC in Singapore. The accuracy of the digital twin is validated, with an observed RMAE of approximately 3% for holistic DC energy footprint estimation. We also show that the digital twin can learn with high dimensional DC operational data with limited dynamical range while vanilla deep neural work fails to learn, which indicates the necessity of physics-informed modeling in the DC application. Furthermore, we compare our design against various baseline methods, demonstrating comparable performance to the oracle scheme with perfect knowledge of uncertainties and achieving approximately 27% more carbon reduction compared to the existing industrial practice. The necessity of runtime re-estimation in the MPC planner is also demonstrated, along with the robustness of our design in the presence of noisy forecasting. We further compare different real-time schedulers' performance and conclude that the job duration should be considered under time-varying capacity provisioning.

## References


[1] J. Davis, D. Bizo, A. Lawrence, O. Rogers, and M. Smolaks, "Uptime institute global data center survey 2022: Resiliency remains critical in a volatile world," 2022, https://uptimeinstitute.com/resources/research-and-reports/uptime-institute-global-data-center-survey-results-2022.

[2] D. Hardcastle, G. Mattios, V. Kulkarni, L. Paglia, and F. Teo, "Southeast asia's green economy 2021 report: Opportunities on the road to net zero," 2021, https://www.bain.com/globalassets/noindex/2021/green-economy/bain-microsoft-temasek-sea-green-economy-2021-report-road-to-net-zero-main.pdf.

[3] Z. Cao, X. Zhou, X. Wu, Z. Zhu, T. Liu, J. Neng, and Y. Wen, "Data center sustainability: Revisits and outlooks," *IEEE Transactions on Sustainable Computing*, pp. 1–13, 2023.

[4] Z. Cao, X. Zhou, H. Hu, Z. Wang, and Y. Wen, "Toward a systematic survey for carbon neutral data centers," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 2, pp. 895–936, 2022.

[5] A. Radovanović, R. Koningstein, I. Schneider, B. Chen, A. Duarte, B. Roy, D. Xiao, M. Haridasan, P. Hung, N. Care *et al.*, "Carbon-aware computing for datacenters," *IEEE Transactions on Power Systems*, vol. 38, no. 2, pp. 1270–1280, 2022.

[6] E. Jaureguialzo, "Pue: The green grid metric for evaluating the energy efficiency in dc (data center). measurement method using the power demand," in *2011 IEEE 33rd International Telecommunications Energy Conference (INTELEC)*. IEEE, 2011, pp. 1–8.

[7] D. B. Crawley, L. K. Lawrie, F. C. Winkelmann, W. F. Buhl, Y. J. Huang, C. O. Pedersen, R. K. Strand, R. J. Liesen, D. E. Fisher, M. J. Witte *et al.*, "Energyplus: creating a new-generation building energy simulation program," *Energy and buildings*, vol. 33, no. 4, pp. 319–331, 2001.

[8] A. H. Mahmud and S. Ren, "Online capacity provisioning for carbon-neutral data center with demand-responsive electricity prices," *ACM SIGMETRICS Performance Evaluation Review*, vol. 41, no. 2, pp. 26–37, 2013.

[9] H. Dou, Y. Qi, W. Wei, and H. Song, "Carbon-aware electricity cost minimization for sustainable data centers," *IEEE Transactions on Sustainable Computing*, vol. 2, no. 2, pp. 211–223, 2017.

[10] M. J. Neely, "Stochastic network optimization with application to communication and queueing systems," *Synthesis Lectures on Communication Networks*, vol. 3, no. 1, pp. 1–211, 2010.

[11] Í. Goiri, K. Le, M. E. Haque, R. Beauchea, T. D. Nguyen, J. Guitart, J. Torres, and R. Bianchini, "Greenslot: scheduling energy consumption in green datacenters," in *Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis*, 2011, pp. 1–11.

[12] Z. Liu, Y. Chen, C. Bash, A. Wierman, D. Gmach, Z. Wang, M. Marwah, and C. Hyser, "Renewable and cooling aware workload management for sustainable data centers," in *Proceedings of the 12th ACM SIGMETRICS/PERFORMANCE joint international conference on Measurement and Modeling of Computer Systems*, 2012, pp. 175–186.

[13] Í. Goiri, W. Katsak, K. Le, T. D. Nguyen, and R. Bianchini, "Parasol and greenswitch: Managing datacenters powered by renewable energy," *ACM SIGPLAN Notices*, vol. 48, no. 4, pp. 51–64, 2013.

[14] K. Zhang, P. Wang, N. Gu, and T. D. Nguyen, "Greendrl: managing green datacenters using deep reinforcement learning," in *Proceedings of the 13th Symposium on Cloud Computing*, 2022, pp. 445–460.

[15] B. Acun, B. Lee, F. Kazhamiaka, K. Maeng, U. Gupta, M. Chakkaravarthy, D. Brooks, and C.-J. Wu, "Carbon explorer: A holistic framework for designing carbon aware datacenters," in *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*, 2023, pp. 118–132.

[16] M. Shafto, M. Conroy, R. Doyle, E. Glaessgen, C. Kemp, J. LeMoigne, and L. Wang, "Modeling, simulation, information technology & processing roadmap," *National Aeronautics and Space Administration*, vol. 32, no. 2012, pp. 1–38, 2012.

[17] B. Kouvaritakis and M. Cannon, "Model predictive control," *Switzerland: Springer International Publishing*, vol. 38, 2016.

[18] Z. Cao, R. Wang, X. Zhou, and Y. Wen, "Toward model-assisted safe reinforcement learning for data center cooling control: A lyapunov-based approach," in *Proceedings of the 14th ACM International Conference on Future Energy Systems*, 2023, pp. 333–346.

[19] R. Wang, Z. Cao, X. Zhou, Y. Wen, and R. Tan, "Green data center cooling control via physics-guided safe reinforcement learning," *ACM Transactions on Cyber-Physical Systems*, 2022.

[20] ——, "Phyllis: Physics-informed lifelong reinforcement learning for data center cooling control," in *Proceedings of the 14th ACM International Conference on Future Energy Systems*, 2023, pp. 114–126.

[21] H. D. Vu, K. S. Chai, B. Keating, N. Tursynbek, B. Xu, K. Yang, X. Yang, and Z. Zhang, "Data driven chiller plant energy optimization with domain knowledge," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 1309–1317.

[22] R. Yadav and W. Zhang, "Mereg: managing energy-sla tradeoff for green mobile cloud computing," *Wireless Communications and Mobile Computing*, vol. 2017, no. 1, p. 6741972, 2017.

[23] R. Yadav, W. Zhang, H. Chen, and T. Guo, "Mums: Energy-aware vm selection scheme for cloud data center," in *2017 28th International Workshop on Database and Expert Systems Applications (DEXA)*. IEEE, 2017, pp. 132–136.

[24] R. Yadav, W. Zhang, O. Kaiwartya, P. R. Singh, I. A. Elgendy, and Y.-C. Tian, "Adaptive energy-aware algorithms for minimizing energy consumption and sla violation in cloud computing," *Ieee Access*, vol. 6, pp. 55 923–55 936, 2018.

[25] R. Yadav, W. Zhang, K. Li, C. Liu, M. Shafiq, and N. K. Karn, "An adaptive heuristic for managing energy consumption and overloaded hosts in a cloud data center," *Wireless Networks*, vol. 26, pp. 1905–1919, 2020.

[26] R. Yadav, W. Zhang, K. Li, C. Liu, and A. A. Laghari, "Managing overloaded hosts for energy-efficiency in cloud data centers," *Cluster Computing*, pp. 1–15, 2021.

[27] M. D. M. da Silva, A. Gamatié, G. Sassatelli, M. Poss, and M. Robert, "Optimization of data and energy migrations in mini data centers for

carbon-neutral computing," *IEEE Transactions on Sustainable Computing*, vol. 8, no. 1, pp. 68–81, 2023.

[28] L. Lin and A. A. Chien, "Adapting datacenter capacity for greener datacenters and grid," in *Proceedings of the 14th ACM International Conference on Future Energy Systems*, 2023, pp. 200–213.

[29] N. Hogade, S. Pasricha, H. J. Siegel, A. A. Maciejewski, M. A. Oxley, and E. Jonardi, "Minimizing energy costs for geographically distributed heterogeneous data centers," *IEEE Transactions on Sustainable Computing*, vol. 3, no. 4, pp. 318–331, 2018.

[30] M. S. Hasan, Y. Kouki, T. Ledoux, and J.-L. Pazat, "Exploiting renewable sources: When green sla becomes a possible reality in cloud computing," *IEEE Transactions on Cloud Computing*, vol. 5, no. 2, pp. 249–262, 2015.

[31] P. Wiesner, I. Behnke, D. Scheinert, K. Gontarska, and L. Thamsen, "Let's wait awhile: How temporal workload shifting can reduce carbon emissions in the cloud," in *Proceedings of the 22nd International Middleware Conference*, 2021, pp. 260–272.

[32] C. Li, R. Wang, T. Li, D. Qian, and J. Yuan, "Managing green datacenters powered by hybrid renewable energy systems," in *11th International Conference on Autonomic Computing (ICAC 14)*, 2014, pp. 261–272.

[33] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[34] J. D. Moore, J. S. Chase, P. Ranganathan, and R. K. Sharma, "Making scheduling" cool": Temperature-aware workload placement in data centers." in *USENIX annual technical conference, General Track*, 2005, pp. 61–75.

[35] T. L. Bergman, *Fundamentals of heat and mass transfer*. John Wiley & Sons, 2011.

[36] W. E. Forsthoffer, *Forsthoffer's best practice handbook for rotating machinery*. Elsevier, 2011.

[37] S. Bergsma, T. Zeyl, A. Senderovich, and J. C. Beck, "Generating complex, realistic cloud workloads using recurrent neural networks," in *Proceedings of the ACM SIGOPS 28th Symposium on Operating Systems Principles*, 2021, pp. 376–391.

[38] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[39] D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski, "Deepar: Probabilistic forecasting with autoregressive recurrent networks," *International Journal of Forecasting*, vol. 36, no. 3, pp. 1181–1191, 2020.

[40] J. Xing, B. Acun, A. Sundarrajan, D. Brooks, M. Chakkaravarthy, N. Avila, C.-J. Wu, and B. C. Lee, "Carbon responder: Coordinating demand response for the datacenter fleet," *arXiv preprint arXiv:2311.08589*, 2023.

[41] B. Liang, T. Mitchell, and J. Sun, "Ncvx: A general-purpose optimization solver for constrained machine and deep learning," *arXiv preprint arXiv:2210.00973*, 2022.

[42] E. Cortez, A. Bonde, A. Muzio, M. Russinovich, M. Fontoura, and R. Bianchini, "Resource central: Understanding and predicting workloads for improved resource management in large cloud platforms," in *Proceedings of the 26th Symposium on Operating Systems Principles*, 2017, pp. 153–167.

[43] J. W. Park, A. Tumanov, A. Jiang, M. A. Kozuch, and G. R. Ganger, "3sigma: distribution-based cluster scheduling for runtime uncertainty," in *Proceedings of the Thirteenth EuroSys Conference*, 2018, pp. 1–17.

[44] EMA., "Energy transformation (chapter 2)," 2022, https://www.ema.gov.sg/singapore-energy-statistics/Ch02/index2.

[45] X. Cao and K. R. Liu, "Online convex optimization with time-varying constraints and bandit feedback," *IEEE Transactions on automatic control*, vol. 64, no. 7, pp. 2665–2680, 2018.

[46] L. Janson, E. Schmerling, and M. Pavone, "Monte carlo motion planning for robot trajectory optimization under uncertainty," in *Robotics Research: Volume 2*. Springer, 2017, pp. 343–361.

[47] A. Guideline *et al.*, "Measurement of energy, demand, and water savings," *ASHRAE guideline*, vol. 4, pp. 1–150, 2014.

[48] M. Tirmazi, A. Barker, N. Deng, M. E. Haque, Z. G. Qin, S. Hand, M. Harchol-Balter, and J. Wilkes, "Borg: the next generation," in *Proceedings of the fifteenth European conference on computer systems*, 2020, pp. 1–14.

[49] C. Zhang and A. A. Chien, "Scheduling challenges for variable capacity resources," in *Job Scheduling Strategies for Parallel Processing: 24th International Workshop, JSSPP 2021, Virtual Event, May 21, 2021, Revised Selected Papers 24*. Springer, 2021, pp. 190–209.

[50] Y. Ran, X. Zhou, H. Hu, and Y. Wen, "Optimizing data center energy efficiency via event-driven deep reinforcement learning," *IEEE Transactions on Services Computing*, vol. 16, no. 2, pp. 1296–1309, 2023.

[51] W. Deng, F. Liu, H. Jin, C. Wu, and X. Liu, "Multigreen: Cost-minimizing multi-source datacenter power supply with online control," in *Proceedings of the fourth international conference on Future energy systems*, 2013, pp. 149–160.

**Zhiwei Cao** received the B.Eng. degree in communications engineering from Jilin University in 2018, and the M.Eng. degree in information and signal processing from the School of Electronics Engineering and Computer Science, Peking University, in 2021. He is currently pursuing a Ph.D. degree with the College of Computing and Data Science, Nanyang Technological University, Singapore. His current research interests include digital twins, sustainable computing, and data center modeling and optimization.

**Dr. Ruihang Wang** received his Ph.D. degree from the School of Computer Science and Engineering, Nanyang Technological University, Singapore, in 2023. Before that, he received his M.Eng. and B.Eng. degrees from the School of Precision Instrument and Opto-electronics Engineering, Tianjin University (TJU), China, in 2019 and 2016, respectively. He is currently a research fellow at Nanyang Technological University (NTU), Singapore. His research interests include cyber-physical systems, machine learning, and energy-efficient data centers.

**Dr. Xin Zhou** received M.E. and Ph.D. degrees from the Department of Information Engineering, Hiroshima University, Japan, in 2013 and 2016, respectively. He is now an Assistant Professor in the School of Information and Mechatronics Engineering at Jiangxi Science and Technology Normal University, China. His current research focuses on learning-based data center ICT and facility optimization. He received the Industrial Technical Excellence Award of the IEEE Technical Committee on Cyber-Physical Systems in 2020.

**Dr. Rui Tan** is currently an Associate Professor with the College of Computing and Data Science of NTU. He is a Senior Member of IEEE. His research expertise is in sensor networks, the Internet of Things (IoT), and cyber-physical systems. He obtained his PhD degree in computer science from the City University of Hong Kong in 2010. Now, he is leading the NTU IoT Sensing Group which focuses on the research, design, and evaluation of networked, energy-efficient, and secure sensing systems.

**Dr. Yonggang Wen** is a Professor and President's Chair in Computer Science and Engineering at Nanyang Technological University (NTU), Singapore. He is Fellow of IEEE and the Singapore Academy of Engineering, and ACM Distinguished Member. He received his Ph.D. in Electrical Engineering and Computer Science from Massachusetts Institute of Technology (MIT), Cambridge, USA 2008. Dr. Wen has published over 300 papers in top journals and prestigious conferences. His research interests include cloud computing, green data centers, big data analytics, multimedia networks, and mobile computing. He is the Editor in Chief of IEEE Transactions on Multimedia (TMM), and serves or has served on editorial boards for multiple IEEE and ACM transactions.

**Dr. Yuejun Yan** is currently the Technical Lead of Global Data Center Energy & Carbon Innovation, Alibaba Cloud. She received her Ph.D. degree from the University of Pennsylvania. Her research interest is in Green AI and green Cloud Computing.

**Zhaoyang Wang** is currently the General Manager of the Global Data Center, Alibaba Cloud. He leads global data center planning, delivery, R&D, and operations.