energies

Review

Simulators for Conversing Power to Thermal on Green Data Centers: A Review

Danyang Li [1], Jie Song [1,*], Hui Liu [2] and Jingqing Jiang [3]

[1] Software College, Northeastern University, Chuangxin Road No. 195, Shenyang 110819, China; 2110497@stu.neu.edu.cn
[2] School of Metallurgy, Northeastern University, Wenhua Road 3 Lane No. 11, Shenyang 110819, China; liuhui@mail.neu.edu.cn
[3] College of Computer Science and Technology, Inner Mongolia MinZu University, Linhe Road No. 536, Tongliao 028000, China; jiangjingqing@aliyun.com
* Correspondence: songjie@mail.neu.edu.cn

Abstract: This paper aims to help data center administrators choose thermal simulation tools, which manage thermal conduction from power for energy savings. When evaluating and suggesting data center thermal simulators for users, questions such as "What are the simulator's differences? Are they easy to use? Which is the best choice?" are frequently asked. To answer these questions, this paper reviews the thermal simulation works for data centers in the last ten years. After that, it proposes the versatility and dexterity metrics for these simulators and discovers that it is difficult to choose them despite their similar design purpose and functions. Empowered by the survey, we claim that the widespread practice simulators still need more enhancement in data center scenarios. We back up our claim by comparing typical simulators and propose improvements to thermal simulators for future studies.

Keywords: electric energy; simulator; green data centers; energy conversion; energy management

check for updates

Citation: Li, D.; Song, J.; Liu, H.; Jiang, J. Simulators for Conversing Power to Thermal on Green Data Centers: A Review. *Energies* **2024**, *17*, 5631. https://doi.org/10.3390/en17225631

Academic Editor: Andrea Frazzica

Received: 20 October 2024
Revised: 6 November 2024
Accepted: 7 November 2024
Published: 11 November 2024

1. Introduction

With the age of big data and global informationization, data centers have become an indispensable infrastructure for modern society [1]. The growing online services, such as cloud computing, mobile applications, and big data storage and processing, rely on the support of such infrastructure. As shown in [2], the incoming load to data centers doubled from 2015 to 2020. Therefore, large data center operators and vendors have to invest more in computing resources. Increasing computational power and density imply a greater cooling cost. For example, the cooling infrastructure is responsible for about 30–40 percent of the total system's power consumption [3]. A data center produces a large amount of heat due to its high energy consumption. Thermal imbalance and hotspots would occur in a data center without proper thermal management, further reducing computing efficiency and hardware stability and even damaging hardware and breaking down the data center [4].

The state-of-art studies on thermal management generally consider the following terms: cooling strategies, hotspot elimination, thermal air management, thermal-aware workload scheduling, resource allocation, and virtual machine consolidation. In these studies, experiments that prove the optimization effects can hardly run on a real-world data center because establishing hardware, software, and energy supply is complicated and costly. Therefore, most studies prefer simulated data centers rather than real-world data centers; namely, experiments are performed on a thermal simulator. Figure 1 shows the thermal management studies and their methods, objects, models, and relationships with a data center simulator. According to our statistics, 75% of related works have adopted simulators in their experiments in the past ten years. Although simulators of these studies follow the essential simulation method, their design purposes, functionality, and usability vary. For example, simulators are designed for static optimization on data center design only or support rich-featured input workloads. Unfortunately, few studies survey these simulators and analyze their differences. As a result, the evaluation and selection of thermal simulators in data centers has become a challenge.

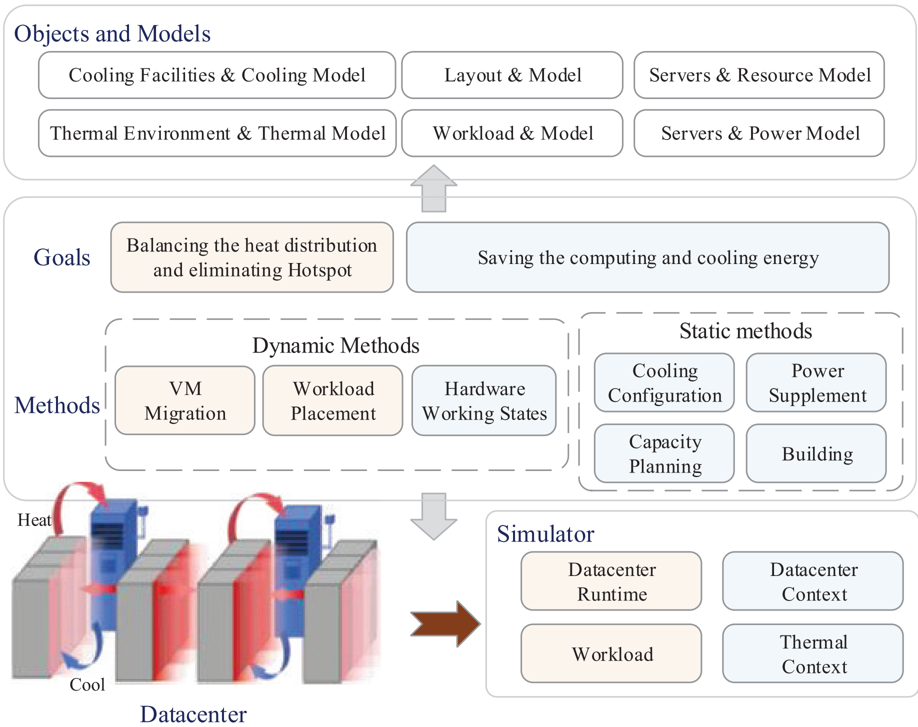

Figure 1. Experimental studies on thermal management with simulator against real-world datacenter. The objective of studies is to balance the heat distribution and/or save energy through static or dynamic methods. The adjustable objects and corresponding models are at the top of the figure. All the models and methods can be executed in a simulator with context and runtime simulation.

This paper comprehensively investigates state-of-the-art simulators. These simulators have similar targets but distinguish functions. Moreover, some simulators are easy to learn, access, deploy, and extend. The usability of these simulators also varies. We conclude with two general metrics to evaluate these simulators: versatility and dexterity. The versatility is that simulators could provide rich functions for various and complex experimental contexts. The dexterity is about usability: simulators could be simple, intuitive, efficient, and easy to use. The survey results show that Fluent [5] is the most popular data center thermal simulator, suitable for many scenarios. The other simulators, such as CloudSim [6] and MatLab [7], also show their advances in other scenarios. For example, Fluent is not the best choice in the scenarios of dynamic thermal management considering larger-scale distributed workloads, while CloudSim also requires improvement. To this end, we propose improvements for the thermal simulator in terms of both versatility and dexterity for widespread simulators, such as Fluent, in given scenarios. We back up our proposition with uniform controlling metrics and comparisons on typical simulators. The key contributions of our research are as follows:

- The analysis of state-of-the-art simulators brings a comprehensive view of the research topic. Researchers who study thermal management in data centers could quickly understand experimental environments and methods through our work.
- The suggestion for better simulators based on the proposed versatility and dexterity metrics complements the common scene of simulator selection.

- The summaries on thermal simulators are from two dimensions: how does a simulator model context, heating, and cooling, and how easy is it to learn, deploy, execute, and extend a simulator? They bring new experiences to the thermal management study.

The remainder of the paper is organized as follows: Section 2 introduces our research method, including research questions and literature collection. The following three sections answer the three research questions: Sections 3 and 4 analyze the versatility and dexterity of state-of-the-art thermal simulators. Section 5 compares typical simulators and proposes suggestions for selection. Section 6 gives the outlook of thermal simulators. The last section concludes the paper.

2. Our Methods

2.1. Research Questions

RQ1: Do state-of-the-art simulators have the same functions as thermal simulators for green data centers? What are these functions designed for?

RQ2: How easy is it to apply these simulators in thermal management research for green data centers?

RQ3: Are the widely accepted simulators, such as Fluent, enough for thermal simulation of data centers? Are there more expectations?

2.2. Search Methods

We conduct a preliminary search about data centers and thermal management using DBLP.org (https://dblp.uni-trier.de, accessed on 26 October 2023) and IEEE Explore. In IEEE Explore, the query conditions are "data center AND simulation AND (thermal OR heat OR cooling OR hotspot)". In DBLP.org, the search keywords are "thermal data center" since it can only search titles. Moreover, we focus on English-written peer-reviewed papers published in journals and proceedings of conferences after 2014. Meanwhile, papers about building, construction, architecture, and civil engineering may not adopt the software simulation and solution, or papers whose quality of service, energy, security, and other optimization techniques address thermal management issues only in an incidental manner are also excluded. Then, we obtain 86 candidates, examine the experimental environment of the candidates, and select 54 papers that engaged data center simulations as survey targets. Among the exclusions, some papers are without experiments, and 12 papers perform experiments in real-world data centers.

2.3. Inclusive Objectives

This subsection summarizes the research objectives of the 54 related studies because these objectives are requirements for the simulators. The popular simulators are countable, and the experiments on these simulators are also for one goal, namely, the green data center simulation; however, the objectives of these experimental simulations vary according to the research objective of each work. The objectives have four aspects: goal, opportunity, object, and method:

- The goal means the primary research objective, such as balancing the heat distribution, saving the energy of data centers, both of the former two, and performance optimization.
- The opportunity means when the optimization occurs. It could be the static optimization in the data center design stages, such as the building, capacities, power supplement, and cooling configuration. It also involves dynamic optimization in the data center runtime, such as workload placement and VM (virtual machine) migration.
- The object means the concepts or components the optimization aims at, such as servers, workload, cooling facilities, and thermal environment.
- The method means the optimization method adopted in a study, such as hardware acceleration, thermal-oriented method, heuristic and meta-heuristic algorithms, and AI (artificial intelligence) empowered method.

Table 1 lists the references for inclusive objectives. The percentages of cooling facilities, thermal environment, workload, and servers are 38%, 23.9%, 23.9%, and 14.1%, respectively. Because cooling facilities directly change the temperature of data centers, it accounts for the most significant proportion of the four objects. Thermal environment control is an indirect method of adjusting the temperature of data centers. Furthermore, servers are the source of heat, so the studies on the servers' working state (runtime) and thermal state (context) are new directions. For example, workload-oriented dynamic optimization for balancing heat distribution on data center runtime with heuristics and AI empower methods.

Table 1. Summarizing on objectives of reviewed papers.

Title 1	Title 2	References
Goals	Thermal balance (13)	[4,5,8–18]
	Energy Saving (26)	[6,19–41]
	Performance optimization (8)	[4,26,42–47]
	Both thermal and energy (3)	[48–50]
Opportunities	Static optimization on data center design (12)	[8–10,14,15,21,25,30,31,34,41]
	Dynamic optimization on data center runtime (22)	[6,11–13,15–17,19–21,23,26–29,33,35,36,39,42,48,49]
Objects	Servers (7)	[6,12,17,21,35,36,39]
	Workload (12)	[4,6,13,21,24,25,28–30,49]
	Cooling facilities (19)	[4–6,11,14,16,23,26,27,30,31,33,34,38,40,41,43,48,50]
	Thermal environment (12)	[5,8–10,15,18–21,32,42,44]
Methods	Hardware (7)	[5,11,14,23,34,41,46]
	Planning (13)	[4,8,9,21,27,31,32,43,44,48,50]
	Heuristic (7)	[5,6,13,17,19,25,29]
	AI empower (9)	[15,16,18,20,24,33,36,38,40]

2.4. Inclusive Simulators

Table 2 lists the mentioned simulators in the selected 54 papers. These papers mainly employ five typical simulators: Fluent, CloudSim, MatLab, 6SigmaDC, and self-developed. The other simulators mentioned in Table 2, such as Energyplus [39] and Openfoam, have only a few references. Furthermore, the "self-developed" is not a single simulator but a collection of simulators that are developed for non-general purposes in some papers. It only applies to the corresponding studies. However, some of them have common features. Therefore, the rest of the paper groups the featured ones as the "self-developed simulator" for comparison, discusses versatility and dexterity, performs comparative analysis, and answers the research questions based on the five typical simulators.

Table 2. References for related works.

Simulator (Count)	References
Fluent (19)	[5,8–15,19–24,42,43,48,51]
MatLab (11)	[7,16,21,25–30,50]
Self-developed (5)	[37,38,45–47]
6SigmaDC (5)	[31–34,44]
CloudSim (5)	[4,6,17,18,35]
Energyplus (2)	[39,40]
Anysis icepak (2)	[4,41]
Tile flow (2)	[52,53]
TRNSYS (1)	[54]

Fluent is the most popular one, accounting for about 40% of all simulation experiments. Next, MatLab, self-developed, CloudSim, and 6SigmaDC take about 20%, 10%, 8.5%, and 8.5%, respectively. Since Fluent is a professional thermal simulator, it is the most common in data center thermal simulation. MatLab is a classic mathematical simulator. It is widely used in all kinds of simulation experiments, such as data center thermal simulation. CloudSim and 6SigmaDC are simulators specially designed for data center simulation. They take as large a proportion as the self-developed simulator.

Figure 2 shows the time distribution of references for the five simulators. The trend of applying these simulators is pronounced as follows: Fluent is always popular. Fluent was the only choice for the first three years until MatLab integrated the data center thermal models in 2015. With the development of large-scale distributed systems and cluster systems, CloudSim also integrated with thermal models and has become a popular simulator since 2015, but it is still not comparable with Fluent and MatLab. As a commercial simulator, 6SigmaDC only appeared from 2017 to 2020. However, researchers have tended to develop their own in the last three years. Thus, self-developed simulators spread widely.

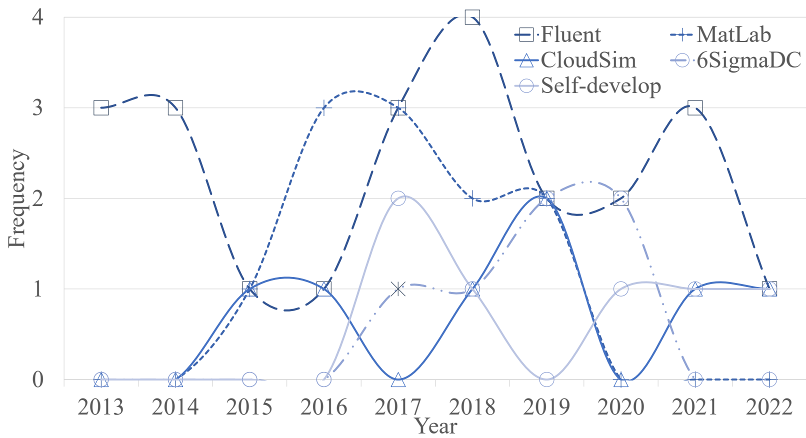

Figure 2. Time distribution of references for the five simulators. We normalized the referring frequency of a simulator in a year only according to the simulator itself. The frequency is zero if there is no reference in the year; it is 1 if the reference number of this year is the minimum among these years; and it is n if the reference number of this year is n times the minimum one.

3. Versatility

This section studies the versatility of thermal simulators for data centers. The versatility means that simulators could provide rich functions for various and complex experimental contexts. It is also closely related to the objectives mentioned in Section 2.3. First, simulators follow the same models as shown in Figure 1:

- The data center context is the static configuration of a data center, such as the building, the environment, and the servers. It is determined in the design stage.
- The cooling facilities, known as the configuration of cooling equipment, are part of the context.
- The data center runtime represents the dynamic state when a data center runs for workloads. How to deal with workloads, such as generation, scheduling, placement, and execution, dominates the data center runtime.
- The power model defines how the data center's power changes with its runtime, such as the power input and energy consumption models.
- The thermal model defines how the data center's thermal environment changes with its runtime, such as thermal distribution, transfer, and exchange.

This section studies versatility from the above five aspects. It calls them the five dimensions and discusses them in the following five subsections. For a subsection, first, it surveys the common and different functions of the simulations. The commonalities mean most studies adopt these features, and the differences mean the opposite. Differences are the key to distinguishing and evaluating the simulators. Therefore, it defines the functional "indicators" abstracted from the differences in the corresponding "definition". Then, a "reference table" summarizes the references for each indicator, and a "heatmap" shows quantified values of indicators distributed in studies. The quantified value of a simulator's indicator is the proportion of the studies that adopted the indicator. Finally, it discusses the findings on the dimension according to the reference table and the heatmap and shows them in the corresponding "remark". For example, the advantages and disadvantages of different simulators on these indicators. After reviewing all dimensions, the last subsection answers the RQ1.

3.1. Data Center Context

When modeling the data center context, the geographical attributes, building structures, and environments impact the simulation effect. However, such an impact does not change with the runtime. Ninety percent of studies do not mention the data center context, and ten percent of studies mentioned part of them, such as the latitude and longitude [55], covered area [56], temperature [57], and humidity. Notice that the cooling configuration is also a context and highlighted in the next dimension. Furthermore, 95/96 studies mention one critical configuration: the data center scale.

The scale of the data center dominates the data center runtime and thermal environment, which are the two main objectives of current studies. Most of the studies consider homogeneous data centers, and only a few studies define the data center bandwidth. The server specification is mandatory. It includes scale, cores, memory, and storage. A few works defined the specification in terms of known cases, such as the publicly available technical report [58], data center infrastructure of IBM and HP, Eco4Cloud, PlanetLab traces, and PUE value [59]. Furthermore, most works defined the performance specification by servers' capacities, namely, how fast the servers handled workloads. Despite the various server specifications, the diversity is enumerable, and the difference is insignificant. In other words, the different specifications of a single server can not greatly change the runtime and environment; however, the number of servers in a data center can. In this way, the server scale is a typical attribute of the data center scale.

Servers configured to the experiments in the studies are more or less. Meanwhile, no simulator has a fixed-scale data center, leading to a considerable overlap in the number of servers. For example, Tang et al. simulate a single server data center. In contrast, Fang et al. [21] simulate the clusters with several thousands of servers. The data center scale also expands from small to large. For example, Alkharabsheh et al. adopted Fluent to simulate data centers ranging from 10 servers to 1000 servers. Shi et al. [32] also perform the simulation on 6SigmaDC, ranging from 100 to 1000 servers. Studies focused on the data center runtime prefer larger scales because they give space for optimization, such as heuristic-based or AI-empowered workload management. However, it brings challenges to the thermal simulation model. On the contrary, studies focused on the data center thermal prefer small scales because it is easy to calculate and simulate the detailed changes in the thermal environment.

As a result, the number of servers is the typical difference in the data center context, and it is a good representation of the data center scale.

Definition 1 (**Data Center Context Indicator**)**.** *The static context indicators show the data center scale in the number of servers. Three indicators are small-scale, middle-scale, and large-scale. The orders of numbers represent that the number of servers in the simulated data center is up to one hundred, less than a thousand, and more than a thousand, respectively.*

Remark 1 (**Simulation Scale**)**.** *CloudSim is best suited to simulate large-scale data centers. Table 3 shows the references of three static context indicators on five simulators. Figure 3 shows the heatmap of three data center context indicators on five simulators. Fluent, the most popular simulator, supports all scales; however, experiments with smaller scale data centers are more likely* *to choose Fluent than ones with a more significant scale. 6SigmaDC has only a few applications with all scales. CloudSim is designed for cloud system simulation, so it is mainly for simulating a large number of servers. In contrast, MatLab and self-developed simulators are used to simulate middle- and small-scale data centers, possibly because the efforts to develop a more extensive scale are enormous.*

Table 3. References of three static context indicators on five simulators.

	Small Scale	Middle Scale	Large Scale
Fluent	[5,8–11,13,23,48,51]	[10,20,24,49]	[12,21]
CloudSim	None	[18]	[4,6,17,35]
MatLab	[28,50]	[7,25,27,30]	None
6SigmaDC	[44]	[32,34]	[31,32]
Self-developed	[37,38]	None	None

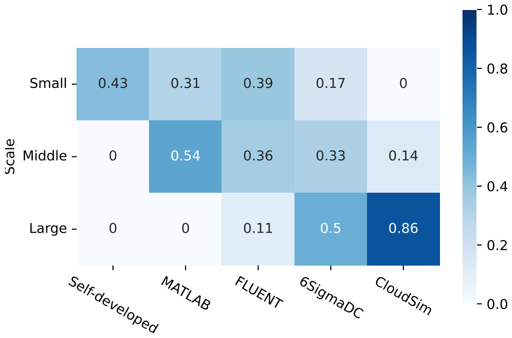

Figure 3. Heatmap of the data center context indicators on five simulators, ordered by the ability to simulate large data centers.

3.2. Cooling Facilities

Cooling simulation is the mandatory function of thermal simulators for the green data center. Users configure the type and amount of cooling facilities according to their experimental purposes. In these papers, data centers have three kinds of cooling equipment to be simulated: fans, CRAC, and CRAH (Computer Room Air Handler).

CRAC is the most common cooling facility widely used in all kinds of traditional data centers. It works like a conventional home air conditioner, passing air through cooling coils filled with refrigerant. CRAH is a novel cooling facility for large data centers in recent years [27]. The device draws warm air from the computer room through chilled water coils. Heat is transferred from the air to the water and back to the cooler. Due to the difference in principle, CRAH is more effective than CRAC; its usage is more flexible, but its arrangement is more complex. The fan does not directly lower the ambient temperature compared with CRAC and CRAH. Since the heat value is the product of density, flow rate, specific heat capacity, and temperature, the fan increases the airflow rate by enhancing the heat elimination ability and reduces the temperature in data centers [7].

A green data center employs one or all of them as the cooling equipment. For example, Sakanova et al. [19] and Jafarizadeh et al. [8] adopt fans and CRAH simultaneously, thus obtaining excellent performance in the data center simulation through Fluent. Alkharabsheh et al. adopt fans and CRAC together through Fluent. Mousavi et al. [27] combine CRAC and CRAH through MatLab.

In the simulation of data centers with fans, the heat transmission process, such as air velocity and heat transfer efficiency in the air, must be considered. In the thermal simulation of CRAC and CRAH, the layout of cold and hot channels and the refrigeration efficiency of the equipment should be fully considered. As a result, the type of cooling equipment, such as air-flowing fans, air-compression CRAC, and liquid-flowing CRAH, are the typical differences in modeling cooling facilities.

Definition 2 (**Cooling Indicator**)**.** *The cooling indicators show the techniques and equipment for cooling the green data center. The three indicators are air compression, liquid circulation, and air circulation. As their names, the first represents CRACs. The second and the third represent any CRAHs and fans in simulation, respectively.*

Remark 2 (**Cooling Facilities Simulation**)**.** *MatLab has the best cooling simulation abilities, while Cloudsim has the worst one. Table 4 shows references of three cooling indicators on five simulators. Figure 4 shows the heatmap of three cooling indicators on five simulators. All simulators have the most extensive application for fundamental air compression. Although simulators can simulate heat in data centers, the popular thermal simulator Fluent and the classical mathematical simulator MatLab obviously have better versatility in simulating cooling facilities. However, they are more suitable for simulating small and medium-sized data centers. Large-scale data centers and intensified cooling facilities will be the trends, and liquid circulation is more suitable for large data centers because of its high cooling efficiency. CloudSim is the best for large data center simulation; however, it does not support liquid-circulation. From the simulation perspective, liquid circulation and air compression have no more differences than the heat transformation coefficients. Therefore, extending CloudSim's cooling facilities for liquid circulation in future works is feasible.*

Table 4. References of three cooling indicators on five simulators.

	Air com.	Liquid cir.	Air cir.
Fluent	[5,10,12,20,21,23,48,49,51]	[8,19]	[8,11,19,24]
CloudSim	[4,6,17,18]	None	None
MatLab	[7,27,30,50]	[27,28]	[26]
6SigmaDC	[31–34,44]	None	None
Self-developed	[37,47]	[47]	None

Figure 4. Heatmap of three cooling indicators on five simulators, ordered by the comprehensive ability of cooling simulation.

3.3. Data Center Runtime

The data center runtime mainly includes the working states of the server, the communication equipment, and the peripherals. The runtime of communication equipment and peripherals affects neither the thermal simulation goals, such as energy consumption and heat balance, nor the control objectives, such as servers and cooling facilities. The runtime of servers includes resource utilization, energy efficiency, and heat generation. The latter two depend on the server's hardware features. Therefore, resource utilization, which relies on the amount of workload executed on it, dynamically affects the server and data center runtime. As a result, the workload runtime determines the data center runtime. Simulation on workload runtime has two sides: workload generation and workload placement. The simulation for the data center has three considerations when generating workload:

- The workload scale should match the data center scale. A workload that is too excess or scarce will invalidate the simulation objectives. For example, Van Damme et al. [30]

adopted the small-scale workload measured in the laboratory when simulating the data center with small-scale servers. To simulate the cloud data center, Ali et al. [6] chose the access data of Amazon as the workload.

- The workload has homogeneous or heterogeneous tasks. Most simulations choose the heterogeneous one [7] to be more practical. However, Internet data centers may deal with various types of workloads on the Internet, such as content access, database query, transaction processing, and searching. They are fine-grained workloads and approximately treated as homogeneous. In contrast, heavy, long-term, and enumerable workloads served by HPC (High Performance Computing) data centers could be heterogeneous.
- The workload is generated through distributions or from trace files. The former is like Uniform Distribution, Poisson Distribution, Exponential Distinction, and Zipf distribution. The publicly available traces are like Google traces in [60], Amazon traces in [17], PlanetLab traces in [7], Real Parallel Workloads in [37], FaceBook trace in [61], and so on.

The workload placement directly affects the runtime of the data center thermal simulation. The placement approaches vary according to the research goals. A large number of data center thermal simulation studies, such as [4,31,51], place workload on servers randomly. A few studies choose the server straightforwardly, such as following the arrival sequence freely, called FCFS (First Come First Service) [7]. However, simulators also provide sophisticated workload placement simulations because the placement is critical to the servers' working state and further determines the amount of heat they generate. Alkharabsheh et al. and Shao et al. [5] dynamically place the workload according to server temperature, known as thermal-aware placement. Such placement can effectively reduce the number of hotspots.

In recent years, many studies have employed VMs (virtual machines) to assign workload, such as [21]. In data centers, VMs can dynamically allocate computing resources and automatically adjust resource allocation based on requirements to achieve load balancing. Thus, the simulator should support workloads running on VMs. For example, VanGilder et al. [9] schedule tasks for VMs based on their arrival status. Gao et al. [12] take a more advanced approach to VMs. VMs are runtime environments for incoming tasks and servers' workloads. The VM live migration provides more flexible resource management, as does thermal management. As a result, the ways to map the workload to the servers, such as simple, thermal-aware, and VM-supported manners, are the typical differences in modeling dynamic context.

Definition 3 (**Data Center Runtime Indicator**). *The dynamic context indicators show how to map the workload to the servers in a data center simulation. The three indicators are FIFO (First In First Out), temperature, and VM-supported. As their names, the first refers to the mapping rule as simple as random or FIFO; the second means mapping to the servers with lower temperatures; and the third is the mapping to the VMs on servers.*

Remark 3 (**Runtime Simulation**). *CloudSim has the best dynamic simulation ability in the runtime, while Fluent has the worst one. Table 5 shows the references of three runtime indicators on five simulators. Figure 5 shows the heatmap of three dynamic context indicators on five simulators. All simulators achieve the simple workload placement approach, which means that all simulators focus on the data center dynamic context indicators. Due to the limitations of CloudSim for thermal simulation, it does not have the thermal-aware workload placement. Compared with thermal-aware, applying the VM-based workload placement method is more promising. CloudSim has the highest usage in VM-supported because the VM is a fundamental function for a large-scale cloud data center.*

Table 5. References of three runtime indicators on five simulators.

	FIFO	Temperature	VM
Fluent	[8,10,15,48,51]	None	[9,12,21]
CloudSim	[4,6]	None	[6,35]
MatLab	[25,28]	[7,27,30,50]	None
6SigmaDC	[31,34]	[33]	None

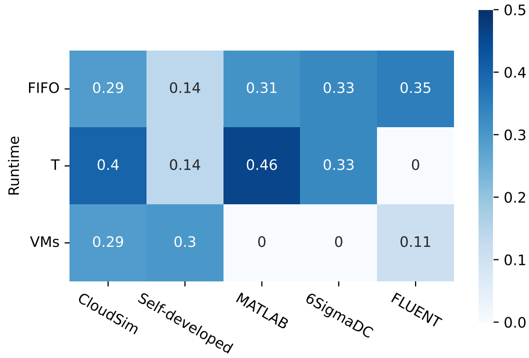

Figure 5. Heatmap of three dynamic context indicators on five simulators, ordered by the comprehensiveness of control method in runtime.

3.4. Power Model

The power model is the necessary function of thermal simulators for the green data center. All power models have four groups based on the characteristics of the proposed power and energy consumption models:

- The most simple power model is adding the power consumption of all parts in the data center together [25]. This model is suitable for simultaneous control of multiple facilities in data centers, such as servers and cooling facilities, where the power of each part should be considered independently.
- In data centers, the servers do not always remain in the active state, as servers also switch to idle mode. Therefore, the server power model has two parts: idle and active power [26]. This model could be applied to the simulation where the research object is the server, the computing power of the data center should be accurately calculated, and the power of other devices could be derived from the server power.
- The regression models capture the fixed or idle power consumption and the dynamic power consumption with changing activity across the functional units of the servers [26]. An approximate power model is drawn by fitting the dynamic operation characteristic curve of the server. The power model reflects the changing trend and conforms to a specific form, which is suitable for the scenarios where the research object is the data center environment.
- The utilization-based power model means the power-consuming and maximum power are in the proportion of resource utilization. Since CPU is the most power-consuming component in a data center, the utilization-based power models leverage CPU utilization as their metric in modeling the server power consumption [6].

As a result, the ways to calculate the power, such as accumulation, max-and-min, regression, and utilization, are the typical differences in the power model.

Definition 4 (**Power Indicator**)**.** *The power indicators show the modeling of power consumption, according to the four groups above. The four indicators are accumulation, active-idle, regression, and utilization, respectively. As their names suggest, accumulation means adding the power of every component, active-idle means considering the maximum and minimum power of servers, regression* *means drawing power functions through experimental data, and utilization means representing power as a resource utilization function.*

Remark 4 (**Power Model Simulation**). *CloudSim has the best power model simulation ability, while Fluent has the worst one. Table 6 shows the references of four power model indicators on five simulators. Figure 6 shows the heatmap of four power indicators on five simulators. Fluent is very weak for data center power simulation; nearly 80% of applications do not mention the power model. 6SigmaDC provides a built-in accumulation model, and some studies extend it to utilization models. However, there are only a few power-related studies with Fluent and 6SigmaDC. CloudSim provides a flexible power component that supports all power models, from simple to complex ones, so that the power models are well addressed in CloudSim. The open-source and rich programming features also contribute to the versatility of power models. The same as MatLab, in which the active-idle model is widely adopted for its simplicity. Most self-developed simulators do not involve power models, but if they do, both four models can easily be implemented by programming.*

Table 6. References of four power model indicators on five simulators.

	Accumulate	Active-Idle	Regress	Utilize
Fluent	[5,43,49]	[13,20,21]	None	[23]
CloudSim	[6,17]	[4]	[18]	[6,35]
MatLab	[25]	[21,26–30]	None	[30,50]
6SigmaDC	[34]	None	None	[31]
Self-developed	[37,45]	None	[47]	[38]

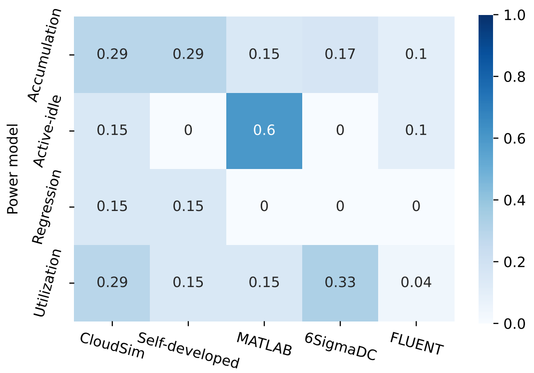

Figure 6. Heatmap of four power indicators on five simulators, ordered by the comprehensiveness of power model.

3.5. Thermal Model

Thermal simulation for green data centers mainly provides a closed cycle in which heat flows from servers to servers or back to cooling facilities through the data center's aisles. The thermal model contains the thermal generation model and the thermal propagation model. The thermal generation models have three types:

- The thermal model directly transforms the server's power into its heat according to a specific function, as discussed in [4,31]. It is the most fundamental one in heat simulation. The simulation details are available in [25].
- An RC (Resistance-Capacitance) thermal model considers the relationship between heat transfer to the ambient environment and the RC circuit's electrical phenomenon. The model was adopted by Ilager et al. [6] to estimate the CPU temperature, and Pierson et al. [38] utilized the model, taking into account both spatial and temporal temperature behaviors.

- An empirical model predicts the heat generation of data centers based on historical data or thermal behavior, such as [40]. The prediction is not only for simulation but also for improving the performance of data centers and balancing the heat distribution. The thermal propagation models also have three types.
- The aerodynamic formula [30] is an effective model for calculating thermal propagation with reasonable accuracy. For example, Damme et al. [30] choose an effective heat model for real-time controlling the data center temperature. Zhao et al. [50] adopt the fast feedback conditioning model to enhance data center performance. Simon et al. [33] argue that the aerodynamic model performs better than the directly transformed model.
- Fluid dynamics, or CFD (Computational Fluid Dynamics), is an efficient and accurate technology for modeling how heat is transferred and exchanged thermally [48]. Studies not only use but also improve the CFD model in the simulators. For example, Tian et al. [10] improved the CFD model to make the solution faster, which reduces the complexity of the solution process on the model. Alimohammadi et al. [11] trim the CFD model to fit data center thermal simulation.
- Applying machine learning to the thermal propagation model is a new emerging technology. A learning-based model can dynamically change with the data center. For example, Mirhosein-iNejad et al. [38] proposed a holistic thermal model, namely the adoptive learning-based model (ALTM), which can predicate the temperature of the critical thermal zones using DC operational variables as inputs and outputs.

As a result, users choose different thermal generation models depending on the study's objective. However, different heat generation models do not affect the thermal simulation of data centers. The ways to similar the thermal propagation model, such as thermodynamics, CFD, and machine learning, are the typical differences in modeling thermal context.

Definition 5 (**Thermal Indicator**)**.** *The thermal indicators show the underlying theories and models of simulation on thermal transferring and exchange. The three indicators are thermodynamics, fluid dynamics, and learning-based. As their names, they refer to the corresponding thermal propagation model mentioned above.*

Remark 5 (**Thermal Model Simulation**)**.** *Fluent has the best thermal model simulation ability, while CloudSim has the worst one. Table 7 shows the references of three thermal model indicators on five simulators. Figure 7 shows the heatmap of three thermal indicators on five simulators. The Fluent, MatLab, and 6SigmaDC cover all the thermal indicators. CFD is a heavy model. It is costly to develop and execute a CFD simulation. Fluent and 6SigmaDC, as commercial software, are born for thermal simulation, especially in support of CFD. MatLab is a programming environment with plenty of components supporting various thermal models. CFD is too heavy for CloudSim and self-developed simulators. Besides, thermal propagation in a large-scale data center, simulated by CloudSim, is too large for CFD mode to be solved. As for the new and promising model, learning-based is the most frequently applied in CloudSim. At last, thermodynamics is very lightweight and fundamental and should be supported by all simulators.*

Table 7. References of three thermal model indicators on five simulators.

	Thermodynamics	Fluid-Dynamics	Learning-Based
Fluent	[9]	[8,10,12,15,21,48,51]	[15,20,24]
CloudSim	None	None	[18,36]
MatLab	[27,30,50]	[21]	[16]
6SigmaDC	[33]	[32,44]	[33]
Self-developed	[46]	None	[38]

Figure 7. Heatmap of three thermal indicators on five simulators, ordered by the comprehensiveness of thermal model.

3.6. Answers to RQ1

Yes. In general, state-of-the-art simulators have the same core and abstract functions as thermal simulators for green data centers. The survey shows that all simulator functions of data centers fall within the four dimensions: modeling static context and dynamic runtime, as well as simulating heating and cooling. However, not every simulator supports all indicators of each dimension. According to the remarks of each section, both Fluent and MatLab are well-implemented when simulating small and middle-scale data centers, but CloudSim can operate when simulating large-scale data centers. Fluent does best in simulating cooling facilities and the coverage of thermal models, while CloudSim has the weakest ability. However, CloudSim has the most robust simulation ability when simulating data center systems, such as runtime and power models, while all the others have shortcomings. Obviously, Fluent performs well in terms of versatility; MatLab and CloudSim are slightly less impressive; and self-developed versatility is very uneven. Section 5 shows a quantitative comparison between them.

4. Dexterity

This section studies the dexterity of thermal simulators in data centers. Dexterity is the ability to perform a complex action quickly and skillfully with the hands. We liken the “action” to data center simulation and “hands” to simulators. The dexterity could be measured through the feeling of “how easily a researcher uses a simulator”. Such feelings emerge in a long-term evaluation process that includes learning the simulator from the beginning, developing it while building the simulation environment, executing various experimental conditions while simulating the data center, customizing different tests with various components or databases, and visualizing the final results. Sometimes researchers use cheaper or low-resource software due to budget or hardware limitations. It is also considered in this section.

Therefore, this section studies dexterity from the above five aspects. It calls them the five dimensions: easy to learn, easy to develop, easy to execute, easy to customize, and easy to visualize. And then, the following five subsections discuss each of them, respectively. For a subsection, first it surveys the observations about the dexterity of typical simulators. Second, because distinct observations for each simulator are the key to distinguishing and evaluating the simulators, it abstracts these observations and defines the indicators. Third, it qualitatively analyzes the five typical simulators on these indicators, for example, whether a simulator supports the indicator, and then shows the results in a comparative table. Finally, it discusses the findings (*remarks*) on the dimension according to the table. For example, the advantages and disadvantages of different simulators on these indicators. After reviewing all dimensions, the last subsection answers the RQ2.

4.1. Easy to Learn

When learning to use a new simulator, the first question a researcher faces is whether the simulator is open-sourced. Some users may directly reject closed-source simulators.

A closed-source simulator does not allow updating algorithms. Thus, developers have to blindly trust the implementation because they cannot access the source code. In contrast, open source helps ease the development and verification process. Open-sourced and self-developed simulators have an open implementation that lets developers freely control the case-building process. Moreover, an open-sourced simulator helps gain more knowledge of the algorithms, know how they work in different scenarios, and encourage further improvements.

Secondly, the resources that introduce how to use the simulator also significantly impact the ease of learning a simulator. Moreover, the programming language required for the simulator is a significant challenge for users. The most widely used programming languages today are Python, C++, and Java. Python has a wide range of prospects, which is the best way to get started. It is very suitable for scripting. Both C++ and Java are suitable for large-scale operations. Surveys show that 80 percent of programming professionals prefer Java [62]. This means that C++ and Java are better at simulating data centers.

Definition 6 (**Learning Indicator**). *The learning indicators show how easy it is to learn to use a simulator. The three indicators are openness, programming, and resource. As their names, the first is whether the simulator is open-sourced. The second represents the programming language as the interface between programmers and a simulator. The second represents the availability of learning resources, training courses, and documentation that may influence the learning process, such as a well-documented solution alongside plenty of online resources and tutorials.*

Remark 6 (**Easy to Learn**). *CloudSim is the easiest to learn. Table 8 shows the comparison of three learning indicators on five simulators. Although thermal simulators are built and developed to simulate the data center, the ways to reach the goals differ from one simulator to the others due to their diverse nature. For example, Fluent is an application that uses C/C++. Moreover, as commercial software, Fluent has enough online learning resources and training courses to reduce the learning effort. MatLab, as a mathematical simulator, prefers to model every part of the data center and reflect the performance through solving functions. 6SigmaDC is designed to simulate data centers using C/C++. However, it is commercial, closed-source software, and lacks learning resources. On the contrary, CloudSim is an open-source data center simulator with Java programming language and full of shared experiences. Some self-developed simulators aim for toolkits or standalone models with Python programming language. They have very few documents except the related papers, making them harder to learn.*

Table 8. Comparison of three learning indicators on five simulators.

	Openness	Program	Resource
Fluent	Closed	C/C++	Plenty of online resources and training courses
CloudSim	Open	Java	Full documentation and shared experiences
MatLab	Closed	C/C++	Online resources and full documentation
6SigmaDC	Closed	C/C++	Less
Self-developed	Open	Java/Python	No sources

4.2. Easy to Develop

Developing the simulation case is the initial purpose and core task of each simulator. Users all prefer an easy case-building process through a simulator. Fluent is for any Computational Fluid Dynamics (CFD) system, including data centers. 6SigmaDC is a commercial tool specially designed to simulate thermal data centers, but it is not popular and mainly focuses on static room-scale and hardware-scale simulation. CloudSim is the special simulator for data center runtime but lacks the thermal model. MatLab is for running mathematical models, but users should develop the models themselves. In other words, CloudSim supports running servers and dynamic systems but not CFD. Fluent supports CFD but does not run as servers. MatLab manually models both the CFD and running servers. Therefore, a simulator has unique features that distinguish it from the others, and measuring the developing dexterity is also related to the simulation task itself. Some tasks might be easier to develop on a specific simulator than others. As a result, those criteria indicate developing dexterity.

Definition 7 (Developing Indicator). *The developing indicators show how dexterous a simulator is to build a simulation case. The three indicators are main-focus, runtime-case, and system. As their names, the first represents the case mainly supported by a simulator. The second shows how complex a simulation case could be modeled. The third means what kind of software a simulator looks like.*

Remark 7 (Easy to Develop). *CloudSim is the easiest to develop, like self-developed. Table 9 shows the comparison of three developing indicators on five simulators. Commercial Fluent and 6SigmaDC are designed for CFD computation, but the former focuses on the general case and the latter on the data center infrastructure. However, none support the detailed simulation of data center runtime state, such as workload placement and VM migration. Developers would put more effort into it. In contrast, open-sourced CloudSim is designed to simulate data center runtime but lacks full support for simulating the thermal environment. Self-developed simulators support both the runtime and the thermal case of a data center. However, they cannot be as detailed as the commercial ones. MatLab is a commercial IDE (Integrated Development Environment). It is more open than others because it provides many functions, libraries, and components. Building a detailed case for both runtime and the thermal case of a data center is feasible but time-consuming.*

Table 9. Comparison of three developing indicators on five simulators.

	Main Focus	Runtime Case	System
Fluent	General CFD	Simple	Application
CloudSim	Data Center Runtime	Detailed	Toolkit and Framework
MatLab	General	Moderate	IDE
6SigmaDC	CFD and Hardware	Simple	Application
Self-developed	Thermal model and Data Center Runtime	Detailed	Toolkit and Framework

4.3. Easy to Execute

Experimental studies mainly mention the hardware environment, which is the required resource for running the simulation [7]. These resources represent the execution dexterity, namely, how easily the simulation case is executed. More resource requirements mean more "difficulty" in executing the case, especially when requiring GPUs. Besides, resources could be hardware, software, and platforms. We call it"integration" if a simulator needs third-party software or frameworks to execute cases. Finally, operating the system as a "platform resource" is another concern for execution dexterity. As a result, the above three are indicators of execution dexterity.

Definition 8 (Execution Indicator). *The execution indicators show the environment and the communication methods required to execute the simulation case in the runtime. The three indicators are hardware, integrated software, and operating system. As their names, the first means the hardware context. The second represents the integrated third-party software, such as components, frameworks, packages, environments, and databases. The third refers to the required operating system.*

Remark 8 (Easy to Execute). *CloudSim is the second place only to MatLab in execution indicators. Table 10 shows the comparison of three execution indicators on five simulators. Fluent and 6SigmaDC, as commercial simulators, require heavy resources to compute the CFD, which is even better* *on GPUs. Although Fluent does not need any integration, it is considered to be the most difficult for execution since it requires vast computing resources. In contrast, CloudSim is very lightweight on computing resources, but it requires package integration, such as those in [4,6]. MatLab has a decent execution dexterity similar to CloudSim. Self-developed simulators have various resource requirements except for the hardware. Most of them can run on commercial computers or servers, and the execution dexterity for the developers is minor.*

Table 10. Comparison of three execution indicators on five simulators.

	Hardware	Integrated Software	Operating System
Fluent	Heavy, GPU	Not Needed	Windows and Linux
CloudSim	Light, no GPU	Needed	All
MatLab	Light, no GPU	Not Needed	All
6SigmaDC	Heavy, GPU	Not Needed	Windows
Self-developed	Light, no GPU	Needed	All

4.4. Easy to Customize

Customizing a simulator is the ability to be flexible and changeable in its functions. Customization enables users to perform more complex simulations according to their demands. Users may have the requirement of customization from the following three aspects:

- Function extension is the ability of the simulator to add extra functions and features by the use of external packages or programming languages, such as experiments in [45];
- Adaptability is the ability of the simulator to accept changes in runtime environments conveniently. For example, it can reload or refine the modified case without stopping running, such as experiments in.
- Meshing is the ability of the simulator, such as Fluent, to properly define the physical shape of the object through the meshing process. The process breaks down continuous geometry or geometric space into thousands of shapes [48].

As a result, the above three are indicators of customization dexterity.

Definition 9 (Customization Indicator). *The customization indicators show how easy to customize a simulator according to the user's requirements. The three indicators are extensibility, adaptability, and meshing. As their names, the first represents the capability of introducing new functions. The second represents the capability of adapting to a changeable runtime environment. The third means the capability of breaking down continuous geometry or geometric space of the simulated object into thousands of shapes.*

Remark 9 (Easy to Customize). *CloudSim and MatLab are both the easiest to customize. Table 11 shows the comparison of three customization indicators on five simulators. In terms of extensibility, Fluent supports extension through UFD (User Defined Function), which enables the user to enhance the simulation functions. For example, Jafarizadeh et al. [8] reduce the server heat load to 100 W when its average server inlet temperature exceeds 40 °C. CloudSim supports the extension capability better than Fluent because it is an open-source framework. It naturally does not contain thermal simulation. However, users have extended various thermal models [6]. For example, Ali et al. [6] extend the delay overheads associated with server power mode to simulate the thermal management of a data center. MatLab has better extensibility because it provides the programming environment for customized models. Self-developed simulators are designed for particular purposes, but extensibility may not be one of them. 6SigmaDC's extensibility is unknown because we failed to find the documents about it. In terms of adaptability, Fluent and CloudSim need to stop the running simulation and accept changes. In contrast, running cases in MatLab and several self-developed simulators can be modified online. The primary importance of meshing in CFD lies in solving* *the governing partial differential equations in the allocated cell. Thus, Fluent and 6SigmaDC are famous for meshing, and MatLab also supports it through plugins, but the other two fail.*

Table 11. Comparison of three customization indicators on five simulators.

	Extensibility	**Adaptability**	**Meshing**
Fluent	Good	Good	Best
CloudSim	Best	Better	Normal
MatLab	Better	Best	Normal
6SigmaDC	Unknown	Normal	Good
Self-developed	Normal	Normal	None

4.5. Easy to Visualize

Visualizing the results is an essential part of the simulation process. Typical thermal simulators have proper visualization capability, but their underlying techniques may differ. The dexterity of visualization relates to the adopted techniques, such as remote visualization, 3D visualization, animated visualization, and rendering. All simulators can show the results in 2-D or 3-D static visuals; however, the animations are better for showing the changes dynamically, and rendering technology gives more impressive visualization. As a result, these techniques are indicators of visualization dexterity.

Definition 10 (**Visualization Indicator**)**.** *The visualization indicators show how easy it is to visualize the simulation results using the simulator. The three indicators are remote, animation, and rendering. As their names suggest, the first is a simulator's ability to connect to a remote server and visualize the output. The second is the ability of a simulator to create animation and show the results dynamically. The third is the ability to render the graphics through the OpenGL API. OpenGL is a cross-platform application programming interface (API) for rendering 2D and 3D vector graphics. The API is typically used to interact with a GPU to achieve hardware-accelerated rendering.*

Remark 10 (**Easy to Visualize**)**.** *Fluent is the easiest to visualize. Table 12 shows the comparison of three visualization indicators on five simulators. Fluent supports animation and OpenGL visualization. As a commercial solution, it provides the user with a remote visualization option where the user can take advantage of powerful servers. In contrast, another simulator, 6SigmaDC, also a commercial one, does not support sophisticated visualization, probably because it is only for data center simulation. Such simulation may not require complicated animation and rendering. Self-developed simulators could technologically support the three indicators by taking full advantage of programming languages, such as Python; however, few of them put effort into visualization. The developers typically offer visualization in static graphs by integrating third-party components. CloudSim with Java has the same situation as the self-developed. As a programming environment and language, MatLab provides more support in animation and rendering because such support may also benefit other applications.*

Table 12. Comparison of three visualization indicators on five simulators.

	Remote	**Animation**	**Rendering**
Fluent	Supported	Supported	Supported
CloudSim	Not Supported	Not Supported	Not Supported
MatLab	Not Supported	Supported	Supported
6SigmaDC	Supported	Not Supported	Not Supported
Self-developed	Not Supported	Not Supported	Not Supported

4.6. Answers to RQ2

The above five subsections describe "how easy to apply these simulators in thermal management research for green data centers". Tables 8–12 show that the simulators cover the most critical features of usability. These simulators are generally easy to handle, even if they have different ways to support these indicators, or a few are not yet supported. CloudSim, a popular open-source platform, has strong usability in all dimensions. Thus, it has the most abundant learning resources. CloudSim reaches the simulation goals more easily than other simulators. MatLab is the most straightforward for users to operate. It is the easiest simulator to execute. Fluent, a famous commercial software, performs well on customization and visualization. It can show rich simulation visualization, but it has poor usability on the other three dimensions. The dexterity of self-developed and 6SigmaDC are the worst. In dexterity, CloudSim, MatLab, and Fluent all perform best in different dimensions. It is difficult to compare which is the most dexterous. Therefore, Section 5 shows a quantitative comparison between them.

5. Comparison

This section compares the versatility and dexterity of data center simulators. This paper considers both objective and subjective factors. It cannot be concluded by a simple test. Therefore, we choose the comprehensive analysis method to compare the simulator performance. On the one hand, it calculates the versatility matrix of a simulator on the five versatility dimensions in Tables 3–7, Section 3. The indicators of a dimension are aggregated to a value through PCA (Principal Component Analysis). It is the most widely used data dimensionality reduction algorithm. Table 13 summarizes the versatility matrix, whose elements quantitatively evaluate the versatility dimensions of a simulator.

Table 13. Versatility matrix on five simulators.

	Context	Cooling	Runtime	Power	Thermal
Fluent	0.633	0.636	0.528	0.166	0.734
CloudSim	0.585	0.41	0.713	0.564	0.428
MatLab	0.487	0.703	0.637	0.337	0.728
6SigmaDC	0.626	0.52	0.533	0.22	0.687
Self	0.35	0.43	0.673	0.38	0.53

On the other hand, this section calculates the dexterity matrix of a simulator on the five dexterity dimensions in Tables 8–12, Section 4. The indicators of a dimension are aggregated to a value through FCE (Fuzzy Comprehensive Evaluation). It is widely used in multi-attribute decision-making problems. It comprehensively evaluates multiple factors and chooses the best solution. The five indexes are defined through the expert evaluation method. For each dexterity indicator, experts rank simulators according to their performance in three sub-indicators, with a five-point Likert scale, and then normalize to the [0, 1] range. The experts give the weights of five dimensions as [0.24, 0.12, 0.28, 0.16, 0.2]. Table 14 summarizes the dexterity matrix whose elements quantitatively evaluate the dexterity dimensions of a simulator.

Table 14. Dexterity matrix on five simulators.

	Learn	Deploy	Execute	Customize	Visualize
Fluent	0.4	0.2	0.4	1	1
CloudSim	1	0.6	0.8	0.8	0.6
MatLab	0.8	0.4	1	0.6	0.8
6SigmaDC	0.2	0.8	0.2	0.4	0.4
Self	0.6	1	0.6	0.2	0.2

5.1. Results

Two radar charts in Figure 8 compare the versatility and dexterity matrices of five simulators, where red, blue, yellow, green, and dot-lined represent Fluent, CloudSim, MatLab, 6SigmaDC, and self-developed. The self-developed simulator, marked as an unfilled dot, is not a particular simulator but a combination of many experimental simulators.

Remark 11 (**Versatility Comparison**). *MatLab is the most balanced simulator; however, it shows no best on each indicator. Fluent is the best at cooling and thermal models. It also provides fine-grained context modeling to implement detailed and precise thermal simulation; however, it is not designed for data centers only, so it shows the weaknesses in data center runtime, such as workload and power model. 6SigmaDC is similar to Fluent but designed for data centers only; however, its main focus is still CFD and fails to show advances in runtime modeling. The 6SigmaDC's polygon is like a shrink one of Fluent's and the smallest in the radar charts because 6SigmaDC is not popular in research. Self-developed is a "choice" made by the communities because these simulators focus on the functions that are frequently required but not well addressed by the other four simulators. Its advances show the expectation, and its weakness shows the unconcerned. CloudSim's polygon is the closest to self-developed's. It shows the advances in the workload and power model, the moderate in the data center context, and the weakness in the cooling and thermal model. Compared with the self-developed, CloudSim may need an enhancement on thermal and cooling models, but not necessarily to be as remarkable as Fluent and 6SigmaDC.*

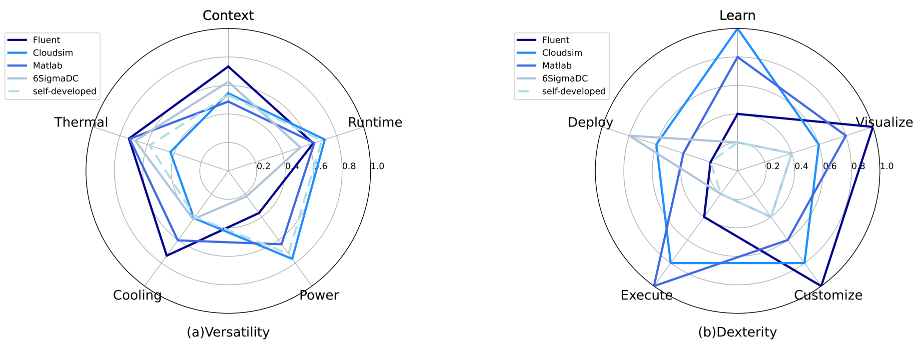

Figure 8. Radar chart of the versatility and dexterity on five simulators.

Remark 12 (**Dexterity Comparison**). *MatLab is also not the best but the most balanced simulator in dexterity, thanks to its IDE feature and programming interface. Fluent is not a dexterous simulator. Thanks to the sophisticated CFD calculation and rendering, it has the best customization and visualization. However, the other three indicators are poor. Compared with Fluent, 6SigmaDC is easy to develop simulation cases because it is designed for data center simulation only; except for this indicator, it is even more difficult to use because it is not popular in research. Self-developed is the worst one in dexterity. Dexterity cannot be guaranteed accordingly when combining many practical simulators makes it versatile. CloudSim is the best in dexterity. It also adopts programming interfaces compared with MatLab. However, CloudSim is designed for data centers similar only,* *and MatLab is a general IDE. These are reasons why CloudSim's polygon is like the enlargement of MatLab's.*

5.2. Answers to RQ3

The answer to the first question in RQ3 is negative. Take Fluent as an example; it is not very applicable for the thermal simulation of green data centers because it aims for thermal functions rather than the datacenter functions. The discussion in Sections 3 and 4 backs our opinion well. Fluent performs the best in the CFD and thermal simulation. When employing Fluent for thermal simulation in data centers, operators must model the physical layouts of the data center and import the geometry file into Fluent for preprocessing, meshing, and running simulation. The complex pre- and post-processing software makes Fluent neither a simple simulator to grasp nor very convenient for a larger-scale simulation and dynamic thermal environment simulation of data centers. In the green datacenter simulation, studies may prefer a full-fledged simulator for dynamic datacenter runtime, such as servers, virtual machines, workload, and scheduling, as well as a moderate, even holistic thermal simulation. MatLab and CloudSim have almost the same versatility as Fluent but much better dexterity. They could be the alternative. They are expected a group of enhancements to reach the proper versatility with maximum dexterity. The next section draws five expected enhancements of current simulators and explains the second question in RQ3 as the outlook.

6. Outlooks

The ideal data center thermal simulator should be fully functional and easy to apply. This section proposes five outlooks for the simulators in the future.

6.1. Resources Highlighted Simulation

The trend of thermal studies on data centers switches from the statistical context design to the dynamic runtime adjustment. Studies on context design can avoid defective thermal design in data centers. However, they cannot optimize according to the dynamic data center runtime, such as the workload, resource usage, and power. If users change the resource usage distribution of servers, the thermal distribution is also altered accordingly. A data center simulator should fully highlight resource management functions to support such a trend. Resource management is a vital issue in data center studies, and particular simulators, such as CloudSim, SimGrid [63], and iFogSim [64], have preliminarily integrated the issue.

6.2. Thermal Light-Weighted Simulation

The thermal simulation is required to reflect the thermal context of each part of data centers, how it changes with the runtime, and what the optimization effect is. Thermal distribution is not necessarily modeled and simulated in a fine granularity on time and space. For example, users cannot expect a resource scheduler to significantly affect thermal distribution immediately because both runtime and thermal conditions are not stabilized. Complex CFD calculations with powerful computing resources are usually required to simulate detailed and real-time thermal dynamics states, bringing huge costs on dexterity and hardware investment. However, the simulation results may not practically benefit the studies. Therefore, the ideal simulator would prefer a lightweight thermal simulation with coarse granularity on both time and space, easy modeling and execution, and less hardware requirement.

6.3. AI-Integrated Simulation

Artificial intelligence (AI) has already altered the world and raised significant progress for society, the economy, and governance. In many fields, AI empowers people, communities, machines, and computers to work together and make significant improvements, the same as the AI-empowered thermal management for green data centers. For example, Li et al. [40] used a Neural Network to predict the temperature inside the data center, which can limit the temperature in advance to achieve a good thermal balance. AI can easily achieve the challenge goals. Therefore, the ideal simulator should integrate and incorporate AI. For example, it could support the popular AI platform and support the deployment and execution environment for AI models such as Deep Neural Networks and Deep Reinforcement Learning. AI-integrated simulators enable researchers to run their AI-empowered optimization on the simulation case.

6.4. Larger-Scale Heterogeneous and Cross-Regional Data Centers Simulation

Data centers have been known for their scale ever since, and now cross-regional data centers containing thousands of servers are the main focus of the research. The new focus brings two challenges to data center simulation: heterogeneity and geo-distribution. On the one hand, a heterogeneous data center may bring all kinds of computing resources together: x86s and GPUs, FPGAs, and even other processor types like ARM processors. Their computing, energy, and thermal features vary. A simulator would face the complexity brought by heterogeneity. On the other hand, cross-regional data centers, being connected and serving as a union, bring several benefits: providing services to the cross-region areas efficiently, storing data with cross-region replication to ensure data availability, such as Azure Geo-Redundant Storage (GRS) replications, and fully utilizing and cooperating regional resources and requirements, such as energies, computing power, data, and requests. The "Eastern Data Western Computing" plan in China is a good example of the above scenario. To this end, the next-generation data center thermal simulator should enable the simulation of heterogeneous and cross-regional data centers on a larger scale.

6.5. Thermal Extensions on CloudSim

As a famous commercial software, Fluent has good versatility and acceptable dexterity. In contrast, CloudSim, a popular open-source platform, has strong potential to be better. CloudSim is more applicable to simulating the runtime of a larger-scale data center, such as task scheduling, workload placement, server state, VM migration, and resource provision, which are frequently investigated in state-of-the-art studies. Fluent has a mature and sophisticated thermal model that is difficult to reproduce. In contrast, the thermal model integrated into CloudSim is primitive. However, a lightweight thermal model may cover most thermal simulations, especially studies oriented to data center runtime. CloudSim is also more open. Thus, the attributes of an ideal data center thermal simulator, which have been analyzed in previous sections, could possibly be extended to the CloudSim. To this end, the most promising CloudSim has considerable potential to be an ideal simulator after extension, especially on thermal models.

7. Conclusions

This paper reviews the various simulators for data center thermal simulation over the past decade. It mainly compares and evaluates existing data center simulators from two aspects: Do state-of-the-art simulators have the same functions as thermal simulators for green data centers? What are these functions designed for? How easy is it to apply these simulators in thermal management research for green data centers? The analysis and experimental results show that the most widely used data center simulators, Fluent, are versatile but still need improvement in data center scenarios. CloudSim has the expected potential to be an ideal simulator if it is well extended. Finally, the paper gives the characteristics of the ideal data center thermal simulator and explains the improvement strategy.

Based on the answers of RQ1, RQ2, and RQ3, relevant researchers can quickly choose the right simulator. Fluent can provide accurate static thermal simulation. Cloudsim can provide dynamic thermal simulation.

Based on the "remarks" mentioned in this paper, the simulators could be extended in multiple directions. To improve versatility, researchers should try to dig out new functions that have not been covered by existing simulators so far. These new functions could be a cross-regional green awareness scheduler on geographically distributed data centers, a sophisticated battery system that improves energy efficiency, and a multifarious energy model that supports more types of renewable energy. We expect a general-purpose simulator with our metrics as a guideline for the studies of the thermal-efficient data centers.

Author Contributions: D.L. and J.S.; methodology, D.L.; software, D.L.; validation, J.S., H.L. and J.J.; formal analysis, D.L.; investigation, J.S.; resources, J.J.; data curation, H.L.; writing—original draft preparation, D.L.; writing—review and editing, D.L.; visualization, J.S.; supervision, H.L.; project administration, H.L.; funding acquisition, J.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China under Grant No. 62162050, No. 62302086.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Hanna, B.N.; Abdalla Abou-Jaoude, N.G.P.T.; Lohse, C. Navigating Economies of Scale and Multiples for Nuclear-Powered Data Centers and Other Applications with High Service Availability Needs. *Energies* **2024**, *17*, 5073. [CrossRef]
2. Son, A.Y.; Son, D.; Shin, Y.R.; Huh, E.N. Resource-Aware Migration Scheme for QoS in Cloud Datacenter. In *Advances in Computer Science and Ubiquitous Computing*; Park, J.J., Park, D.S., Jeong, Y.S., Pan, Y., Eds.; Springer: Singapore, 2020; pp. 85–90.
3. Wang, S.; Tu, R.; Chen, X.; Yang, X.; Jia, K. Thermal performance analyses and optimization of data center centralized-cooling system. *Appl. Therm. Eng.* **2023**, *222*, 119817. [CrossRef]
4. Pierson, J.M.; Stolf, P.; Sun, H.; Casanova, H. MILP formulations for spatio-temporal thermal-aware scheduling in Cloud and HPC datacenters. *Clust. Comput.* **2020**, *23*, 421–439. [CrossRef]
5. Shao, Z.; Islam, M.A.; Ren, S. Heat Behind the Meter: A Hidden Threat of Thermal Attacks in Edge Colocation Data Centers. In Proceedings of the 2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA), Seoul, Republic of Korea, 27 February–3 March 2021; pp. 318–331.
6. Ali, A.; Ozkasap, O. Workload Management in Distributed Data Centers: Thermal and Spatial Awareness. In Proceedings of the 2016 IEEE International Conference on Smart Cloud (IEEE SmartCloud), New York, NY, USA, 18–20 November 2016; pp. 158–163.
7. Li, D.; Zhang, Y.; Song, J.; Liu, H.; Jiang, J. Energy Saving with Zero Hot Spots: A Novel Power Control Approach for Sustainable and Stable Data Centers. *Sustainability* **2022**, *14*, 9005. [CrossRef]
8. Jafarizadeh, M.; Tsai, P.J.; Zheng, R. Thermal Piloting: A Novel Approach for Sensor Localization in Data Center Monitoring. In Proceedings of the 2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), Santorini Island, Greece, 29–31 May 2019; pp. 17–24.
9. VanGilder, J.; Vaishnani, Y.; Tian, W.; Condor, M. A Compact Rack Model for Data Center CFD Modeling. In Proceedings of the 2020 19th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm), Orlando, FL, USA, 21–23 July 2020; pp. 350–356.
10. Tian, W.; VanGilder, J.; Condor, M.; Han, X.; Zuo, W. An Accurate Fast Fluid Dynamics Model for Data Center Applications. In Proceedings of the 2019 18th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm), Las Vegas, NV, USA, 28–31 May 2019; pp. 1275–1281.
11. Alimohammadi, S.; McEvoy, J.; Delauré, Y.; Persoons, T. Benchmarking the Application of Detached Eddy Simulation Techniques in Data Center Server Flow Modelling using Stereoscopic Particle Image Velocimetry. In Proceedings of the 2018 24rd International Workshop on Thermal Investigations of ICs and Systems (THERMINIC), Stockholm, Sweden, 26–28 September 2018; pp. 1–4.
12. Gao, X.; Xu, Z.; Wang, H.; Li, L.; Wang, X. Why "Some" Like It Hot Too: Thermal Attack on Data Centers. In Proceedings of the 2017 ACM SIGMETRICS/International Conference on Measurement and Modeling of Computer Systems, Urbana-Champaign, IL, USA, 5–9 June 2017; pp. 23–24.
13. Sun, H.; Stolf, P.; Pierson, J.M. Spatio-temporal thermal-aware scheduling for homogeneous high-performance computing datacenters. *Future Gener. Comput. Syst.* **2017**, *71*, 157–170. [CrossRef]
14. Hu, B.; Lin, C.X.; Patel, D.; Joshi, Y.; Vangilder, J.; Seymour, M. A Comprehensive CFD Study of Tile Flow Rate Distribution in a Compact Data Center Laboratory. In Proceedings of the 2020 19th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm), Orlando, FL, USA, 21–23 July 2020; pp. 374–382.
15. Athavale, J.; Joshi, Y.; Yoda, M. Artificial Neural Network Based Prediction of Temperature and Flow Profile in Data Centers. In Proceedings of the 2018 17th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm), San Diego, CA, USA, 29 May–1 June 2018; pp. 871–880.
16. MirhoseiniNejad, S.; Garcia, F.M.; Badawy, G.; Down, D.G. ALTM: Adaptive learning-based thermal model for temperature predictions in data centers. In Proceedings of the 2019 IEEE Sustainability Through ICT Summit (StICT), Montréal, QC, Canada, 18–19 June 2019; pp. 1–6.
17. Ilager, S.; Ramamohanarao, K.; Buyya, R. ETAS: Energy and thermal-aware dynamic virtual machine consolidation in cloud data center with proactive hotspot mitigation. *Concurr. Comput. Pract. Exp.* **2019**, *31*, e5221. [CrossRef]

18. Ilager, S.; Ramamohanarao, K.; Buyya, R. Thermal Prediction for Efficient Energy Management of Clouds Using Machine Learning. *IEEE Trans. Parallel Distrib. Syst.* **2021**, *32*, 1044–1056. [CrossRef]
19. Sakanova, A.; Alimohammadi, S.; McEvoy, J.; Persoons, T. Hybrid-Cooled Data Center Server Layout Optimization for Air-Side Heat Recovery. In Proceedings of the 2018 24rd International Workshop on Thermal Investigations of ICs and Systems (THERMINIC), Stockholm, Sweden, 26–28 September 2018; pp. 1–4.
20. Wang, Q.; Song, M.; Fang, Q.; Wang, J. Thermal-aware Flow Field Optimization for Energy Saving of Data Centers. In Proceedings of the 2018 Annual American Control Conference, ACC 2018, Milwaukee, WI, USA, 27–29 June 2018; IEEE: New York, NY, USA, 2018; pp. 3744–3749.
21. Fang, Q.; Wang, J.; Gong, Q.; Song, M. Thermal-Aware Energy Management of an HPC Data Center via Two-Time-Scale Control. *IEEE Trans. Ind. Inform.* **2017**, *13*, 2260–2269. [CrossRef]
22. Chen, H.; Chen, T.; Li, D.; Qi, Y.; Zhong, M.; Li, Y.; Huo, X.; Ding, Y. Simulation Investigation of Airflow Distribution in Data Center using the Field Synergy Principle. In Proceedings of the 2021 IEEE 5th Conference on Energy Internet and Energy System Integration (EI2), Taiyuan, China, 22–24 October 2021; pp. 4134–4139.
23. Matsuoka, M.; Matsuda, K.; Kubo, H. Liquid immersion cooling technology with natural convection in data center. In Proceedings of the 2017 IEEE 6th International Conference on Cloud Networking (CloudNet), Prague, Czech Republic, 25–27 September 2017; pp. 1–7.
24. Yuksel, A.; Demetriou, D.W.; Hu, Y.; Mahaney, V. Thermal Design of Portable Modular High Performance Computing Data Centers. In Proceedings of the 2021 20th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (iTherm), San Diego, CA, USA, 1–4 June 2021; pp. 505–509.
25. Nejad, S.M.M.; Badawy, G.; Down, D.G. EAWA: Energy-Aware Workload Assignment in Data Centers. In Proceedings of the 2018 International Conference on High Performance Computing & Simulation (HPCS), Orleans, France, 16–20 July 2018; pp. 260–267.
26. Mousavi, A.; Yavarian, A.; Vyatkin, V.; Zhang, X. Power quality assessment of energy efficient cooling systems in data centers. In Proceedings of the IECON 2017—43rd Annual Conference of the IEEE Industrial Electronics Society, Beijing, China, 29 October–1 November 2017; pp. 7191–7196.
27. Mousavi, A.; Berezovskaya, Y.; Vyatkin, V.; Zhang, X. Energy efficient decision making in data centers with multiple cooling methods. In Proceedings of the IECON 2017—43rd Annual Conference of the IEEE Industrial Electronics Society, Beijing, China, 29 October–1 November 2017; pp. 8785–8790.
28. Berezovskaya, Y.; Mousavi, A.; Vyatkin, V.; Zhang, X. Smart Distribution of IT Load in Energy Efficient Data Centers with Focus on Cooling Systems. In Proceedings of the IECON 2018—44th Annual Conference of the IEEE Industrial Electronics Society, Washington, DC, USA, 21–23 October 2018; pp. 4907–4912.
29. Kim, S. Dual-Level Cooperative Game Approach for Energy-Aware Resource Allocation in Data Centers. *IEEE Access* **2019**, *7*, 113642–113652. [CrossRef]
30. Van Damme, T.; De Persis, C.; Tesi, P. Optimized Thermal-Aware Job Scheduling and Control of Data Centers. *IEEE Trans. Control. Syst. Technol.* **2019**, *27*, 760–771. [CrossRef]
31. Petrongolo, J.; Nemati, K.; Fouladi, K. Simulation-Based Optimization of Data Center Cooling Performance Using Performance Indicators. In Proceedings of the 2019 35th Annual Semiconductor Thermal Measurement, Modeling and Management Symposium (SEMI-THERM), San Jose, CA, USA, 18–22 March 2019; pp. 91–95.
32. Shi, B.; Huang, Y.; Zhang, Y. Simulation study of air distribution in a data center in Changsha. In Proceedings of the 2020 International Conference on Virtual Reality and Intelligent Systems (ICVRIS), Zhangjiajie, China, 18–19 July 2020; pp. 122–128.
33. Simon, V.S.; Siddarth, A.; Agonafer, D. Artificial Neural Network Based Prediction of Control Strategies for Multiple Air-Cooling Units in a Raised-floor Data Center. In Proceedings of the 19th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm), Orlando, FL, USA, 21–23 July 2020; pp. 334–340.
34. Battaglia, F.; Maheedhara, R.; Krishna, A.; Singer, F.; Ohadi, M.M. Modeled and Experimentally Validated Retrofit of High Consumption Data Centers on an Academic Campus. In Proceedings of the 2018 17th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm), San Diego, CA, USA, 29 May–1 June 2018; pp. 726–733.
35. Wang, J.V.; Cheng, C.T.; Tse, C.K. A power and thermal-aware virtual machine allocation mechanism for Cloud data centers. In Proceedings of the 2015 IEEE International Conference on Communication Workshop (ICCW), London, UK, 8–12 June 2015; pp. 2850–2855.
36. Haghshenas, K.; Pahlevan, A.; Zapater, M.; Mohammadi, S.; Atienza, D. MAGNETIC: Multi-Agent Machine Learning-Based Approach for Energy Efficient Dynamic Consolidation in Data Centers. *IEEE Trans. Serv. Comput.* **2022**, *15*, 30–44. [CrossRef]
37. Malik, S.U.; Bilal, K.; Khan, S.U.; Veeravalli, B.; Li, K.; Zomaya, A.Y. Modeling and Analysis of the Thermal Properties Exhibited by Cyberphysical Data Centers. *IEEE Syst. J.* **2017**, *11*, 163–172. [CrossRef]
38. Liu, H.; Aljbri, A.; Song, J.; Jiang, J.; Hua, C. Research advances on AI-powered thermal management for data centers. *Tsinghua Sci. Technol.* **2022**, *27*, 303–314. [CrossRef]
39. Cheung, H.; Wang, S.; Zhuang, C.; Gu, J. A simplified power consumption model of information technology (IT) equipment in data centers for energy system real-time dynamic simulation. *Appl. Energy* **2018**, *222*, 329–342. [CrossRef]
40. Li, Y.; Wen, Y.; Tao, D.; Guan, K. Transforming Cooling Optimization for Green Data Center via Deep Reinforcement Learning. *IEEE Trans. Cybern.* **2020**, *50*, 2002–2013. [CrossRef]
41. Skach, M.; Arora, M.; Hsu, C.-H.; Li, Q.; Tullsen, D.; Tang, L. Thermal Time Shifting: Decreasing Data Center Cooling Costs with Phase-Change Materials. *IEEE Internet Comput.* **2017**, *21*, 34–43. [CrossRef]

42. Tian, W.; VanGilder, J.; Condor, M. Potential Flow Modeling for Fast Data Center Thermal Simulation. In Proceedings of the 2021 20th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (iTherm), San Diego, CA, USA, 1–4 June 2021; pp. 341–349. [CrossRef]
43. Islam, M.A.; Ren, S.; Wierman, A. Exploiting a Thermal Side Channel for Power Attacks in Multi-Tenant Data Centers. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, Dallas, TX, USA, 30 October–3 November 2017; pp. 1079–1094. [CrossRef]
44. Cheong, K.H.; Tang, K.J.W.; Koh, J.M.; Yu, S.C.M.; Acharya, U.R.; Xie, N.-G. A Novel Methodology to Improve Cooling Efficiency at Data Centers. *IEEE Access* **2019**, *7*, 153799–153809. [CrossRef]
45. Kanellou, E.; Chrysos, N.; Bilas, A. A Flexible Datacenter Simulator. *Procedia Comput. Sci.* **2018**, *136*, 72–81. [CrossRef]
46. Yang, S.; Ordonez, J.C.; Xu, Y.; Cvetkovic, I. vemPEBB: Rapid PEBB Thermal Management Tool. In Proceedings of the 2021 IEEE Electric Ship Technologies Symposium (ESTS), Arlington, VA, USA, 3–6 August 2021; pp. 1–6.
47. Kheradmandi, M.; Down, D.G. Data driven fault tolerant thermal management of data centers. In Proceedings of the 2020 International Conference on Computing, Networking and Communications (ICNC), Big Island, HI, USA, 17–20 February 2020; pp. 736–740.
48. Zhang, S.; Zhou, T.; Ahuja, N.; Refai-Ahmed, G.; Zhu, Y.; Chen, G.; Wang, Z.; Song, W. Real time thermal management controller for data center. In Proceedings of the Fourteenth Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm), Orlando, FL, USA, 27–30 May 2014; pp. 1346–1353.
49. Al-Qawasmeh, A.M.; Pasricha, S.; Maciejewski, A.A.; Siegel, H.J. Power and Thermal-Aware Workload Allocation in Heterogeneous Data Centers. *IEEE Trans. Comput.* **2015**, *64*, 477–491. [CrossRef]
50. Zhao, X.; Peng, T.; Qin, X.; Hu, Q.; Ding, L.; Fang, Z. Feedback Control Scheduling in Energy-Efficient and Thermal-Aware Data Centers. *IEEE Trans. Syst. Man Cybern. Syst.* **2016**, *46*, 48–60. [CrossRef]
51. Sjolund, J.; Vesterlund, M.; Delbosc, N.; Khan, A.; Summers, J. Validated Thermal Air Management Simulations of Data Centers Using Remote Graphics Processing Units. In Proceedings of the IECON 2018—44th Annual Conference of the IEEE Industrial Electronics Society, Washington, DC, USA, 21–23 October 2018; pp. 4920–4925.
52. Nada, S.A.; Said, M.A.; Rady, M.A. Numerical investigation and parametric study for thermal and energy management enhancements in data centers' buildings. *Appl. Therm. Eng.* **2016**, *98*, 110–128. [CrossRef]
53. Nada, S.A.; Said, M.A. Effect of CRAC units layout on thermal management of data center. *Appl. Therm. Eng.* **2017**, *118*, 339–344. [CrossRef]
54. Song, X.; Liu, L.; Zhu, T.; Zhang, T.; Wu, Z. Comparative analysis on operation strategies of CCHP system with cool thermal storage for a data center. *Appl. Therm. Eng.* **2016**, *108*, 680–688. [CrossRef]
55. Alves, D.S.F.; Obraczka, K.; Kabbani, A. GDSim: Benchmarking Geo-Distributed Data Center Schedulers. In Proceedings of the 10th IEEE International Conference on Cloud Networking, CloudNet 2021, Cookeville, TN, USA, 8–10 November 2021; IEEE: New York, NY, USA, 2021; pp. 148–156.
56. Kannari, L.; Piira, K.; Bistrom, H.; Vainio, T. Energy-data-related digital twin for office building and data centre complex. In Proceedings of the 37th Conference on Design of Circuits and Integrated Systems, DCIS 2022, Pamplona, Spain, 16–18 November 2022; IEEE: New York, NY, USA, 2022; pp. 1–5.
57. Wang, Q.; Yu, Y.; Li, B.; Zhu, Y. Tensor-Based Optimal Temperature Control of CRACs in Multi-Datacenters. *IEEE Access* **2019**, *7*, 41445–41453. [CrossRef]
58. Zhou, Y.; Chen, Y.; Taneja, S.; Chavan, A.; Qin, X.; Zhang, J. ThermoBench: A thermal efficiency benchmark for clusters in data centers. *Parallel Comput.* **2020**, *98*, 102671. [CrossRef]
59. Chaudhry, M.T.; Jamal, M.H.; Gillani, Z.; Anwar, M.W.; Khan, M.S. Thermal-benchmarking for cloud hosting green data centers. *Sustain. Comput. Inform. Syst.* **2020**, *25*, 100357. [CrossRef]
60. Venkataraman, S.; Agrawal, D. Datacenters as Computers: Google Engineering & Database Research Perspectives. *Proc. VLDB Endow.* **2014**, *7*, 1720–1721.
61. Park, J.; Naumov, M.; Basu, P.; Deng, S.; Kalaiah, A.; Khudia, D.S.; Law, J.; Malani, P.; Malevich, A.; Satish, N.; et al. Deep Learning Inference in Facebook Data Centers: Characterization, Performance Optimizations and Hardware Implications. *arXiv* **2018**, arXiv:1811.09886.
62. Traini, L.; Cortellessa, V.; Pompeo, D.D.; Tucci, M. Towards effective assessment of steady state performance in Java software: Are we there yet? *Empir. Softw. Eng.* **2023**, *28*, 13. [CrossRef]
63. Del-Pozo-Puñal, E.; Carballeira, F.G. A Proposal of Mobility Support for the SimGrid Toolkit: Application to IoT simulations. In Proceedings of the 30th Euromicro International Conference on Parallel, Distributed and Network-Based Processing, PDP 2022, Valladolid, Spain, 9–11 March 2022; González-Escribano, A., García, J.D., Torquati, M., Skavhaug, A., Eds.; IEEE: New York, NY, USA, 2022; pp. 177–180.
64. Mahmud, M.R.; Pallewatta, S.; Goudarzi, M.; Buyya, R. iFogSim2: An extended iFogSim simulator for mobility, clustering, and microservice management in edge and fog computing environments. *J. Syst. Softw.* **2022**, *190*, 111351. [CrossRef]