A Taxonomy and Future Directions for Sustainable Cloud Computing: 360 Degree View

SUKHPAL SINGH GILL and RAJKUMAR BUYYA, The University of Melbourne, Australia

The cloud-computing paradigm offers on-demand services over the Internet and supports a wide variety of applications. With the recent growth of Internet of Things (IoT)–based applications, the use of cloud services is increasing exponentially. The next generation of cloud computing must be energy efficient and sustainable to fulfill end-user requirements, which are changing dynamically. Presently, cloud providers are facing challenges to ensure the energy efficiency and sustainability of their services. The use of a large number of cloud datacenters increases cost as well as carbon footprints, which further affects the sustainability of cloud services. In this article, we propose a comprehensive taxonomy of sustainable cloud computing. The taxonomy is used to investigate the existing techniques for sustainability that need careful attention and investigation as proposed by several academic and industry groups. The current research on sustainable cloud computing is organized into several categories: application design, sustainability metrics, capacity planning, energy management, virtualization, thermal-aware scheduling, cooling management, renewable energy, and waste heat utilization. The existing techniques have been compared and categorized based on common characteristics and properties. A conceptual model for sustainable cloud computing has been presented along with a discussion on future research directions.

Categories and Subject Descriptors: A.1 [**General Literature**]: Introductory and Survey; C.0 [**General**]: Systems Architectures; C.2.4 [**Computer-Communication Networks**]: Distributed Systems; D.4.1 [**Process Management**]: Scheduling; H.3.4 [**Systems and Software**]: Distributed Systems; J.7 [**Distributed Parallel and Cluster Computing**]; K.6.2 [**Management of Computing and Information Systems**]: Installation Management

General Terms: Documentation, Cloud Computing, Methodical Analysis, Conceptual Model, Focus of Study, Research Challenges, Theory, Reference Architecture, Trade-off, Future Directions, Management, Systematic Review, Survey

Additional Key Words and Phrases: Energy efficiency, sustainability, cloud datacenters, quality of service, green computing, holistic management, sustainable cloud computing, application design, energy management, renewable energy, thermal-aware scheduling, virtualization, sustainable cloud datacenters, capacity planning, sustainable metrics, cooling management, and waste heat utilization

ACM Reference format:
Sukhpal Singh Gill and Rajkumar Buyya. 2018. A Taxonomy and Future Directions for Sustainable Cloud Computing: 360 Degree View. *ACM Comput. Surv.* 51, 5, Article 104 (December 2018), 33 pages.
https://doi.org/10.1145/3241038

This work is supported by the Melbourne-Chindia Cloud Computing (MC3) Research Network and Australian Research Council (DP160102414).
Authors' addresses: S. S. Gill and R. Buyya, Cloud Computing and Distributed Systems (CLOUDS) Laboratory, School of Computing and Information Systems, The University of Melbourne, Parkville, Australia - 3010; emails: sukhpal.gill@unimelb.edu.au, rbuyya@unimelb.edu.au.
Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.
© 2018 Association for Computing Machinery.
0360-0300/2018/12-ART104 $15.00
https://doi.org/10.1145/3241038

1 INTRODUCTION

Cloud computing offers a flexible and powerful computing environment to provide on-demand, subscription-based online services over the Internet to host applications on a pay-as-you-go basis. The various cloud providers, such as Microsoft, Google, and Amazon, make extensive use of Cloud Data Centers (CDCs) to fulfill the requirements (memory, data, compute, or network) of the digital world. To reduce the service delay and maintain the Service Level Agreement (SLA), fault tolerance should be provided through replicating the compute abilities redundantly [1]. To ensure the availability and reliability of services, the components of CDCs—such as network devices, storage devices, and servers—should be run 24/7 [2]. Large amounts of data are created by digital activities such as data streaming, file sharing, searching and social networking websites, e-commerce, and sensor networks. That data can be stored as well as processed efficiently using CDCs [3, 4]. The energy cost is added by creating, processing, and storing each bit of data, which increases carbon footprints that further impacts on the sustainability of cloud services. Due to the large consumption of electricity by CDCs, the research community is addressing the challenge of designing sustainable CDCs [5].

With the continuous growth of Internet of Things (IoT)–based applications, the use of cloud services is increasing exponentially, which further increases the electricity consumption of CDCs by 20% to 25% every year [6]. Existing studies claimed that 78.7 million metric tons of CO_2 are produced by datacenters, which is equal to 2% of global emissions [7]. CDCs in the United States consumed 100 billion kilowatt hours (kWh) in 2015, which is sufficient for powering Washington, DC [11] for a year. The consumption of electricity will reach 150 billion kWh by 2022, that is, increase by 50% [12]. Energy consumption in CDCs can be increased to 8000 terawatt hours (TWh) in 2030 if controlled mechanisms are not identified [122, 81]. Due to underloading and overloading of resources in infrastructure (cooling, computing, storage, networking, etc.), the energy consumption in cloud datacenters is not efficient and the energy is consumed mostly while some of the resources are in idle state, which increases the cost of cloud services [11]. Carbon footprints produced by CDCs are the same as that of the aviation industry [13, 135]. In the current scenario, CDC service providers are finding alternative ways to reduce the carbon footprint of their infrastructure. The prominent cloud providers—such as Google, Amazon, Microsoft, and IBM—have vowed to attain zero production of carbon footprints and they are finding the new ways to make CDCs and cloud-based services eco-friendly [3]. Therefore, CDCs need to provide cloud services with a minimum carbon footprint and minimum heat release in the form of greenhouse gas emissions [71, 136]. The energy consumption of different components of a CDC [3, 71, 95] is shown in Figure 16 of Appendix A. The types of sustainability spheres are shown in Figure 17 of Appendix A. The investment in cloud computing is shown in Figure 18 of Appendix A.

To solve this challenge of energy-efficient cloud services, a large number of researchers proposed resource management policies, algorithms and architectures, but energy efficiency is still a challenge for future researchers. To ensure a high level of sustainability, holistic management of resources can solve new open challenges existing in resource scheduling. There is a need for methods that harness renewable energy to decrease carbon footprints without the use of fossil fuels. Further, cooling expenses can be decreased by developing waste heat utilization and free cooling mechanisms. Location-aware ideal climatic conditions are needed for an efficient implementation of free cooling and renewable energy production techniques [47, 57, 91]. Moreover, waste heat recovery locations are required to be identified for an efficient implantation of waste heat recovery predictions. CDCs can be relocated based on (i) opportunities for waste heat recovery, (ii) accessibility of green computing resources and (iii) proximity of free cooling resources. Cloud providers such as Google, Amazon, IBM, Facebook, and Microsoft are using more green energy resources instead of grid electricity [31, 58].

1.1 Background

The background of sustainable cloud computing is given in Appendix A.

1.2 Related Surveys and Our Work

The comparison of our work with related surveys is presented in Table 1 of Appendix A.1.

1.3 Our Contributions

- A comprehensive taxonomy for sustainable cloud computing is proposed.
- A broad review has been conducted to explore various existing techniques for sustainable cloud computing.
- The current research on sustainable cloud computing is organized into several categories: application design, sustainability metrics, capacity planning, energy management, virtualization, thermal-aware scheduling, cooling management, renewable energy, and waste heat utilization.
- Existing techniques have been compared and categorized based on common characteristics and properties.
- A conceptual model for sustainable cloud computing has been proposed.
- The taxonomy and survey results are used to find the open challenges that have not been fully explored in the research.

1.4 Article Structure

The rest of the article is organized as follows: Section 2 describes the review technique used to find and analyze the available existing research, research questions, and searching criteria. Section 3 presents a proposed comprehensive taxonomy and systematic review of existing techniques for sustainable cloud computing. Based on common characteristics and properties, techniques have been compared and categorized. Section 4 describes the outcomes of this systematic review. Section 5 introduces the open challenges and future research directions along with implications of this research in the area of sustainable cloud computing. Section 6 offers the conceptual model for sustainable cloud computing. Section 7 summarizes this research. ***Note:*** Important information regarding sustainable cloud computing is included in the online Appendix; see the Appendix for the full picture.

2 REVIEW METHODOLOGY

The review technique [45] used in this systematic review is described in Appendix B.

3 SUSTAINABLE CLOUD COMPUTING: A TAXONOMY

The ever-increasing demand for cloud computing services that are deployed across multiple cloud datacenters harnesses significant amount of power, resulting in not only high operational cost but also high carbon emissions [87, 46]. In sustainable cloud computing, the CDCs are powered by renewable energy resources by replacing the conventional fossil fuel-based grid electricity or brown energy to effectively reduce carbon emissions [2]. Employing energy-efficiency mechanisms also makes cloud computing sustainable by reducing carbon footprints to a great extent [144]. Waste heat utilization from heat dissipated through servers and employing mechanisms for free cooling of the servers make the CDCs sustainable [3, 71, 80]. Thus, sustainable cloud computing employs the following elements in making the datacenter sustainable [4]: (i) using renewable energy instead of grid energy generated from fossil fuels, (ii) using the waste heat generated from heat dissipating servers, (iii) using free cooling mechanisms, and (iv) using energy-efficient mechanisms [145]. All of these factors contribute to reducing carbon footprints, operational cost, and energy

consumption to make CDCs more sustainable [146, 165, 166]. Figure 20 of Appendix C presents various elements that impact or support sustainable cloud computing (360-Degree View), which have been broken out into nine categories: application design, sustainability metrics, capacity planning, energy management, virtualization, thermal-aware scheduling, cooling management, renewable energy, and waste heat utilization based on the existing literature. Table 6 of Appendix C contains the mapping of aspects of sustainable CDCs to types of sustainability spheres based on Figures 17 and 20.

3.1 Application Design

In sustainable cloud computing, the design of an application plays a vital role and the efficient structure of an application can improve energy efficiency of CDCs. The resource manager and scheduler follow different approaches for application modelling [25, 26]. For example, the scheduling algorithm for the Map Reduce model follows a different approach compared with other models such as workflow, web application, streaming application, and graph processing. To make the infrastructure sustainable and environmentally eco-friendly, there is a need for green ICT-based innovative applications [19, 140, 141, 142]. Effective design of cloud applications contains APIs or services. Applications (e.g., web applications) designed following three-tiered architectures contain user interfaces, application processing, and databases [29, 27, 42]. The functionality of each tier should be independent to run at different providers to improve its performance, simplicity, and reliability [138, 79]. The components of applications should have minimum dependency, that is, they should be loosely coupled. Applications can be ported from one server to another without affecting their execution [43, 44]. At the software level, cloud users can use applications in a flexible manner, which are running on cloud datacenters [78].

Recent technological developments such as the Internet of Things (IoT) and software-defined cloud-based applications are creating new research areas for sustainable cloud computing [1, 56]. The emerging IoT-based applications, such as smart cities and health care services, are increasing, which need appropriate application design model for fast data processing that improves the performance of computing systems [17, 3]. However, these applications are facing high delay and response times because computing systems need to transfer data to the cloud and then from the cloud to an application, which affects the sustainability of cloud computing [2, 24, 137]. Due to a large amount of data processing in the cloud, a computing system does not process at the required speed, which leads to communication failures. Moreover, data security is also a high-priority requirement of sustainable computing to protect critical information from attackers in the case of e-commerce applications [38, 20]. There is a need for reevaluation of existing application models of cloud computing to address research issues such as energy efficiency, sustainability, privacy, security, and reliability. The evolution of application design techniques (see Figure 21) and their comparison along with open research challenges [29, 18, 138, 79, 78, 56, 24, 137, 38, 20, 26, 42, 25, 35, 54, 44] are presented in Table 7 of Appendix C.1.

3.1.1 Application Design-Based Taxonomy. Different types of applications are running in cloud environments, such as computation intensive or data intensive. To improve the performance of cloud computing systems, it is necessary to execute applications in parallel. Based on the requirements of the cloud user, quality of service (QoS) parameters for every application are identified as well as provisions of the resources for execution. The components of the application design taxonomy are (i) QoS parameter, (ii) application model, (iii) workload type, and (iv) type of architecture, as shown in Figure 1. Each of these taxonomy elements are discussed below along with relevant examples. The comparison of existing techniques (discussed in Appendix C.1) based on our application design taxonomy is given in Table 8 of Appendix C.1.

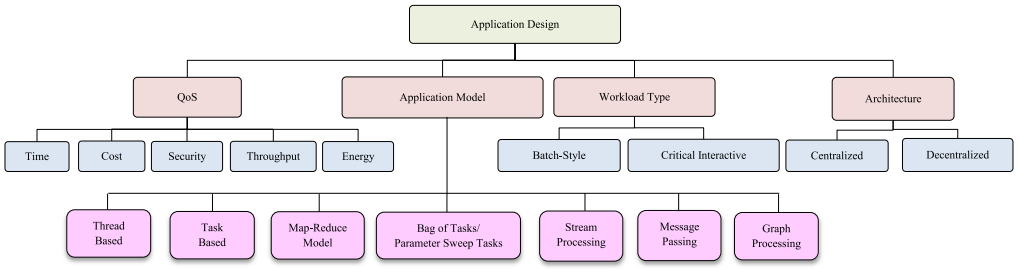


Fig. 1. Taxonomy based on application design.

3.1.1.1 Quality of Service (QoS). Different applications have their different QoS requirements. There are five main types of QoS parameters for sustainable computing as identified from the literature [20, 33, 39, 58, 59, 77], such as execution cost, time, energy consumption, security, and throughput. *Execution cost* is total money that can be spent in 1 hour to execute the application successfully. *Execution time* is the amount of time required to execute an application successfully. *Energy* is the amount of electricity expended by a resource to complete the execution of an application. *Security* is an ability of the computing system to protect the system from malicious attacks. *Throughput* is the ratio of total number of tasks of an application to the amount of time required to execute the tasks. Other QoS requirements of cloud service can be reliability, availability, scalability, and latency.

3.1.1.2 Application Models. The complexity of applications is increasing day by day and the cloud platform can be used to handle user applications. Different types of application models are being developed for a wide range of domains to satisfy the different types of customers for sustainable computing [25, 42-44, 81, 125]. There are seven types of *application models* as identified from the literature [47, 48]: (1) thread-based, (2) task-based, (3) Map-Reduce model, (4) bag-of-tasks or parameter sweep tasks, (5) stream processing, (6) message passing, and (7) graph processing. In the *thread-based* model, one process is divided into multiple threads, which execute concurrently and share resources such as memory, network, and processor to complete execution. In the *task-based* model, a large task is divided into small tasks that are executed in parallel on different non-sharable cloud resources. *Map-reduce tasks* split the input dataset into independent chunks and in a parallel execution, which is used to execute the mapped tasks. Further, the outputs of the maps are sorted and used as an input to the reduce tasks. *Bag-of-tasks* or *parameter sweep tasks* refers to the jobs that are parallel, among which there are no dependencies and are identical in their nature and differ only by the specific parameters used to execute them: for example, video coding and encoding. *Stream processing* is the processing of small-sized data (in kilobytes) generated continuously by thousands of data sources (geospatial services, social networks, mobile or web applications, online gaming, and video streaming), which typically send data records simultaneously. An example of a stream processing model can be a video processing application. The *Message Passing* interface provides a communication functionality between a set of processes, which are mapped to nodes or servers in a language-independent way and encouraged development of portable and scalable large-scale parallel applications. *Graph processing* involves the process of analyzing, storing, and processing graphs to produce effective outputs.

3.1.1.3 Workload Types. For workload management in sustainable cloud computing, there are mainly two types of IT workloads that are considered for sustainable computing: batch style and critical interactive [32, 33]. *Batch-style* workloads are submitted to a job queue and will be executed when resources become available. Multiple batch jobs are often submitted without any deadline

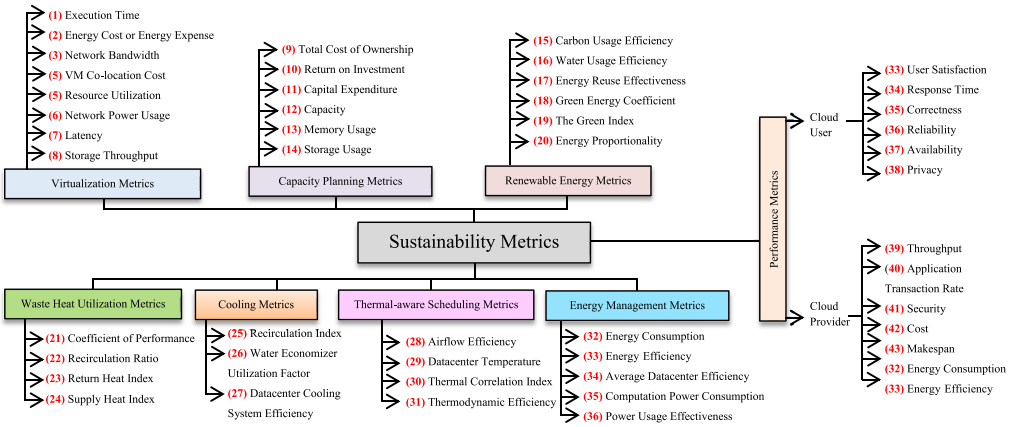


Fig. 2. Taxonomy of metrics for sustainable cloud computing.

constraint together and are executed with maximum resource use. The workloads that need immediate response but whose execution should be completed before their deadline are called *critical interactive* workloads.

3.1.1.4 Architecture. The architecture is an important component of sustainable cloud computing. There are basically two types of architectures: centralized and decentralized [46, 55, 58, 137]. In *centralized* architectures, there is a central controller that manages all the tasks that need to be executed and executes those tasks using scheduled resources. The central controller is responsible for the execution of all tasks. In *decentralized* architectures, resources are allocated independently to execute the tasks without any mutual coordination. Every resource is responsible for its own task execution.

The performance of QoS parameters of different cloud applications is measured using different metrics as discussed in Section 3.2.

3.2 Sustainability Metrics

As use of cloud infrastructure is growing exponentially, it is important to monitor and measure the performance of CDCs regularly. We have identified different types of metrics from the literature [9, 14, 15, 16, 22, 23, 28, 30, 32, 34, 36, 51, 52, 60, 67, 83, 84, 85, 86, 96, 125] and present a taxonomy of metrics for different categories for sustainable cloud computing based on the core operations of CDCs. Figure 2 shows the taxonomy of metrics for application design, capacity planning, energy management, virtualization, thermal-aware scheduling, cooling management, renewable energy, and waste heat utilization. The detailed description of metrics for sustainable cloud computing can be found in [36]. Table 9 (in Appendix C.2) presents the year-wise use of sustainability metrics in different categories of sustainable cloud computing to measure the performance of numerous infrastructure components of CDCs. Table 10 (in Appendix C.2) presents the brief definition of sustainability metrics. Effective capacity planning in the cloud era demands some resource flexibility due to changing application requirements and hosting infrastructure, which is discussed in Section 3.3.

3.3 Capacity Planning

Cloud service providers must initiate effective and organized capacity planning to enable sustainable computing. Capacity planning can be done for power infrastructure, IT devices, and cooling.

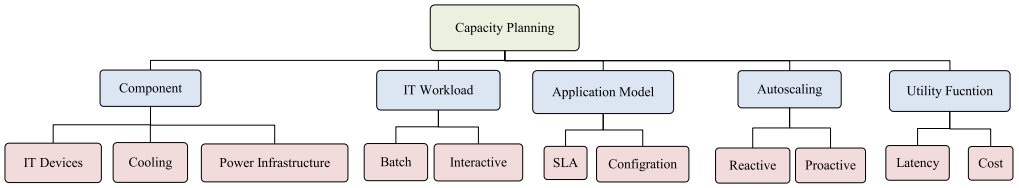


Fig. 3. Taxonomy based on capacity planning.

The capacity of a CDC can be planned effectively by considering the devices of end users, for example, encoding techniques for a video on-demand application [109].

An SLA should be there for important parameters such as backup and recovery, storage, and availability to improve user satisfaction, which attracts more customers in future. There is a need to consider important utilization parameters per application to maximize the use of resources through virtualization by finding the applications, which can be merged. Merging applications improves resource utilization and reduces capacity cost, which makes cloud infrastructure more sustainable. For efficient capacity planning, cloud workloads should be analyzed before execution to finish their execution for deadline-oriented workloads [11, 108]. To manage power infrastructure effectively, virtual machine (VM) migration should be provided for migration of workloads or machines to successfully complete the execution of workloads with minimum use of resources, which improves the energy efficiency of CDCs. Effective capacity planning can truly enable a sustainable cloud environment. The evolution of capacity planning techniques (see Figure 22) and their comparison along with open research challenges [149, 114, 113, 112, 109, 111, 110] can be found in Table 11 of Appendix C.3.

3.3.1 Capacity Planning-Based Taxonomy. Capacity planning is done based on component, IT workload, model, autoscaling, and utility function, as shown in Figure 3. Each of these taxonomy elements are discussed below along with relevant examples. The comparison of existing techniques (discussed in Appendix C.3) based on our capacity planning taxonomy is given in Table 12 of Appendix C.3.

3.3.1.1 Component. Capacity planning is required for every *component* of a CDC, such as IT devices, cooling, and power infrastructures [109, 110, 115]. *IT devices* are an important component, which are required to execute the operations of CDCs. Due to consumption of huge amount of energy, an efficient planning of *cooling* is required to maintain the temperature of CDCs. The planning of the *power infrastructure* is the most important element of a CDC to run it every time, that is, 24×7.

3.3.1.2 IT Workload. There are mainly two types of *IT workloads,* which are considered for capacity planning: batch style and critical interactive, as described in Section 3.1.1.3.

3.3.1.3 Application Models. There are two types of design *models* for effective capacity planning: SLA based and configuration based [111, 114]. In the *SLA*-based model, capacity of CDCs is planned based on the QoS requirements of the workloads without SLA violations. The *configuration*-based model focuses on the configuration of the CDC, such as processor, memory, network devices, cooling, and storage, which are required to execute the workloads effectively.

3.3.1.4 Autoscaling. The capacity of a CDC is also planned for *autoscaling*, which may be proactive or reactive [109, 113, 116]. *Reactive* autoscaling works based on feedback methods and manages the requirements based on their current state to maintain its performance. *Proactive* autoscaling manages the capacity requirements based on the prediction and assessment of performance in

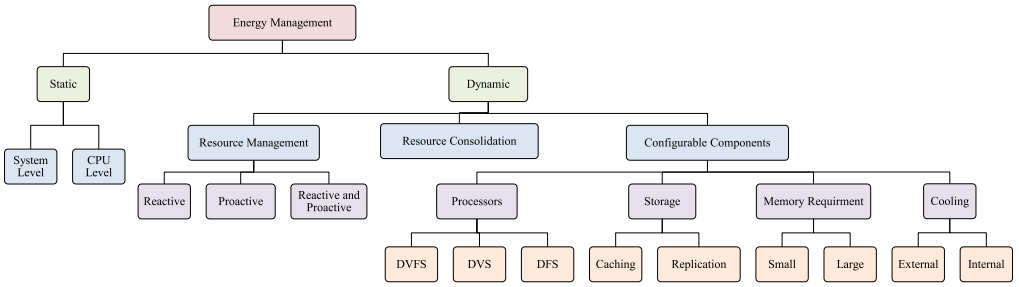


Fig. 4. Taxonomy based on energy management.

terms of QoS values. Based on previous data, predictions have been identified and required action is planned to optimize CDC performance.

3.3.1.5 Utility Functions. Latency and cost-based *utility functions* are defined to measure the aspects of capacity planning [113, 114]. *Cost* is defined as the amount of money that can be spent to design a CDC with required configuration. *Latency* is the amount of execution delay with a particular configuration of CDC.

To manage power infrastructure for capacity planning, energy management of resources is required for execution of workloads, which improves the energy efficiency of CDCs.

3.4 Energy Management

Energy management in sustainable computing is an important issue for cloud service providers. Ficco and Rak [5] reported that more than 2.4% of electricity is consumed by CDCs, with a large economic impact of $30 billion globally. The energy requirement to manage the CDCs is also rising in proportion to the operational cost. IBM spends 45% of total expenses on CDC electricity bills and the consumption of electricity will be increased to 101.5 billion kWh by 2022 [10]. Sustainable cloud services are attracting more cloud customers and making it more profitable [62, 60, 64, 65]. Improving energy use reduces electricity bills and operational costs to enable sustainable cloud computing. The essential requirements of sustainable CDCs are optimal software system design, optimized air ventilation, and installing temperature monitoring tools for adequate resource utilization, which improves energy efficiency [67, 74, 120]. There are mainly three levels where energy consumption can be optimized: software level (efficient use of registers, buffers, etc.), hardware level (transistors, voltage supply, logical gates, and clock frequency) and intermediate level (energy-aware resource provisioning techniques) [80]. The evolution of energy management techniques (see Figure 23) and their comparison along with open research challenges [3, 25, 60, 61, 62, 66, 75, 76, 41, 60, 67, 125, 83, 74, 80] are presented in Table 13 of Appendix C.4.

3.4.1 Energy Management-Based Taxonomy. Energy management has two important components (static and dynamic), as shown in Figure 4. These taxonomy elements are discussed below, accompanied by relevant examples. The comparison of existing techniques (discussed in Appendix C.4) based on our energy management taxonomy is given in Table 14 of Appendix C.4.

3.4.1.1 Static Energy Management. Static energy management is a more engineering-oriented approach, in which circuitry systems are considered more by offline energy management [60]. During design time, the entire optimization happens at system level and deals with factorization, path balancing, transistor sizing, instruction sets, redesigning of architectures, circuit manipulation, and processing centers [67].

Low-power use components are employed in this management approach to reduce energy consumption as much as possible. Static energy management performs at two levels: system level and CPU level. Existing studies [125, 83] found that the *CPU* offers a big scope of optimization of energy consumption because computing components of the CPU consumes 35% to 40% of energy [3, 68, 69, 71]. The optimization of energy at *CPU level* can be performed at the instruction set level or register level. Researchers designed different instruction set architectures to improve resource utilization, such as reduced bit-width architecture at the instruction set level [70]. On the other hand, the activities of the register transfer level are optimal for decreasing energy consumption. Figure 24 of Appendix C.4. shows the energy cost and carbon dioxide emission for static and dynamic energy management techniques [122, 144, 145].

Other components of the *system* besides the CPU that consume large amounts of energy are software systems, network facility, and memory components [143]. Researchers proposed different management techniques to optimize the power consumption of these components based on the setup techniques used in system design [73]. At design time, it is very difficult to select the right components to design a cloud system with maximum synchronization among the components [61, 62, 66]. Other important challenges during system design can be: (i) type of application and software, (ii) selection of operating system, and (iii) placement of servers to reduce delay. Gordan and Fast Array of Wimpy Nodes (FAWN) [3] architecture has been designed to improve the performance of cloud systems by balancing the input-output activities and computation processes by coupling datacenter powering systems and local flash storage with low-power CPUs. Energy consumption can be reduced by proper distribution of resources geographically and the selection of suitable network topologies and components with maximum compatibility [72].

3.4.1.2 Dynamic Energy Management. Software-based policies are used in dynamic energy management to improve energy utilization. There is a different dynamic power range for every component. During low-activity modes, a CPU consumes 30% of the peak value of its energy consumption and can be scaled up and down up to 70% [80]. The dynamic range of energy consumption for disk drives is 50%, 25% for memory, and 15% for network devices such as routers and switches [83]. To improve energy utilization, the number of components can be scaled up or down based on the range of dynamic power. Dynamic energy management is divided into three categories based on the reduction of the dynamic power range: (i) configurable components, (ii) resource consolidation, and (iii) resource management.

Configurable components include the *CPU,* which supports low-activity modes at the component level. Dynamic energy management can be used to control the CPU. The CPU is the main source of energy consumption. Thus, existing research work mainly focused on optimization of energy consumption by the CPU or processor and memory. There is a relationship between power supply, voltage, and operational frequency [66, 62]: ($Power_{Dynamic} = Utilization_{CPU} \times$ Frequency $\times Voltage^2$). Based on the different values of voltage and operational frequency, a CPU can run in different activity modes or C-modes in advance processor architectures. As supply voltage increases, the energy consumption increases quadratically in Complementary Metal Oxide Semiconductor (CMOS) circuits [3]. The values of linear relations can be exploited by changing operation frequency (DFS), voltage (DVS) or both simultaneously (DVFS) [60]. DVFS for energy management is described in Appendix C.4.1 and C-states or C-modes for energy management is described in Appendix C.4.2.

There are a number of methods proposed to control energy consumption by scaling down the high voltage supply, but the best way is to exploit the *stall time*. A high amount of clock speed is wasted while waiting for the data because of the speed gap between processor and main memory. Energy may be saved by reducing the processor frequency through manipulation of supply

voltage. For different devices, semiconductor chip vendors optimizing energy consumption use different frequency scaling policies. Eight different kinds of operational frequencies are available in Intel's Woodcrest Xeon Processor [3]. Two CPU throttling technologies developed by AMD are PowerNow and CoolnQuiet [3, 125]. Another, the SpeedStep CPU throttling technology, has been developed by Intel to control energy consumption [62]. The *cooling* can be *internal* (fans) or *external* (as discussed in Section 3.7) for a CDC.

The management of *storage* devices such as disk drives is handled by scalable storage systems to reduce energy consumption because disk drives consume significant amounts of energy. The storage of data can be managed using either *replication* or *caching*. Mechanical operations of storage components consume one-third of the total electricity provided to CDCs, and disks also consume one-tenth during standby mode. The need for storage components is increasing by 60% annually [66, 67]; thus, research on energy consumption control is imperative. Disk drives use only 25% of their storage space, which remains underutilized in large CDCs [3, 71]. Power use can be minimized by reducing underutilization by switching off unnecessary disks. Many mechanisms have been proposed to improve the energy efficiency of disk drives [2]. In large-scale CDCs, the *memory* component may be considered to decrease power use, but it is the least addressed component by researchers. Memory consumes 23% of energy to run a specific workload [83, 125]. The dynamic range for memories is 50%, as discussed above; thus, there is a chance to improve energy consumption in this component [61, 62]. DVFS is also applicable to memory components by reducing frequency and voltage. Storage arrays are the most important components of DRAMs in which power consumption can be reduced. It is challenging to develop energy-aware memory components in cloud computing to reduce power consumption without degradation of performance. Also, it is difficult to manufacture energy-efficient memory devices with lower cost. Existing memory management infrastructures can minimize energy consumption up to 70% [6].

Resource consolidation is a technique for effective use of resources (processor, memory, or network devices) to minimize the number of resources and locations of servers that a cloud company requires to serve user requests [71]. A resource scheduler allocates resources to execute workloads dynamically to avoid over utilization and under-utilization of resources. *Resource management* is an significant challenge because of the following factors: (i) heterogenous resources, (ii) varying costs, (iii) applications with varying requirements (compute, data, network, memory), and (iv) user QoS requirements. Effective resource management includes resource allocation, resource scheduling, and resource monitoring to achieve effective utilization of resources [32]. Many issues need to be addressed to achieve this, including the following [32, 71]:

(a) How to allocate the resources in an energy-efficient manner for the execution of workloads
(b) When to migrate workloads from one machine to another to save energy consumption
(c) Which devices need to be switched off to save energy consumption without degradation of performance

Based on existing research, the techniques above addressed issues to improve energy utilization. These techniques are classified into the following categories: (a) Proactive, (b) Reactive, and (c) Proactive and Reactive. *Proactive* management manages the resources based on the prediction of future performance of the system instead of its current state. The resources are selected based on the previous executions of the system in terms of reliability, energy consumption, throughput, and the like. The predictions are required to be based on previous data and appropriate actions are formulated to optimize energy consumption during resource execution. *Reactive* management works based on feedback methods and manages the resources based on their current state to optimize energy. There is a need of continuous monitoring of resource allocation to find whether

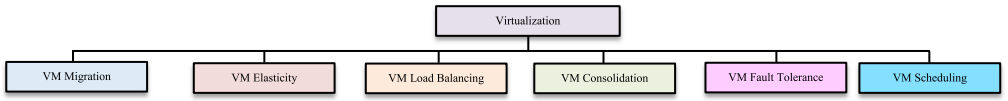


Fig. 5. Taxonomy based on virtualization.

the energy is consumed less than its threshold value or not (threshold value can be based on energy as well as resource utilization). If power usage is higher than threshold value, then corrective action will be taken to optimize the energy consumption. The accuracy of the monitoring module improves the productivity of reactive management. In the case of underutilization of resources, energy consumption can be reduced through VM consolidation or migration as discussed in Section 3.5. Increase in energy consumption also requires effective cooling management because temperature increases due to large amounts of heat. *Reactive and proactive* management manages the resources with minimum value of power usage and maximum value of resource utilization to handle every situation by (i) monitoring the resource execution continuously and (ii) performing the actions based on predicted failures. In real time, it is challenging to accurately forecast the behavior of a system in proactive management. In reactive management, there is a larger overhead, which causes unnecessary delay as well as energy inefficiency [60].

A virtualization technology reduces the number of physical machines or resources and executes the workloads using virtual resources, which leads to a reduction in energy consumption.

3.5 Virtualization

Virtualization technology is an important part of sustainable CDCs to support energy-efficient VM migration, VM elasticity, VM load balancing, VM consolidation, VM fault tolerance, and VM scheduling [88]. Operational costs can be reduced by using VM scheduling to manage cloud resources using efficient dynamic provisioning of resources [102]. During the execution of workloads, VM load balancing is required to balance the load effectively owing to decentralized CDCs and renewable energy resources. Owing to the lack of on-site renewable energy, VM techniques migrate the workloads to the other machines distributed geographically. VM technologies also offer migration of workloads from renewable energy-based CDCs to the CDCs using the waste heat at another site [105]. To balance the workload demand and renewable energy, VM-based workload migration and VM consolidation techniques provide virtual resources using few physical servers. VM fault tolerance creates and maintains the identical secondary VM for the replacement of the primary VM in a failover situation without affecting the availability of cloud service. VM elasticity maintains the performance of the computing system by providing the dynamic adaptation of computing resources or capacity to fulfill the changing requirements of workloads. Waste heat use and renewable energy resource alternatives are harnessed by VM migration techniques to enable sustainable cloud computing [82, 104]. It is a great challenge for VM migration techniques to improve energy savings and network delays while migrating workloads between resources distributed geographically. The evolution of virtualization technologies (see Figure 25) and their comparison along with open research challenges [40, 63, 159, 160, 162, 157, 158, 161, 163, 88, 101, 99, 102, 105, 106, 156, 37] can be found in Table 15 of Appendix C.5.

3.5.1 Virtualization-Based Taxonomy. Based on the literature, virtualization consists of the following components: VM migration, VM elasticity, VM load balancing, VM consolidation, VM fault tolerance, and VM scheduling, as shown in Figure 5. Each of these taxonomy components are discussed below along with their subcomponents and relevant examples.

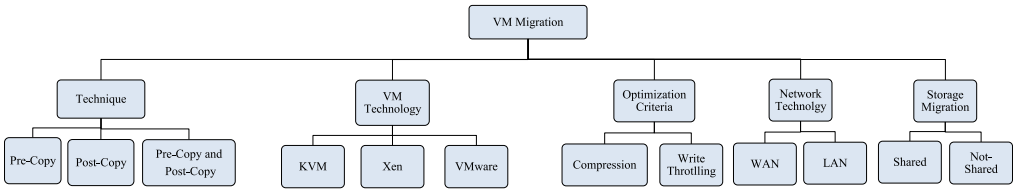


Fig. 6. Taxonomy based on VM migration.

The comparison of existing techniques (discussed in Appendix C.5) based on our virtualization taxonomy is given in Table 16 of Appendix C.5 (VM migration, VM elasticity and VM load balancing) and Table 17 of Appendix C.5 (VM consolidation, VM fault tolerance and VM scheduling).

3.5.1.1 VM Migration-Based Taxonomy. VM migration is a process of relocation of a running VM from one physical machine to another without affecting the execution of user application. Based on the literature [101, 102, 106, 164], VM migration consists of the following components: (i) technique, (ii) VM technology, (iii) optimization criteria, (iv) network technology, and (v) storage migration, as shown in Figure 6.

3.5.1.1.1 Technique. VMs can be migrated from one place to another for better utilization of resources, reducing the under utilization and over utilization of resources [99]. Three types of techniques have been proposed for VM migration: (1) Pre-copy, (2) Post-copy, and (3) Pre-copy and Post-copy. There are two different phases of *pre-copy* technique: warm-up and stop and copy. In the *warm-up* phase, the hypervisor copies the state from the source server to the destination server, which contains the information about the memory state and the CPU state. The *stop and copy* phase copies the pending files (if any file is modified during the warm-up phase) from the source to destination servers and starts the execution at the destination server [88]. In *post-copy*, it stops the VM at the source server, transfers all the details, such as CPU state and memory state, to the destination server, and starts execution. Some VM migration mechanisms use both *pre-copy* and *post-copy* together to transfer states from one server to another.

3.5.1.1.2 VM technology. There are three different types of technology that are available in the literature for VM migration: KVM, Xen, and VMware. *KVM* is a kernel-based VM, which permits many operating systems (OSs) to share a single resource or hardware. *Xen* works based on a micro-kernel design to share the same resources to run multiple OSs. *VMware* can be used for application consolidation to provide services through virtualization [101].

3.5.1.1.3 Optimization criteria. It has been determined that optimization criteria for virtualization technology can be compressed or go through write throttling. ESXi is an independent hypervisor, which offers memory *compression* cache to increase the performance of VMs and further increases the capacity of the CDC [105]. *Write throttling* is used to perform write and incoming copy operations, which limit the transfer of data [106].

3.5.1.1.4 Network technology. There are two different types of network technologies used for VM migration: WAN and LAN. *Wide Area Network (WAN)* is used to migrate a VM geographically using a wireless connection, while a *Local Area Network (LAN)* is used to migrate a VM from one server to another within a limited area.

3.5.1.1.5 Storage migration. In this technique, storage from one running server to another can be migrated without affecting the workload execution of VMs. Storage migration can also be used

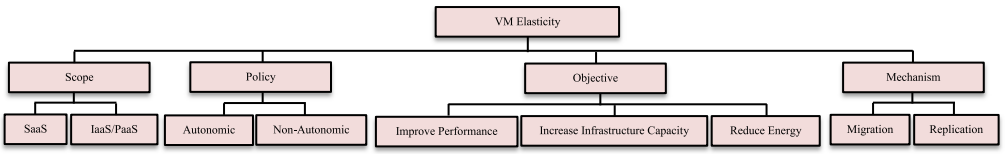


Fig. 7. Taxonomy based on VM elasticity.

to upgrade storage resources or transfer service [101, 32]. The distributed file systems can be used to provide shared storage space.

The main issues with VM migration in geographically distributed datacenters are discussed in Appendix C.5.1. Container as a Service (CaaS) for virtualization is discussed in Appendix C.5.2.

3.5.1.2 VM Elasticity-Based Taxonomy. VM elasticity enables the automatic provisioning and de-provisioning of computing resources to fulfill the changing demand of workloads at runtime. Based on the literature [40, 37, 163, 164], VM elasticity consists of the following components: (i) scope, (ii) policy, (iii) objective, and (iv) mechanism, as shown in Figure 7.

3.5.1.2.1 Scope. It defines the location, where the elasticity actions are managed, which can be application (SaaS) or platform level (PaaS) and infrastructure level (IaaS) [62]. At *IaaS level*, the elasticity controller monitors the application execution and performs different decisions based on resource (hardware) scalability. At *SaaS or PaaS level*, the elasticity controller is implanted in the application or within the execution platform, which performs the dynamic scalability of cloud resources.

3.5.1.2.2 Policy. There are two types of policies for the execution of elasticity actions: autonomic and non-autonomic [66]. In *autonomic* policy, the cloud system or application controls the elasticity actions and performs actions based on the SLA constraints. In *manual* policy, the user monitors the virtual environment and performs the elasticity actions accordingly.

3.5.1.2.3 Objective. VM elasticity techniques have three main objectives: (1) improve performance, (2) increase infrastructure capacity, and (3) reduce energy [67]. The main objective of VM elasticity techniques is to improve *performance*, such as optimal searching of VM and reducing the task rejection rate and makespan. The second objective is to *reduce energy consumption* of CDCs during execution of workloads. The third objective is to *improve the infrastructure capacity* by adding different resources at runtime to execute workloads within their specified budget and deadline.

3.5.1.2.4 Mechanism. There are two different mechanisms for VM elasticity as identified from the literature [83]: migration and replication. *Migration* refers to moving the VM from one physical machine to another for effective use of application load using deconsolidation and consolidation of resources. *Replication* refers to elimination and removal of instances (application modules, containers, VMs) from the virtual environment.

3.5.1.3 VM Load Balancing-Based Taxonomy. VM load balancing refers to the optimization of use of VMs to reduce resource wastage due to underloading and overloading of resources. It helps to achieve QoS and maximize resource use to improve performance of cloud service. Based on the literature [88, 101, 162], VM load balancing consists of the following components: (i) resource-aware, and (ii) performance-aware, as shown in Figure 8.

3.5.1.3.1 Resource-aware. CDCs require different types of resources (memory, processor, cooling, storage, networking etc.) to execute user workloads [74]. *Resource-aware* load balancing

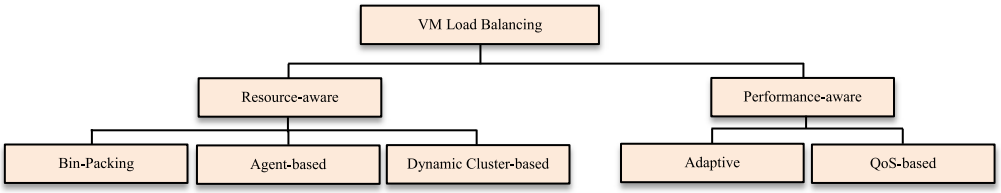


Fig. 8. Taxonomy based on VM load balancing.

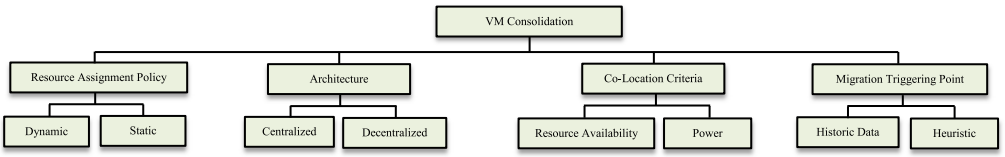


Fig. 9. Taxonomy based on VM consolidation.

algorithms execute workloads, also monitoring and analyzing the different performance parameters related to resources such as energy consumption, degree of resource capacity imbalance, and resource use to perform load balancing. There are three different types of resource-aware load-balancing algorithms: bin-packing, agent-based, and dynamic cluster-based. In *bin-packing*, different bins are used to pack objects of different capacities. Bin packing uses a minimum number of bins to provide the same capacity in a balanced way. In *agent-based*, a software agent is used to monitor the performance of different components, such as network devices, storage devices, and processors, and balances the load effectively. In *dynamic cluster-based*, resources are categorized automatically based on requirements and availability of resources. Further, categorized resources are allocated for execution of workloads with maximum resource utilization and minimum energy consumption.

3.5.1.3.2 Performance-aware. In *performance-aware* load-balancing algorithms, different performance parameters are analyzed to make decisions for effective load balancing of VMs [80]. There are two different types of performance-aware load-balancing algorithms: adaptive and QoS-based. In *adaptive*, performance is maintained using a dynamic computing environment for execution of workloads with changing behavior. In *QoS-based*, resources are provisioned and scheduled for workload execution by fulfilling the QoS requirements of applications such as energy efficiency, makespan, execution cost, and response time.

3.5.1.4 VM Consolidation-Based Taxonomy. VM consolidation refers to the effective use of VMs to improve resource utilization and reduce energy consumption [49]. Based on the literature [99, 102, 159, 160], VM consolidation consists of the following components: (i) resource assignment policy, (ii) architecture, (iii) co-location criteria, and (iv) migration triggering points, as shown in Figure 9.

3.5.1.4.1 Resource assignment policy. This policy defines the mechanism to select resources for VMs within a CDC [125] and can be static or dynamic. In the *dynamic* approach, VMs are reconfigured using dynamic attributes proactively based on the demand of workloads. In the *static* approach, maximum resources are preassigned to a VM for workload execution.

3.5.1.4.2 Architecture. There are two different types of architectures used in VM consolidation techniques: centralized and decentralized, as described in Section 3.1.1.4. There is no risk of a

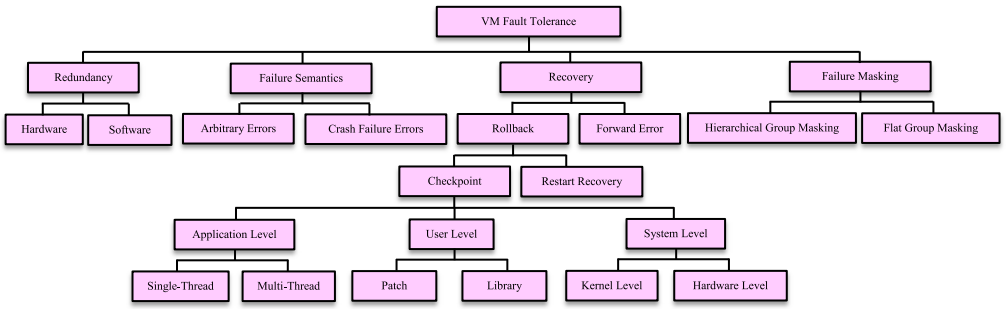


Fig. 10. Taxonomy based on VM fault tolerance.

single failure point in *decentralized* architectures, while *centralized* architectures are prone to a single failure point.

3.5.1.4.3 Co-location criteria. There are two main types of co-location criteria in VM consolidation techniques, which is considered based on resource availability and power [71]. VMs can be co-located from one CDC to another (i) if less resources are *available* in the current CDC or (ii) if there is unavailability of adequate *power* to run the CDC.

3.5.1.4.4 Migration triggering point. VMs can be migrated from one CDC to another for consolidation. The target CDC is identified using two different approaches [67]: historic data and heuristic based. In the *historic data*-based approach, the VM can be migrated to the most efficient CDC based on the historic data of previous performances. In the *heuristic*-based approach, the most efficient CDC can be identified based on performance parameters such as resource utilization, energy consumption, and response time.

3.5.1.5 VM Fault Tolerance-Based Taxonomy. VM fault tolerance supports the primary VM by maintaining the identical secondary VM to provide continuous availability of cloud service in case of VM failure. Based on the literature [105, 106, 157, 161], VM fault tolerance consists of the following components: (i) redundancy, (ii) failure semantics, (iii) recovery, and (iv) failure masking, as shown in Figure 10.

3.5.1.5.1 Redundancy. In the case of resource failure, redundancy provides redundant components to maintain the performance of the computing system, which can be software or hardware [125]. For *hardware* components, the physical redundancy technique adds redundant hardware components to tolerate failures, which support the computing system to continue its service in an efficient manner. For *software* components, two different types of processes are created: active (primary) and passive (backup). The backup process is identical to the primary process; the backup process will be active during the failure of the primary process to maintain the performance of the system.

3.5.1.5.2 Failure semantics. This refers to the selection of failure tolerance method based on the two types of failure modes [83, 100]: arbitrary errors and crash failure errors. An *arbitrary error* occurs when a communication service loses or delay messages or messages may be corrupted. A *crash failure error* occurs when a system suddenly stops processing of instructions. To deal with both type of failures, a computing system needs a duplicate processor.

3.5.1.5.3 Recovery. This mechanism replaces the erroneous state with a stable state using different recovery mechanisms [103, 122]: Forward Error Recovery (FER) and Backward Error Recovery

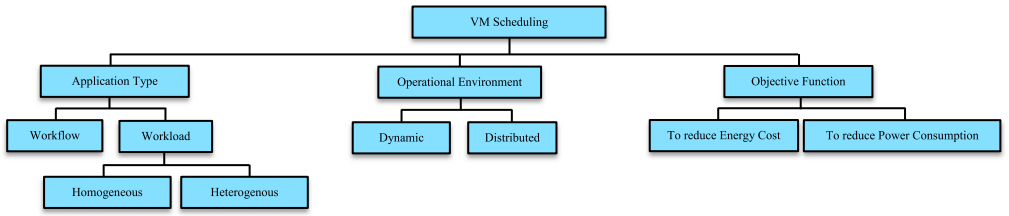


Fig. 11. Taxonomy based on VM scheduling.

(BER, or rollback). The *FER mechanism* tries to correct the errors to move the system into a new correct state and this mechanism is effective when there is a need of continue service. BER or rollback recovery is a widely used fault tolerance mechanism, which consists of two different methods: checkpoint and restart recovery. The *restart recovery* mechanism performs the process of rebooting to recover or restore the system to its correct state. To incorporate fault tolerance into the system, a snapshot of the application's state is saved so that the system can reboot from that point in case of a system crash; this process is called checkpointing. *Checkpoints* can be performed at three different levels: application, user, and system. At the *application level*, a checkpointing code is inserted automatically into the application code if failure has occurred. The checkpointing code can be written using single-thread or multi-thread programming. At the *user level*, an application program is linked to the *library*; Condo [20] and Esky [66] are library implementations. Further, the user can use *patch* to perform user-level checkpointing. At the *system* level, the process of checkpointing can be performed at the OS kernel level and hardware level. The digital hardware is used in *hardware level checkpointing* to modify a group of commodity hardware. *OS kernel* level checkpointing installs the available package for a particular OS.

3.5.1.5.4 Failure masking. The failure masking technique ensures the availability of cloud service during node failure without the user observing any interruption [144, 145]. There are two types of masking techniques: flat and hierarchical group masking. In *flat group masking*, individual workers are appearing as a single worker and hidden from the clients, and a new worker will be selected using a voting process [14] in the case of failure. In *hierarchical group masking*, a central coordinator controls the activities of different workers; the coordinator selects the new worker in the case of failure.

3.5.1.6 VM Scheduling-Based Taxonomy. The VM scheduling algorithm schedules the virtual resources (local or remote) effectively for workload execution. Based on the literature [60, 61, 40, 107, 157, 158], VM scheduling consists of the following components: (i) application type, (ii) operating environment, and (iii) objective function, as shown in Figure 11.

3.5.1.6.1 Application type. Cloud application consists of two different tasks, which need computing resources for their execution [95]: workload and workflow. A *workload* is the execution of a set of instances to achieve desired output and it can be either *homogeneous* (same QoS requirements) or *heterogenous* (different QoS requirements). *Workflow* is a combination of interrelated tasks, which distribute on different resources to achieve a single objective.

3.5.1.6.2 Operational environment. There are two types of operational environments: dynamic and distributed, or an environment can be both [143]. In a *dynamic* environment, VMs are scheduled for workload execution to reduce resource waste and energy consumption. In a *distributed* environment, optimized VMs are scheduled from different CDCs, which are distributed geographically to improve resource utilization for workload execution.

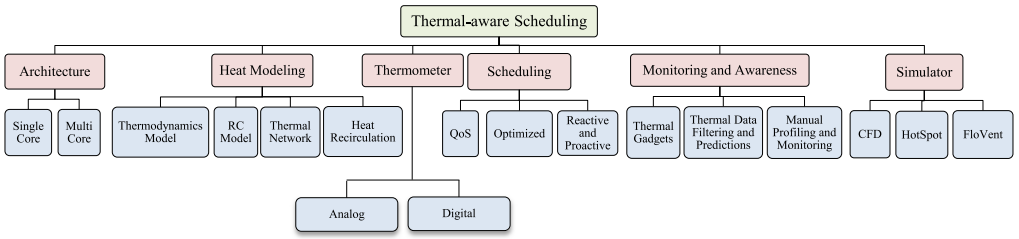


Fig. 12. Taxonomy based on thermal-aware scheduling.

3.5.1.6.3 Objective function. The literature has reported that there are two types of objective functions for VM scheduling: (1) to reduce energy cost and (2) to reduce power consumption. The *energy cost* is a combination of monetary and non-monetary costs associated with energy use for scheduling VMs [3]. The *power consumption* is the amount of electricity expended by a resource to complete the execution of an application [62].

For effective management of virtualized CDCs, thermal-aware scheduling is required to execute workloads on energy-efficient computing resources, which further reduce the heat recirculation and therefore the load on the cooling systems.

3.6 Thermal-Aware Scheduling

CDCs consist of a chassis and racks to place the servers to process the IT workloads. To maintain the temperature of datacenters, cooling mechanisms are used to reduce heat [86]. Thus, there is a need for effective management of temperature to run the CDC efficiently. Servers produce heat during execution of IT workload; thus, cooling management is required to keep room temperature stable [92]. The processor is an important component of a server and consumes the most electricity. Sometimes the heat generation of processors is higher than the threshold because servers are organized in a compact manner [93]. Both cooling and computing mechanisms consume a huge amount of electricity. It would be better to reduce the energy consumption instead of improving the cooling mechanism [90]. To solve the heating problem of CDCs, thermal-aware scheduling is designed to minimize cooling setpoint temperature, hotspots, and thermal gradients. Thermal-aware scheduling is better than heat modeling [142]. Thermal-aware scheduling based on heat modeling performs computational scheduling of workload. Thermal-aware monitoring and profiling module monitors and assess the distribution of heat in CDCs while profiling maintains the details of computational workload, microprocessors, and heat emission of servers. With the use of renewable energy, the load of cooling can be decreased to enable sustainable CDCs. The evolution of thermal-aware scheduling techniques (see Figure 26) and their comparison along with open research challenges [50, 53, 92, 93, 86, 94, 97, 98, 91, 85, 89, 90] can be found in Table 18 of Appendix C.6.

3.6.1 Thermal-Aware Scheduling-Based Taxonomy. The components of thermal-aware scheduling are (i) architecture, (ii) heat modeling, (iii) a thermometer, (iv) scheduling, (v) monitoring and awareness, and (vi) a simulator, as shown in Figure 12. Each of these taxonomy elements are discussed below, along with relevant examples. The comparison of existing techniques (discussed in Appendix C.6) based on our thermal-aware scheduling taxonomy is given in Table 19 of Appendix C.6.

3.6.1.1 Architecture. Thermal-aware scheduling techniques have been designed based on two different architectures: single core and multi-core [86, 92, 93]. Thermal-aware scheduling techniques execute workloads based on their priorities at different processor speeds for *single-core*

architecture, and for execution of a high-priority workload, the current workload can be preempted. Generally, high-priority workloads are running at high speed and the temperature of the processor can reach its threshold value. To optimize the temperature of the processor, low-priority workloads are running at a lower speed to cool down the processor. To improve the execution of thermal-aware scheduling, a *multi-core* processor is used, in which a task is divided into a number of threads and independent threads are running on different cores based on their priorities. Multi-core processors are designed with thermal-aware aspects such as intelligent fan control, clock gating, and frequency scaling. These aspects are working in coordination to control the temperature within its operating limits. If one core is getting hot, then a thread can be transferred to another cooler core to maintain the temperature.

3.6.1.2 Heat-modeling. It is an effective mechanism in thermal-aware scheduling to develop a relationship between eventual heat dissipation and energy consumed by computing devices. The scope of heat models is defined based on evaluation of environmental variables such as temperature, air pressure, and power. The selection of heat model also affects energy efficiency. The types of heat models used in the literature are (i) the thermodynamics model, (ii) RC model, (iii) thermal network, and (iv) heat recirculation. The *thermodynamics model* is used to explore the heat exchange mechanisms in CDCs. The value of heat is quantified using the law of energy conservation [89, 90]. The thermodynamic process produces the details of heat emissions and energy consumption and passes cold air to remove heat from the datacenter. Researchers are still working on this process for further optimization. The *RC model* is basically a resistor–capacitor (RC) circuit that forges a relationship between electrical phenomena of the RC circuit and heat transfer. Temperature difference between two surfaces and energy consumption is used to determine the value of R and C for conductance and convention. The value of RC is not changed after manufacturing the processor package. The RC model is used to determine the value of various thermal parameters. The *thermal network* is based on both the RC model and thermodynamics model. In a thermal network, every node of a CDC belongs to one of the networks, which can be an IT network or cooling network. A server executes a workload by consuming energy and producing heat and the server is part of both the cooling and IT networks. The thermal network is efficient for heat modeling of heterogenous equipment of a datacenter. *Heat recirculation* deals with mixing hot air (coming from server outlets) and cold air (coming from the cooling manager). The temperature of cold air is changing with time after entering into CDCs. To maintain the temperature of a CDC, it is a great challenge to provide the uniform cold air temperature every time. The resource utilization of servers that participates in heat recirculation will be reduced and performance of CDCs is also affected in terms of QoS.

3.6.1.3 Thermometer. This is a device that is used to measure the temperature of CDCs. Two types of thermometers have been identified from the literature [14, 86, 92]: digital and analog. The *digital* or infrared thermometer is an electronic device that uses a digital sensor to provide a digital display. Most digital thermometers are resistive thermal devices that uses a function of electrical resistance to measure temperature variations. The *analog* thermometer contains alcohol, which falls or rises as it contracts or expands with temperature variations. Temperature value is displaying in degrees Celsius or Fahrenheit, which is marked on a glass capillary tube.

3.6.1.4 Scheduling. The energy consumed by CDCs is used for execution of workloads, but it is dissipated as heat. Lower energy is used to remove heat while workloads are scheduling using thermal-aware aspects. Thermal profiles of thermal-aware schedulers are used to determine the resource with minimum dissipation of heat in CDCs. The aim of thermal-aware scheduling is to reduce dissipation of heat from active servers and minimize the active servers by turning off idle

servers. Three types of thermal-aware scheduling used in the literature [85, 86, 89, 92] are (i) QoS, (ii) optimized, and (iii) reactive and proactive. *QoS-based thermal-aware scheduling* schedules the energy-efficient resources to improve the performance of the CDC. The scheduler controls the temperature and reduces the load of overcooling using dynamic thermal management techniques. Further, a challenge of maintaining the SLA based on these QoS parameters is introduced and requires the trade-off between cost saving and compensation or penalty in the case of SLA violations. *Optimized thermal-aware scheduling* schedules workloads using the concept of autonomic computing. These techniques are basically a combination of heat-recirculation and thermal-aware techniques. The main aim of server-based scheduling techniques is to reduce the peak inlet temperature, which is increased by heat recirculation. Heat recirculation can be minimized by placing lesser workloads on servers that are nearer to the floor. Processor-based scheduling techniques execute the workloads by sustaining the steady core temperature, called *throttling*. Earlier, workloads are executed using zig-zag schemes till a temperature threshold is achieved. *Reactive* management works based on feedback methods and manages the temperature based on their current state to maintain its temperature. Continuous monitoring of thermal-aware scheduling is needed to determine whether the temperature is lower than its threshold value or not. If the temperature is higher than a threshold value, then corrective actions will be taken to make it stable. The *proactive* approach manages the resources based on the prediction and assessment of temperature and thermal profiling. Based on previous data, predictions have been identified and required action is planned to reduce temperature during scheduling.

3.6.1.5 Monitoring and Awareness. Thermal monitoring and awareness is used to perform thermal-aware scheduling decisions. The thermal profile is created based on resultant heat dissipation and power consumption for thermal awareness, which is used to rank the servers for future scheduling decisions. There are three different methods of thermal monitoring and awareness, as identified from the literature [14, 32, 85, 86]: (i) manual profiling and monitoring, (ii) thermal gadgets, and (iii) thermal data filtering and predictions. In *manual profiling and monitoring*, heat generation and recirculation and power consumption of individual servers are noted manually to create a thermal profile. If there are no real data available, then simulation tools can be used for manual profiling. Some thermal-aware scheduling techniques [89, 92, 93, 97] estimate the thermal index to evaluate the efficiency of different CDCs and perform their ranking. *Thermal gadgets* such as thermal cameras and sensors are used to generate accurate and timely thermal information automatically. Multiple sensors can be used per unit area and both onboard and external thermal sensors can be used to collect thermal information. In *thermal data filtering and predictions*, a rise in temperature and resulting heat can be predicted for proactive thermal-aware scheduling, which helps to make effective decisions to minimize thermal gradient and peak outlet temperature. The advance prediction of temperature and heat can help to maintain the QoS during workload execution.

3.6.1.6 Simulator. The results of thermal simulators can be used to create thermal profiles. There are three different simulators identified from the literature [14, 86, 92, 14, 32, 85, 86]: (i) CFD, (ii) HotSpot, and (iii) FloVent. The *Computational Fluid Dynamics (CFD)* simulator is used to analyze and optimize airflow and heat transfer for CDCs to create the thermal profile, which further helps to create a thermal map. *HotSpot* is a temperature modeling tool [14], which uses thermal resistances to design the architecture of CDCs based on power density and hence cooling costs, which are rising exponentially. The *FloVent* simulator [14] is used to predict contamination distribution, heat transfer, and 3D airflow for different types of CDCs, which mainly focuses on air conditioning and ventilating systems.

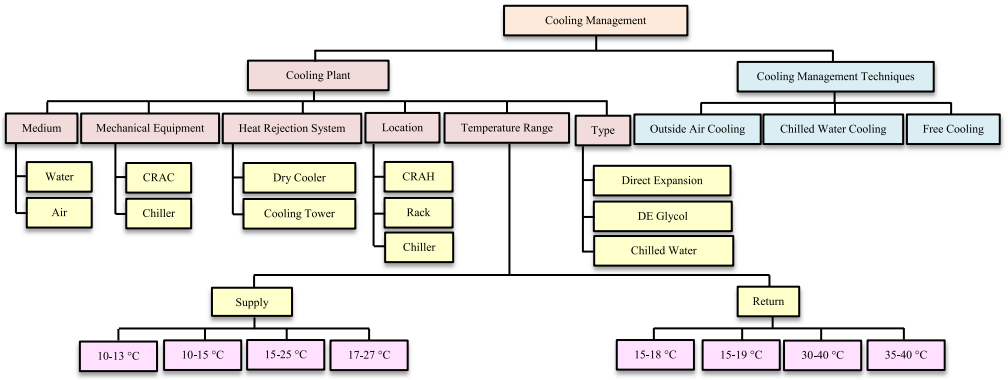


Fig. 13. Taxonomy based on cooling management.

Effective cooling mechanisms are needed to maintain the temperature of CDCs to enable sustainable cloud computing.

3.7 Cooling Management

The increasing demand for computation, networking, and storage expands the complexity, size, and energy density of CDCs exponentially, which consumes a large amount of energy and produces a huge amount of heat [14]. To make CDCs more energy efficient and sustainable, we need an effective cooling management system, which can maintain the temperature of CDCs [21]. Heat dissipation is a critical factor to be considered for cooling management of CDCs, which affects the reliability and availability of the cloud service. In cloud datacenter CDCs, high heat density causes high temperature, which needs to be controlled for smooth functioning of CDCs [86]. Effective cooling management can attain complete environmental control, including pollution concentration, humidity, and air temperature [92]. Thus, it is necessary to discuss the existing and emerging technologies for datacenter cooling systems to determine the effective approach to maintaining CDCs working in a safe and reliable manner. The evolution of cooling management techniques (see Figure 27) and their comparison along with open research challenges [150-155] are provided in Table 20 of Appendix C.7.

3.7.1 Cooling Management-Based Taxonomy. Based on the literature [150-155], cooling management consists of the following components: (i) cooling management techniques and (ii) the cooling plant as shown in Figure 13. Each of these taxonomy elements are discussed below along with relevant examples. The comparison of existing techniques (discussed in Appendix C.7) based on our cooling management taxonomy is given in Table 21 of Appendix C.7.

3.7.1.1 Cooling Plant. The cooling plant is a system that provides cooling to space where the CDC is placed and consists of the following components: (i) medium, (ii) mechanical equipment, (iii) a heat rejection system, (iv) location, (v) type, and (vi) temperature. The cooling system uses two different types of mediums to produce cooling: water and air. The *water-based* cooling system uses a water pumping mechanism to generate cooling, while the *air-based* cooling system uses an air compressor mechanism to produce cooling. *Mechanical equipment* is used to maintain the humidity, air distribution, and temperature in the CDC. Two different types of mechanical equipment are used in cooling systems: Computer Room Air Conditioning (CRAC) and chiller. The *Heat Rejection System (HRS)* performs the process of heat removal via two methods: dry cooler and cooling tower. Different types of temperature range are established for different locations in cooling

systems [14, 86, 92, 150, 155]; location can be (1) chiller, (2) rack, and (3) Computer Room Air Handler (CRAH). There are two types of temperature classification for three different locations with different temperature ranges: (1) supply temperature and (2) return temperature. The different *types* of cooling plants are (1) Direct Expansion (DE) air-cooled systems, (2) DE glycol-cooled systems, and (3) chilled water systems [152, 154]. The DE air-cooled system contains CRAC and an air-cooled condenser as a HRS. In DE glycol-cooled systems, a glycol mixture is used as heat transfer fluid from the CRAC to the dry cooler. In the chilled water system, a chiller provides cold water to the CRAH.

3.7.1.2 Cooling Management Techniques. The literature [14, 86, 92, 150, 155] identified three different types of cooling management techniques: (i) outside air cooling, (ii) chilled water cooling, and (iii) free cooling. In *outside air cooling*, the cooler is used to bring the fresh air from outside and cooled and pushed it through the CRAC, which is better than an air recirculation mechanism. In the *chilled water cooling* system, electricity is used to freeze water at night and circulate this water throughout the CRAC unit during the day. In *free cooling*, air is passed into a chamber, which performs cooling through water evaporation [14].

There is a need to maximize the use of renewable energy for cooling, which further reduces carbon footprints and environmental problems.

3.8 Renewable Energy

Sustainable computing needs energy-efficient workload execution by using renewable energy resources to reduce carbon emissions [117]. Fossil fuels such as oil, gas, and coal generate brown energy, which produces carbon-dioxide emissions in large quantities. Green energy resources such as sun, wind, and water generate energy with nearly zero carbon-dioxide emissions [121]. One type of green energy is hydroelectricity, which is produced using hydraulic power. Wind and solar energy can be purchased from off-site companies or can be generated using on-site equipment [118]. In the next decade, the cost/watt will be reduced by half for renewable energy due to following: (i) government organizations provide monetary incentives for the incorporation of resources of renewable energy, (ii) the storage capacity of rechargeable batteries will be increased, and (iii) advancement in technology to improve capacity of materials such as photovoltaic arrays [124]. Workload migration and energy-aware load-balancing techniques addressed the issue of unpredictability in the supply of renewable energy. To achieve 100% availability of cloud services, adopting hybrid designs of energy generation is recommended, which use energy from renewable resources and grid resources [117]. Mostly, sites of commercial CDCs are located away from abundant renewable energy resources. Consequently, portable CDCs are placed nearer to renewable energy sources to make them cost-effective. Dynamic load-balancing technique and renewable energy-based workload migration are discussed in Appendix C.8. The evolution of techniques for renewable energy (see Figure 28) and their comparison along with open research challenges [117-119, 121, 124, 126, 148] can be found in Table 22 of Appendix C.8.

3.8.1 Renewable Energy-Based Taxonomy. Based on the literature [8], renewable energy consists of the following components: (i) workload scheduling, (ii) focus, (iii) source of energy, (iv) location-aware and (v) storage devices, as shown in Figure 14. These taxonomy elements are discussed below along with relevant examples. The comparison of existing techniques (discussed in Appendix C.8) based on our renewable energy taxonomy is given in Table 23 of Appendix C.8.

3.8.1.1 Workload Scheduling. The scheduling of workloads in renewable energy-aware techniques has been done in two ways: (i) dynamic load balancing and (ii) power preserving. *Dynamic load balancing* is the most well-known approach to make a balance between renewable energy

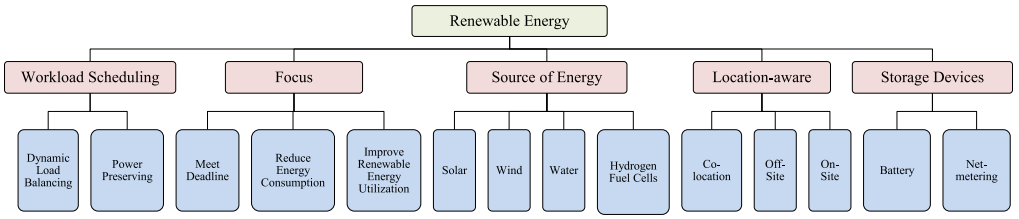


Fig. 14. Taxonomy based on renewable energy.

and grid energy. These techniques supply renewable energy to execute workloads efficiently and predict the amount of energy that can be produced to run a CDC and the amount of energy that is needed to execute the workloads using that energy at the demand side.

There is a great need for renewable energy for deadline-oriented workloads [121, 124]. On the other side, workloads are scheduled using server *power-preserving* techniques. These techniques use power transition and voltage scaling to run a suitable workload using available renewable energy to balance demand-supply. Further, DVFS-based power-preserving techniques are also designed to control the energy based on operational frequency along with voltage scaling. In these approaches [117, 118, 119, 123], the workload (web application) requests are distributed to the specific datacenter by matching the workload demand with the available renewable energy across geo-dispersed CDCs by using a load-balancing algorithm. This approach follows two levels of load balancing: (i) at the local level, redirecting the request within web servers in a datacenter, known as local load balancing; and (ii) at the global level, redirecting the requests among local load balancers related with a CDC, known as global load balancing. Each datacenter has an autoscaler in addition to a local load balancer that adds/removes web servers dynamically in response to the request [124]. The incoming request is distributed among a geo-dispersed CDC based on the place that has a higher availability of renewable energy so that maximum renewable energy is used for making the datacenter sustainable. In the case of not having enough renewable energy, the request is redirected to the location having cheap brown electricity. The global load balancer, uses a "weighted round robin" load-balancing algorithm to redirect the requests.

3.8.1.2 Focus. There are three main objectives of renewable energy-aware techniques, to (i) meet deadline, (ii) reduce energy consumption, and (iii) improve renewable energy utilization [121, 126, 139]. The SLA is an important component and workload should be executed without violation of the SLA. Cloud providers are mainly focused on the *deadline* of the workload during execution. Other renewable energy-aware techniques focus on minimizing *power usage* of CDCs to execute workloads. Further, renewable energy can be used effectively while placing the CDC nearer to the source of *renewable energy* to save more energy and used to process more work.

3.8.1.3 Source of energy. There are four different kinds of energy sources as identified from the literature [118, 117]: (i) solar, (ii) wind, (iii) water, and (iv) hydrogen fuel cells. The renewable energy can be generated using sunlight or it can be generated using *wind* to run a generator to produce electricity. Some techniques use the combination of both solar and wind. Other sources of renewable energy can be *water* as well as *hydrogen fuel cells* [117, 126].

3.8.1.4 Location-aware. In renewable energy generation, energy can be stored using three different localities [121, 124, 127]: (i) on-site, (ii) off-site, and (iii) co-location. In an *on-site* locality, use of renewable energy is done at the same place where energy is produced. *Off-site*, the place of renewable energy use is different than the place generating energy, which means that energy can

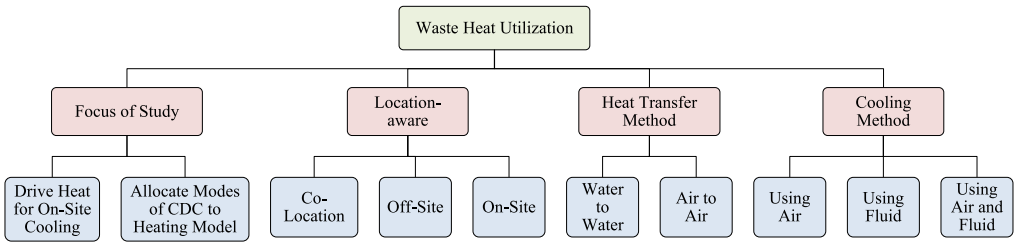


Fig. 15. Taxonomy based on waste heat utilization.

be transported to an off-shore site. On the other hand, CDCs are *co-located* from different places to sites where the chances of renewable energy use exist.

3.8.1.5 Storage devices. There are two main storage devices used by renewable energy-aware techniques to store energy [118, 119, 128]: battery and net-metering. Lithium ion *batteries* are using to store energy effectively. *Net-metering* is another device that can be used to store generated energy for the future. Waste heat can be another source of renewable energy, which can be used in an efficient manner that generates electricity or can be used for heating houses and greatly reduce electricity costs and carbon emissions.

3.9 Waste Heat Utilization

Reuse of waste heat is becoming a solution for fulfilling energy demand in energy conservation systems because fossil fuel deposits are quickly dwindling. Cooling management is necessary to maintain the temperature of CDCs in operational range due to generation of large amounts of heat during energy consumption. The cooling mechanism of CDCs consumes large amounts of electricity: 40% to 50% [3, 71]. Power densities of servers are increased by using stacked and multi-core server designs, which further increases cooling costs. The energy efficiency of CDCs may be improved by reducing the energy used in cooling. There is a need to change the location of CDCs to reduce cooling costs, which can be done through placing the CDCs in an area that has free cooling resources. Due to consumption of large amounts of energy, CDCs are acting as a heat generator [129, 130]. The vapor-absorption-based cooling systems of CDCs can use waste heat, then remove the heat while evaporating. Vapor-absorption-based free cooling mechanisms can make the value of PUE ideal by neutralizing cooling expenses. Low-temperature areas can use the heat generated by CDCs for heating facilities. The literature reports [3, 131] that there are two main solutions to control the temperature of CDCs: (1) relocation of CDCs to nearby waste heat utilization recovery places, and (2) vapor-absorption-based cooling systems. The two waste heat utilization techniques—Air Recirculation and Power Plant Co-location—are discussed in Appendix C.9. The evolution of waste heat utilization techniques (see Figure 29) and their comparison along with open research challenges [130-134, 147] can be found in Table 24 of Appendix C.9.

3.9.1 Waste Heat Utilization-Based Taxonomy. Based on the literature, waste heat utilization consists of the following components: (i) focus of study, (ii) location-aware, (iii) heat transfer method, and (iv) cooling method, as shown in Figure 15. These taxonomy elements are discussed below along with relevant examples. The comparison of existing techniques (discussed in Appendix C.9) based on our waste heat utilization taxonomy is given in Table 25 of Appendix C.9.

3.9.1.1 Focus of study. Existing waste heat utilization techniques focus on two different ways to utilize heat: (i) vapor-absorption-based cooling systems and (ii) give heat to co-located datacenter buildings [130, 131]. The first way is utilizing heat for *on-site cooling* using vapor absorption, in

which heat is generated during the execution of workloads. The second way is to distribute the heat generated from CDCs to the *heating model* using different modes of transfer. Heat modeling is an effective mechanism in thermal-aware scheduling to develop a relationship between eventual heat dissipation and energy consumed by computing devices.

3.9.1.2 Location-aware. In waste heat utilization, heat can be recovered using three different localities: (i) on-site, (ii) off-site, and (iii) co-location, as described in Section 3.8.1.4.

3.9.1.3 Heat transfer method. There are two different methods available for transferring heat: (i) water to water and (ii) air to air [133]. The *water-to-water* heat transfer method is based on a refrigerator mechanism, in which heat is transferred from the source side to the load side using conditioned fluid (hot or cold). Boiler or cooler can be used at both sides of the exchanger based on the purpose of the transfer. The *air-to-air* heat transfer method is based on vapor compression refrigeration, which uses reverse-cycle air conditioners to transfer heat from one place to another.

3.9.1.4 Cooling method. As identified from the literature, there are three types of cooling methods used in existing waste heat utilization techniques: (i) using air, (ii) using water, and (iii) using air and water [132, 134]. An evaporative cooler is a device that uses evaporation of water to cool *air* and it is based on vapor-compression refrigeration cycles. On the other hand, the cooling effect is produced by consumption of *water* through evaporation. Both water- and air-based cooling mechanisms are used by WHU techniques.

4 OUTCOMES

The outcomes of this systematic review are discussed in Appendix D.

5 OPEN CHALLENGES AND FUTURE DIRECTIONS: A SUMMARY

We surveyed 142 research papers in this systematic review and presented them in a categorized manner. The focus of our systematic review is broader than the existing surveys, as discussed in Table 1 of Appendix A. This survey used methodical survey technique to conduct a systematic review and comprises the most recent research related to sustainable cloud computing. In addition to the nine categories of sustainable cloud computing, we covered the other research issues related to the sustainability of emerging technologies, such as Internet of Things and smart cities. A systematic methodology has been used to develop an evolution of categories of sustainable cloud computing that identifies optimization parameters, metrics, open issues, and Focus of Study (FoS). We explored and compared the existing techniques based on the proposed taxonomy. We documented the research issues addressed and open challenges that are still unresolved in sustainable cloud computing and discussed in Appendix E.

5.1 Open Challenges

The identified various open challenges of sustainable cloud computing are discussed in Appendix E.1.

5.2 Implications for Research and Practice

The implications for research and practice are discussed in Appendix E.2.

5.3 Integrated: Sustainability vs. Reliability

The trade-off between sustainability and reliability is discussed in Appendix E.3.

5.4 Emerging Trends and Their Impact

The emerging trends and their impact are discussed in Appendix E.4.

6 SUSTAINABLE CLOUD COMPUTING ARCHITECTURE: A CONCEPTUAL MODEL

The conceptual model for sustainable cloud computing is discussed in Appendix F.

7 SUMMARY AND CONCLUSIONS

The use of large numbers of CDCs results in a huge amount of energy consumption and produces significant amounts of large carbon footprints, which has become the greatest challenge of the 21st century. On the other hand, the use of a combination of grids and renewable energy to run CDCs in smart cities can save energy to a large extent. Consequently, there is a need to manage both energy and QoS together to enable sustainable and energy-efficient cloud services. Existing energy-aware resource management techniques and policies mainly focus on VM consolidation to reduce energy consumption of servers only. However, other resources, such as networks, storage, memory, and cooling, consume a huge amount of energy. Efficient scheduling of traffic flow between servers in CDCs is necessary to save energy. Therefore, holistic management of all resources (networks, memory, processors, cooling, and storage) is required to enable sustainable cloud computing. Further, the effect of QoS on the SLA must be addressed in holistic management techniques. Moreover, self-aware or autonomic management of cloud resources in a holistic manner can manage both energy consumption and QoS simultaneously, which can improve the sustainability of cloud computing systems. In addition, dynamically changing the variable clock rates of processors can must optimize energy use. It has also been recommended that the concept *follow the renewable* can motivate cloud providers to locate their CDCs nearer to green energy resources and load can be distributed geographically. However, geographical distribution of resources affects the QoS of networks, which is an open research challenge for the community. Unfortunately, the need to process a huge amount of data and provide high performance simultaneously can also consume large amounts of energy. To solve this problem, energy consumption, SLAs, and QoS must be managed at same time. Further, there is a need for self-aware management of cloud resources holistically to address these research issues. Currently, the research community is working in this direction, but more advanced research is required to ensure the energy efficiency and sustainability of cloud services. In this article, we proposed a taxonomy of sustainable cloud computing to analyze existing techniques for sustainability, including application design, sustainability metrics, capacity planning, energy management, virtualization, thermal-aware scheduling, cooling management, renewable energy, and waste heat utilization for CDCs. Further, the taxonomy mapping-based comparison has been described. A conceptual model for sustainable cloud computing has been proposed. Through a detailed analysis of related studies in the context of taxonomy, we are able to identify and propose various future research directions.

We assert the following conclusions:

- VM consolidation techniques can minimize energy consumption of servers.
- Optimization scheduling of traffic flows between servers is required.
- There is a need for dynamic task scheduling for energy and QoS optimization.
- New system architectures and algorithms can geographically distribute the CDC.
- There is a need for interplay between IoT-enabled cooling systems and the CDC manager.
- Maximum use of renewable energy–powered resources is required for holistic management of resources and workloads.

We hope that this systematic review will be helpful for practitioners and researchers who want to pursue research in any area of sustainable cloud computing.

ELECTRONIC APPENDIX

The electronic appendix (**A:** Background, **B:** Review Methodology, **C:** Elements, Evolution, Comparison, and Open Research Issues, **D:** Outcomes, **E:** Open Challenges and Future Directions: A Summary, **F:** Sustainable Cloud Computing Architecture: A Conceptual Model and **G:** 360-Degree View of Taxonomy) for this article can be accessed in the ACM Digital Library.

ACKNOWLEDGMENTS

We would like to thank all anonymous reviewers for their valuable comments and suggestions for improving the article. We thank Patricia Arroba, Minxian Xu, and Shashikant Ilager for their useful suggestions.

REFERENCES

[1] Rajkumar Buyya and Sukhpal Singh Gill. 2018. Sustainable Cloud Computing: Foundations and Future Directions. *Business Technology & Digital Transformation Strategies, Cutter Consortium* 21, 6 (2018), 1–10.

[2] Toni Mastelic, Ariel Oleksiak, Holger Claussen, Ivona Brandic, Jean-Marc Pierson, and Athanasios V. Vasilakos. 2015. Cloud computing: Survey on energy efficiency. *ACM Computing Surveys* 47, 2 (2015), 1–33.

[3] Junaid Shuja, Abdullah Gani, Shahaboddin Shamshirband, Raja Wasim Ahmad, and Kashif Bilal. 2016. Sustainable cloud datacenters: a survey of enabling techniques and technologies. *Renewable and Sustainable Energy Reviews* 62 (2016), 195–214.

[4] Sukhpal Singh Gill, Inderveer Chana, Maninder Singh and Rajkumar Buyya. 2018. RADAR: Self-Configuring and Self-Healing in Resource Management for Enhancing Quality of Cloud Services, Concurrency and Computation: Practice and Experience (CCPE), 2018. Retrieved November 24, 2018 from http://buyya.com/papers/RADAR-Cloud-CCPE.pdf. DOI: https://doi.org/10.1002/cpe.4834

[5] Massimo Ficco and Massimiliano Rak. 2016. Economic denial of sustainability mitigation in cloud computing. In *Organizational Innovation and Change*. Springer, Cham, 229–238.

[6] Xiang Li, Xiaohong Jiang, Peter Garraghan, and Zhaohui Wu. 2018. Holistic energy and failure aware workload scheduling in Cloud datacenters. *Future Generation Computer Systems* 78 (2018), 887–900.

[7] Fereydoun Farrahi Moghaddam and Mohamed Cheriat. 2015. Sustainability-aware cloud computing using virtual carbon tax. 2015. arXiv preprint arXiv:1510.05182 (2015).

[8] Josep Subirats and Jordi Guitart. 2015. Assessing and forecasting energy efficiency on Cloud computing platforms. *Future Generation Computer Systems* 45 (2015), 70–94.

[9] Zhou Zhou, Zhi-gang Hu, Tie Song, and Jun-yang Yu. 2015. A novel virtual machine deployment algorithm with energy efficiency in cloud computing. *Journal of Central South University* 22, 3 (2015), 974–983.

[10] Claudio Fiandrino, Dzmitry Kliazovich, Pascal Bouvry, and Albert Zomaya. 2017. Performance and energy efficiency metrics for communication systems of cloud computing datacenters. *IEEE Transactions on Cloud Computing* 5, 4 (2017), 738–750.

[11] Yogesh Sharma, Bahman Javadi, and Weisheng Si. 2015. On the reliability and energy efficiency in cloud computing. In *Proceedings of the 13th Australasian Symposium on Parallel and Distributed Computing*, Parramatta, Sydney, Australia. 111–114.

[12] Dario Pompili, Abolfazl Hajisami, and Tuyen X. Tran. 2016. Elastic resource utilization framework for high capacity and energy efficiency in cloud RAN. *IEEE Communications Magazine* 54, 1 (2016), 26–32.

[13] Dejene Boru, Dzmitry Kliazovich, Fabrizio Granelli, Pascal Bouvry, and Albert Y. Zomaya. 2015. Energy-efficient data replication in cloud computing datacenters. *Cluster Computing* 18, 1 (2015), 385–402.

[14] Muhammad Tayyab Chaudhry, Teck Chaw Ling, Atif Manzoor, Syed Asad Hussain, and Jongwon Kim. 2015. Thermal-aware scheduling in green datacenters. *ACM Computing Surveys (CSUR)* 47, 3 (2015), 1–39.

[15] Konstantinos Domdouzis. 2015. Sustainable cloud computing. In *Green Information Technology: A Sustainable Approach*, Mohammad Dastbaz, Colin Pattinson and Babak Akhgar (Eds.). Elsevier, USA, 95–110.

[16] Zahra Abbasi. 2014. *Sustainable Cloud Computing*. PhD. Dissertation. Arizona State University, Tempe, AZ.

[17] Accenture. 2010. Cloud Computing and Sustainability: The Environmental Benefits of Moving to the Cloud. Online Available at https://download.microsoft.com/download/A/F/F/AFFEB671-FA27-45CF-9373-0655247751CF/Cloud%20Computing%20and%20Sustainability%20-%20Whitepaper%20-%20Nov%202010.pdf.

[18] Prasanna N. L. N. Balasooriya, Santoso Wibowo, and Marilyn Wells. 2016. Green cloud computing and economics of the cloud: Moving towards sustainable future. *GSTF Journal on Computing (JoC)* 5, 1 (2016), 15–20.

[19] Arlitt Martin, Cullen Bash, Sergey Blagodurov, Yuan Chen, Tom Christian, Daniel Gmach, Chris Hyser, et al. 2012. Towards the design and operation of net-zero energy data centers. In *Proceedings of the 13th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm'12)*. IEEE, 552–561.

[20] Francesco Bifulco, Marco Tregua, Cristina Caterina Amitrano, and Anna D'Auria. 2016. ICT and sustainability in smart cities management. *International Journal of Public Sector Management* 29, 2 (2016), 132–147.

[21] Alfonso Capozzoli and Giulio Primiceri. 2016. Cooling systems in data centers: state of art and emerging technologies. *Energy Procedia* 83 (2015), 484–493.

[22] Soundararajan Vijayaraghavan and Joshua Schnee. 2017. Sustainability as a first-class metric for developers and end-users. *ACM SIGOPS Operating Systems Review* 51, 1 (2017), 60–66.

[23] Ana Carolina Riekstin, Bruno Bastos Rodrigues, Kim Khoa Nguyen, Tereza Cristina Melo de Brito Carvalho, Catalin Meirosu, Burkhard Stiller, and Mohamed Cheriet. 2017. A survey on metrics and measurement tools for sustainable distributed cloud networks. *IEEE Communications Surveys & Tutorials* 20, 2 (2017), 1244–1270.

[24] Ryan Bradley, I. S. Jawahir, Niko Murrell, and Julie Whitney. 2017. Parallel design of a product and Internet of Things architecture to minimize the cost of utilizing big data (BD) for sustainable value creation. *Procedia CIRP* 61 (2017), 58–62.

[25] Ruben Van den Bossche, Kurt Vanmechelen, and Jan Broeckhove. 2013. Online cost-efficient scheduling of deadline-constrained workloads on hybrid clouds. *Future Generation Computer Systems* 29, 4 (2013), 973–985.

[26] Cinzia Cappiello, Paco Melia, Barbara Pernici, Pierluigi Plebani, and Monica Vitali. 2014. Sustainable choices for cloud applications: A focus on CO_2 emissions. In *Proceedings of the 2nd International Conference on ICT for Sustainability (ICT4S'14)*. 352–358.

[27] Charith Perera and Arkady Zaslavsky. 2014. Improve the sustainability of Internet of Things through trading-based value creation. In *Proceedings of the World Forum on Internet of Things (WF-IoT)*. IEEE, 135–140.

[28] Altino M. Sampaio and Jorge G. Barbosa. 2016. Energy-efficient and SLA-based resource management in cloud data centers. *Advances in Computers, Elsevier* 100 (2016), 103–159.

[29] Charr Jean-Claude, Raphael Couturier, Ahmed Fanfakh, and Arnaud Giersch. 2015. Energy consumption reduction with DVFS for message passing iterative applications on heterogeneous architectures. In *Proceedings of the IEEE International Parallel and Distributed Processing Symposium Workshop (IPDPSW'15)*. IEEE, 922–931.

[30] Sukhpal Singh Gill and Rajkumar Buyya. 2018. SECURE: Self-protection approach in cloud resource management. *IEEE Cloud Computing* 5, 1 (2018), 60–72.

[31] Ying Zuo, Fei Tao, and A. Y. C. Nee. 2018. An Internet of Things and cloud-based approach for energy consumption evaluation and analysis for a product. *International Journal of Computer Integrated Manufacturing* 31, 4-5 (2018), 337–348.

[32] Sukhpal Singh and Inderveer Chana. 2016. QoS-aware autonomic resource management in cloud computing: a systematic review. *ACM Computing Surveys (CSUR)* 48, 3 (2016), 1–48.

[33] Sukhpal Singh Gill and Rajkumar Buyya. 2018. Failure management for reliable cloud computing: A taxonomy, model and future directions. *IEEE Computing in Science and Engineering* 20, 4 (2018), 1–15.

[34] Li Xiang, Peter Garraghan, Xiaohong Jiang, Zhaohui Wu, and Jie Xu. 2018. Holistic virtual machine scheduling in cloud datacenters towards minimizing total energy. *IEEE Transactions on Parallel and Distributed Systems* 29, 6 (2018), 1317–1331.

[35] NoviFlow Inc. 2012. Green SDN: Software Defined Networking in sustainable network solutions. (2012), 1–7. Online Available at https://noviflow.com/resource/green-sdn-software-defined-networking-in-sustainable-network-solutions/.

[36] V. Dinesh Reddy, Brian Setz, G. Subrahmanya, V. R. K. Rao, G. R. Gangadharan, and Marco Aiello. 2017. Metrics for sustainable data centers. *IEEE Transactions on Sustainable Computing* 2, 3 (2017), 290–303.

[37] Hui Zhao, Jing Wang, Feng Liu, Quan Wang, Weizhan Zhang, and Qinghua Zheng. 2018. Power-aware and performance-guaranteed virtual machine placement in the cloud. *IEEE Transactions on Parallel and Distributed Systems* 29, 6 (2018), 1385–1400.

[38] Dirk Pesch, Susan Rea, J. Ignacio Torrens Galdiz, V. Zavrel, J. L. M. Hensen, Diarmuid Grimes, Barry O'Sullivan, et al. 2017. Globally optimised energy-efficient datacenters. In *ICT-Energy Concepts for Energy Efficiency and Sustainability*. Giorgos Fagas, Luca Gammaitoni, and John P. Gallagher (Eds.). IntechOpen, UK.

[39] Min Chen, Yujun Ma, Jeungeun Song, Chin-Feng Lai, and Bin Hu. 2016. Smart clothing: Connecting human with clouds and big data for sustainable health monitoring. *Mobile Networks and Applications* 21, 5 (2016), 825–845.

[40] Sambit Kumar Mishra, Deepak Puthal, Bibhudatta Sahoo, Prem Prakash Jayaraman, Song Jun, Albert Y. Zomaya, and Rajiv Ranjan. 2018. Energy-efficient VM-placement in cloud data center. *Sustainable Computing: Informatics and Systems* (2018). DOI : https://doi.org/10.1016/j.suscom.2018.01.002

[41] Claudia Battistelli, Padraic McKeever, Stephan Gross, Ferdinanda Ponci, and Antonello Monti. 2018. Implementing energy service automation using cloud technologies and public communications networks. In *Sustainable Cloud and Energy Services*. Wilson Rivera (Ed.). Springer. 49–84.

[42] Jong Hyuk Park, Hyun-Woo Kim, and Young-Sik Jeong. 2014. Efficiency sustainability resource visual simulator for clustered desktop virtualization based on cloud infrastructure. *Sustainability* 6, 11 (2014), 8079–8091.

[43] Kai Ding, Pingyu Jiang, and Mei Zheng. 2017. Environmental and economic sustainability-aware resource service scheduling for industrial product service systems. *Journal of Intelligent Manufacturing* 28, 6 (2017), 1303–1316.

[44] Daniel Gmach, Yuan Chen, Amip Shah, Jerry Rolia, Cullen Bash, Tom Christian, and Ratnesh Sharma. 2010. Profiling sustainability of datacenters. In *Proceedings of the IEEE International Symposium on Sustainable Systems and Technology (ISSST'10)*. 1–6.

[45] Barbara Kitchenham, O. Pearl Brereton, David Budgen, Mark Turner, John Bailey, and Stephen Linkman. 2009. Systematic literature reviews in software engineering–a systematic literature review. *Information and Software Technology* 51, 1 (2009), 7–15.

[46] A. Hameed, A. Khoshkbarforoushha, R. Ranjan, P. P. Jayaraman, J. Kolodziej, P. Balaji, S. Zeadally, Q. M. Malluhi, N. Tziritas, A. Vishnu, and S. U. Khan. 2016. A survey and taxonomy on energy efficient resource allocation techniques for cloud computing systems. *Computing* 98, 7 (2016), 751–774.

[47] R. Basmadjian, P. Bouvry, G. D. Costa, L. Gyarmati, D. Kliazovich, S. Lafond, L. Lefèvre, H. D. Meer, J.-M. Pierson, R. Pries, J. Torres, T. A. Trinh, and S. U. Khan. 2015. Green data centers. In *Large-Scale Distributed Systems and Energy Efficiency: A Holistic View*, J.-M. Pierson (Ed.). John Wiley & Sons, Inc, Hoboken, NJ.

[48] Keke Gai, Meikang Qiu, Hui Zhao, and Xiaotong Sun. 2018. Resource management in sustainable cyber-physical systems using heterogeneous cloud computing. *IEEE Transactions on Sustainable Computing* 3, 2 (2018), 60–72.

[49] Cullen Bash, Tahir Cader, Yuan Chen, Daniel Gmach, Richard Kaufman, Dejan Milojicic, Amip Shah, and Puneet Sharma. 2011. Cloud sustainability dashboard, dynamically assessing sustainability of datacenters and clouds. In *Proceedings of the 5th Open Cirrus Summit*. Hewlett Packard, CA. 13.

[50] Tobias Van Damme, Claudio De Persis, and Pietro Tesi. 2018. Optimized thermal-aware job scheduling and control of data centers. *IEEE Transactions on Control Systems Technology* (2018). DOI : https://doi.org/10.1109/TCST.2017.2783366

[51] Dan Azevedo, M. Patterson, J. Pouchet, and R. Tipley. 2010. Carbon usage effectiveness (CUE): a green grid datacenter sustainability metric. In *The Green Grid*. Online Available at http://airatwork.com/wp-content/uploads/The-Green-Grid-White-Paper-32-CUE-Usage-Guidelines.pdf.

[52] Dan Azevedo, Symantec Christian Belady, and J. Pouchet. 2011. Water usage effectiveness (WUE^{TM}): A green grid datacenter sustainability metric. In *The Green Grid*. Online Available at http://tmp2014.airatwork.com/wp-content/uploads/The-Green-Grid-White-Paper-35-WUE-Usage-Guidelines.pdf.

[53] Mark A. Oxley, Eric Jonardi, Sudeep Pasricha, Anthony A. Maciejewski, Howard Jay Siegel, Patrick J. Burns, and Gregory A. Koenig. 2018. Rate-based thermal, power, and co-location aware resource management for heterogeneous data centers. *Journal of Parallel and Distributed Computing* 112 (2018), 126–139.

[54] Saurabh Kumar Garg, Chee Shin Yeo, Arun Anandasivam, and Rajkumar Buyya. 2011. Environment-conscious scheduling of HPC applications on distributed cloud-oriented datacenters. *Journal of Parallel and Distributed Computing* 71, 6 (2011), 732–749.

[55] Mung Chiang, Sangtae Ha, I. Chih-Lin, Fulvio Risso, and Tao Zhang. 2017. Clarifying fog computing and networking: 10 questions and answers. *IEEE Communications Magazine* 55, 4 (2017), 18–20.

[56] Sukhpal Singh Gill, Inderveer Chana, and Rajkumar Buyya. 2017. IoT-based agriculture as a cloud and big data service: The beginning of digital india. *Journal of Organizational and End User Computing (JOEUC)* 29, 4 (2017), 1–23.

[57] Anne-Cecile Orgerie, Marcos Dias de Assuncao, and Laurent Lefevre. 2014. A survey on techniques for improving the energy efficiency of large-scale distributed systems. *ACM Computing Surveys* 46, 4 (2014), 1–31.

[58] Monica Vitali and Barbara Pernici. 2014. A survey on energy efficiency in information systems. *International Journal of Cooperative Information Systems* 23, 3 (2014), 1–38.

[59] Praveen Kumar Gupta, B. T. Maharaj, and Reza Malekian. 2017. A novel and secure IoT based cloud centric architecture to perform predictive analysis of users activities in sustainable health centres. *Multimedia Tools and Applications* 76, 18 (2017), 18489–18512.

[60] Sukhpal Singh Gill, Rajkumar Buyya, Inderveer Chana, Maninder Singh, and Ajith Abraham. 2018. BULLET: Particle swarm optimization based scheduling technique for provisioned cloud resources. *Journal of Network and Systems Management* 26, 2 (2018), 361–400.

[61] W. O. Brown Nils, Tove Malmqvist, Wei Bai, and Marco Molinari. 2013. Sustainability assessment of renovation packages for increased energy efficiency for multi-family buildings in Sweden. *Building and Environment* 61 (2013), 140–148.

[62] Chia-Yu Hsu, Chin-Sheng Yang, Liang-Chih Yu, Chi-Fang Lin, Hsiu-Hsen Yao, Duan-Yu Chen, K. Robert Lai, and Pei-Chann Chang. 2015. Development of a cloud-based service framework for energy conservation in a sustainable intelligent transportation system. *International Journal of Production Economics* 164 (2015), 454–461.

[63] Christos N. Markides, 2013. The role of pumped and waste heat technologies in a high-efficiency sustainable energy future for the UK. *Applied Thermal Engineering* 53, 2 (2013), 197–209.

[64] Mueen Uddin and Azizah Abdul Rahman. 2012. Energy efficiency and low carbon enabler green IT framework for datacenters considering green metrics. *Renewable and Sustainable Energy Reviews* 16, 6 (2012), 4078–4094.

[65] Maurizio Giacobbe, Antonio Celesti, Maria Fazio, Massimo Villari, and Antonio Puliafito. 2015. Towards energy management in cloud federation: a survey in the perspective of future sustainable and cost-saving strategies. *Computer Networks* 91 (2015), 438–452.

[66] Anna Kramers, Mattias Höjer, Nina Lövehagen, and Josefin Wangel. 2014. Smart sustainable cities–Exploring ICT solutions for reduced energy use in cities. *Environmental Modelling & Software* 56 (2014), 52–62.

[67] Sukhpal Singh Gill, Inderveer Chana, Maninder Singh, and Rajkumar Buyya. 2017. CHOPPER: an intelligent QoS-aware autonomic resource management approach for cloud computing. *Cluster Computing* (2017), 1–39. DOI: https://doi.org/10.1007/s10586-017-1040-z

[68] Felix Wolf, Bernd Mohr, and Dieter an Mey, eds. 2013. *Proceedings of the 19th International Conference on Parallel Processing (Euro-Par'13)*. Vol. 8097. Springer, Aachen, Germany.

[69] Zhao Chen, Ziru Chen, Lin X. Cai, and Yu Cheng. 2017. Energy-throughput tradeoff in sustainable cloud-ran with energy harvesting. arXiv preprint arXiv:1705.02968 (2017).

[70] Ashkan Gholamhosseinian and Ahmad Khalifeh. 2012. *Cloud Computing and Sustainability: Energy Efficiency Aspects*. PhD Dissertation. Halmstad University, Halmstad, Sweden.

[71] Junaid Shuja, Kashif Bilal, Sajjad A. Madani, Mazliza Othman, Rajiv Ranjan, Pavan Balaji, and Samee U. Khan. 2016. Survey of techniques and architectures for designing energy-efficient datacenters. *IEEE Systems Journal* 10, 2 (2016), 507–519.

[72] Chi Xu, Ziyang Zhao, Haiyang Wang, Ryan Shea, and Jiangchuan Liu. 2017. Energy efficiency of cloud virtual machines: From traffic pattern and CPU affinity perspectives. *IEEE Systems Journal* 11, 2 (2017), 835–845.

[73] Maurizio Giacobbe, Antonio Celesti, Maria Fazio, Massimo Villari, and Antonio Puliafito. 2015. A sustainable energy-aware resource management strategy for IoT Cloud federation. In *Proceedings of the IEEE International Symposium on Systems Engineering*. 170–175.

[74] Thomas Dandres, Rejean Samson, Reza Farrahi Moghaddam, Kim Khoa Nguyen, Mohamed Cheriet, and Yves Lemieux. 2016. The green sustainable telco cloud: Minimizing greenhouse gas emissions of server load migrations between distributed datacenters. In *Proceedings of the 12th IEEE International Conference Network and Service Management (CNSM'16)*. 383–387.

[75] Minxian Xu, Amir Vahid Dastjerdi, and Rajkumar Buyya. 2016. Energy efficient scheduling of cloud application components with brownout. *IEEE Transactions on Sustainable Computing* 1, 2 (2016), 40–53.

[76] Tian Wang, Yang Li, Guojun Wang, Jiannong Cao, Md Zakirul Alam Bhuiyan, and Weijia Jia. 2017. Sustainable and efficient data collection from WSNs to cloud. *IEEE Transactions on Sustainable Computing* (2017). DOI: https://doi.org/10.1109/TSUSC.2017.2690301

[77] Sukhpal Singh and Inderveer Chana. 2014. Energy based efficient resource scheduling: a step towards green computing. *International Journal of Energy, Information and Communications* 5, 2 (2014), 35–52.

[78] Jianting Fu, Zhen Zhang, and Dan Lyu. 2018. Research and application of information service platform for agricultural economic cooperation organization based on Hadoop cloud computing platform environment: taking agricultural and fresh products as an example. *Cluster Computing* (2018), 1–12. DOI: https://doi.org/10.1007/s10586-018-2380-z

[79] J. Park and Y. K. Cho. 2018. Use of a mobile BIM application integrated with asset tracking technology over a cloud. In *Proceedings of the 21st International Symposium on Advancement of Construction Management and Real Estate*. 1535–1545.

[80] Saurabh Kumar Garg and Rajkumar Buyya. 2012. Green cloud computing and environmental sustainability. In *Harnessing Green IT: Principles and Practices*, San Murugesan and G. R. Gangadharan (Eds.). Wiley, UK, 315–340.

[81] Sareh Fotuhi Piraghaj, Amir Vahid Dastjerdi, Rodrigo N. Calheiros, and Rajkumar Buyya. 2017. A survey and taxonomy of energy efficient resource management techniques in platform as a service cloud. In *Handbook of Research on End-to-End Cloud Computing Architecture Design*, Jianwen "Wendy" Chen, Yan Zhang, and Ron Gottschalk (Eds.). IGI Global, USA, 410–454.

[82] Abbas Mardani, Ahmad Jusoh, Edmundas Kazimieras Zavadskas, Fausto Cavallaro, and Zainab Khalifah. 2015. Sustainable and renewable energy: An overview of the application of multiple criteria decision-making techniques and approaches. *Sustainability* 7, 10 (2015), 13947–13984.

[83] Sukhpal Singh and Inderveer Chana. 2016. EARTH: Energy-aware autonomic resource scheduling in cloud computing. *Journal of Intelligent & Fuzzy Systems* 30, 3 (2016), 1581–1600.
[84] Mark A. Oxley, Eric Jonardi, Sudeep Pasricha, Anthony A. Maciejewski, Howard Jay Siegel, Patrick J. Burns, and Gregory A. Koenig. 2017. Rate-based thermal, power, and co-location aware resource management for heterogeneous datacenters. *Journal of Parallel and Distributed Computing* 112, 2 (2017), 126–139.
[85] Leandro Cupertino, Georges Da Costa, Ariel Oleksiak, Wojciech Pia, Jean-Marc Pierson, Jaume Salom, Laura Siso, Patricia Stolf, Hongyang Sun, and Thomas Zilio. 2015. Energy-efficient, thermal-aware modeling and simulation of datacenters: the CoolEmAll approach and evaluation results. *Ad Hoc Networks* 25 (2015), 535–553.
[86] Hongyang Sun, Patricia Stolf, Jean-Marc Pierson, and Georges Da Costa. 2014. Energy-efficient and thermal-aware resource management for heterogeneous datacenters. *Sustainable Computing: Informatics and Systems* 4, 4 (2014), 292–306.
[87] Jordi Guitart. 2017. Toward sustainable datacenters: a comprehensive energy management strategy. *Computing* 99, 6 (2017), 597–615.
[88] Xiaoying Wang, Guojing Zhang, Mengqin Yang, and Lei Zhang. 2017. Green-aware virtual machine migration strategy in sustainable cloud computing environments. In *Cloud Computing-Architecture and Applications*, Jaydip Sen (Ed.). InTech, London, UK.
[89] Yuanxiong Guo, Yanmin Gong, Yuguang Fang, Pramod P. Khargonekar, and Xiaojun Geng. 2014. Energy and network aware workload management for sustainable datacenters with thermal storage. *IEEE Transactions on Parallel and Distributed Systems* 25, 8 (2014), 2030–2042.
[90] Hassan Shamalizadeh, Luis Almeida, Shuai Wan, Paulo Amaral, Senbo Fu, and Shashi Prabh. 2013. Optimized thermal-aware workload distribution considering allocation constraints in datacenters. In *Proceedings of the IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing*. 208–214.
[91] Dong Han and Tao Shu. 2015. Thermal-aware energy-efficient task scheduling for DVFS-enabled datacenters. In *Proceedings of the IEEE International Conference on Computing, Networking and Communications (ICNC)*. 536–540.
[92] Lijun Fu, Jianxiong Wan, Ting Liu, Xiang Gui, and Ran Zhang. 2017. A temperature-aware resource management algorithm for holistic energy minimization in datacenters. In *Proceedings of the IEEE Workshop on Recent Trends in Telecommunications Research (RTTR'17)*. 1–5.
[93] Hui Dou, Yong Qi, Wei Wei, and Houbing Song. 2017. Carbon-aware electricity cost minimization for sustainable datacenters. *IEEE Transactions on Sustainable Computing* 2, 2 (2017), 211–223.
[94] Sukhpal Singh, Inderveer Chana, Maninder Singh, and Rajkumar Buyya. 2016. SOCCER: Self-optimization of energy-efficient cloud resources. *Cluster Computing* 19, 4 (2016), 1787–1800.
[95] Corentin Dupont. 2016. *Energy Adaptive Infrastructure for Sustainable CDCs*. PhD Dissertation. University of Trento, Trento, Italy.
[96] Patricia Arroba Garcia. 2017. *Proactive Power and Thermal Aware Optimizations for Energy-Efficient Cloud Computing*, Ph.D. Dissertation. Universidad Politecnica de Madrid, Spain.
[97] Marina Zapater, Patricia Arroba, José Luis Ayala Rodrigo, Katzalin Olcoz Herrero, and José Manuel Moya Fernandez. 2015. Energy-aware policies in ubiquitous computing facilities. In *Cloud Computing with e-Science Applications*, Olivier Terzo and Lorenzo Mossucca (Eds.). CRC Press, USA, 267–284.
[98] Ting-Hsuan Chien and Rong-Guey Chang. 2016. A thermal-aware scheduling for multicore architectures. *Journal of Systems Architecture* 62 (2016), 54–62.
[99] Xiaoying Wang, Zhihui Du, Yinong Chen, and Mengqin Yang. 2015. A green-aware virtual machine migration strategy for sustainable datacenter powered by renewable energy. *Simulation Modelling Practice and Theory* 58 (2015), 3–14.
[100] Ranjit Bose and Xin Luo. 2011. Integrative framework for assessing firms' potential to undertake Green IT initiatives via virtualization–A theoretical perspective. *The Journal of Strategic Information Systems* 20, 1 (2011), 38–54.
[101] Mehiar Dabbagh, Bechir Hamdaoui, Mohsen Guizani, and Ammar Rayes. 2016. An energy-efficient VM prediction and migration framework for overcommitted clouds. *IEEE Transactions on Cloud Computing* (2016). DOI : https://doi.org/10.1109/TCC.2016.2564403
[102] Maurizio Giacobbe, Antonio Celesti, Maria Fazio, Massimo Villari, and Antonio Puliafito. 2015. An approach to reduce carbon dioxide emissions through virtual machine migrations in a sustainable cloud federation. In *Sustainable Internet and ICT for Sustainability (SustainIT'15)*. IEEE. 1–4.
[103] R. Bolla, R. Bruschi, F. Davoli, C. Lombardo, J. F. Pajo, and O. R. Sanchez. 2017. The dark side of network functions virtualization: A perspective on the technological sustainability. In *Proceedings of the IEEE International Conference on Communications (ICC'17)*. 1–7.
[104] Luftus Sayeed and Sam Gill. 2008. An exploratory study on environmental sustainability and IT use. *Proceedings of AMCIS'08*. 55.

[105] Kateryna Rybina, Abhinandan Patni, and Alexander Schill. 2014. Analysing the migration time of live migration of multiple virtual machines. In *Proceedings of the 4th International Conference on Cloud Computing and Services Science (CLOSER'14)*. 590–597.

[106] Atefeh Khosravi, Adel Nadjaran Toosi, and Rajkumar Buyya. 2017. Online virtual machine migration for renewable energy usage maximization in geographically distributed cloud datacenters. *Concurrency and Computation: Practice and Experience* 29, 18 (2017), 1–13.

[107] Grace Metzger, Alison Stevens, Megan Harmon, and Jeffrey Merhout. 2012. Sustainability opportunities for universities: Cloud computing, virtualization and other recommendations. In *Proceedings of the Eighteenth Americas Conference on Information Systems (AMCIS'12)*.

[108] Sukhpal Singh, Inderveer Chana, and Maninder Singh. 2017. The journey of QoS-Aware autonomic cloud computing. *IT Professional* 19, 2 (2017), 42–49.

[109] Rahul Ghosh, Francesco Longo, Ruofan Xia, Vijay K. Naik, and Kishor S. Trivedi. 2014. Stochastic model driven capacity planning for an infrastructure-as-a-service cloud. *IEEE Transactions on Services Computing* 7, 4 (2014), 667–680.

[110] Yousri Kouki and Thomas Ledoux. 2012. SLA-driven capacity planning for cloud applications. In *Proceedings of the IEEE 4th International Conference on Cloud Computing Technology and Science (CloudCom'12)*. 135–140.

[111] Yexi Jiang, Chang-Shing Perng, Tao Li, and Rong N. Chang. 2013. Cloud analytics for capacity planning and instant VM provisioning. *IEEE Transactions on Network and Service Management* 10, 3 (2013), 312–325.

[112] Erica Sousa, Fernando Lins, Eduardo Tavares, Paulo Cunha, and Paulo Maciel. 2015. A modeling approach for cloud infrastructure planning considering dependability and cost requirements. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 45, 4 (2015), 549–558.

[113] Fanxin Kong and Xue Liu. 2016. Greenplanning: 2016. Optimal energy source selection and capacity planning for green datacenters. In *Proceedings of the ACM/IEEE 7th International Conference on Cyber-Physical Systems (ICCPS'16)*. 1–10.

[114] Marcus Carvalho, Daniel A. Menascé, and Francisco Brasileiro. 2017. Capacity planning for IaaS cloud providers offering multiple service classes. *Future Generation Computer Systems* 77 (2017), 97–111.

[115] Daniel A. Menascé and Paul Ngo. 2009. Understanding Cloud Computing: Experimentation and Capacity Planning. In *Proceedings of the International Computer Measurement Group Conference*. 1–11.

[116] Christoph Dorsch and Björn Häckel. 2012. Matching economic efficiency and environmental sustainability: The potential of exchanging excess capacity in cloud service environments. In *Proceedings of the 33rd International Conference on Information Systems (ICIS'12)*. 1–18.

[117] Syed Shabbar Raza, Isam Janajreh, and Chaouki Ghenai. 2014. Sustainability index approach as a selection criteria for energy storage system of an intermittent renewable energy source. *Applied Energy* 136 (2014), 909–920.

[118] F. Pierie, J. Bekkering, R. M. J. Benders, WJ Th van Gemert, and H. C. Moll. 2016. A new approach for measuring the environmental sustainability of renewable energy production systems: Focused on the modelling of green gas production pathways. *Applied Energy* 162 (2016), 131–138.

[119] Adel Nadjaran Toosi, Chenhao Qu, Marcos Dias de Assunção, and Rajkumar Buyya. 2017. Renewable-aware geographical load balancing of web applications for sustainable datacenters. *Journal of Network and Computer Applications* 83 (2017), 155–168.

[120] J. O. Petinrin, and Mohamed Shaaban. 2015. Renewable energy for continuous energy sustainability in Malaysia. *Renewable and Sustainable Energy Reviews* 50 (2015), 967–981.

[121] Eric W. Stein. 2013. A comprehensive multi-criteria model to rank electric energy production technologies. *Renewable and Sustainable Energy Reviews* 22 (2013), 640–654.

[122] Anders S. G. Andrae and Tomas Edler. 2015. On global electricity usage of communication technology: Trends to 2030. *Challenges* 6, 1 (2015), 117–157.

[123] Gang Liu, Ali M. Baniyounes, M. G. Rasul, M. T. O. Amanullah, and Mohammad Masud Kamal Khan. 2013. General sustainability indicator of renewable energy system based on grey relational analysis. *International Journal of Energy Research* 37, 14 (2013), 1928–1936.

[124] Zhenhua Liu, Yuan Chen, Cullen Bash, Adam Wierman, Daniel Gmach, Zhikui Wang, Manish Marwah, and Chris Hyser. 2012. Renewable and cooling aware workload management for sustainable datacenters. *ACM SIGMETRICS Performance Evaluation Review* 40, 1 (2012), 175–186.

[125] Sukhpal Singh, Inderveer Chana, and Rajkumar Buyya. 2017. STAR: SLA-aware autonomic management of cloud resources. *IEEE Transactions on Cloud Computing* (2017). DOI: https://doi.org/10.1109/TCC.2017.2648788

[126] Abbas Mardani, Ahmad Jusoh, Edmundas Kazimieras Zavadskas, Fausto Cavallaro, and Zainab Khalifah. 2015. Sustainable and renewable energy: An overview of the application of multiple criteria decision making techniques and approaches. *Sustainability* 7, 10 (2015), 13947–13984.

[127] Xiaomin Xu, Dongxiao Niu, Jinpeng Qiu, Meiqiong Wu, Peng Wang, Wangyue Qian, and Xiang Jin. 2016. Comprehensive evaluation of coordination development for regional power grid and renewable energy power

supply based on improved matter element extension and TOPSIS method for sustainability. *Sustainability* 8, 2 (2016), 143.

[128] Song Hwa Chae, Sang Hun Kim, Sung-Geun Yoon, and Sunwon Park. 2010. Optimization of a waste heat utilization network in an eco-industrial park. *Applied Energy* 87, 6 (2010), 1978–1988.

[129] Kalyan K. Srinivasan, Pedro J. Mago, and Sundar R. Krishnan. 2010. Analysis of exhaust waste heat recovery from a dual fuel low temperature combustion engine using an Organic Rankine Cycle. *Energy* 35, 6 (2010), 2387–2399.

[130] Sotirios Karellas and Konstantinos Braimakis. 2016. Energy–exergy analysis and economic investigation of a cogeneration and trigeneration ORC–VCC hybrid system utilizing biomass fuel and solar power. *Energy Conversion and Management* 107 (2016), 103–113.

[131] James Freeman, Ilaria Guarracino, Soteris A. Kalogirou, and Christos N. Markides. 2017. A small-scale solar organic Rankine cycle combined heat and power system with integrated thermal-energy storage. *Applied Thermal Engineering* 117 (2017), 1543–1554.

[132] Yong Du, Kefeng Cai, Song Chen, Hongxia Wang, Shirley Z. Shen, Richard Donelson, and Tong Lin. 2015. Thermoelectric fabrics: Toward power generating clothing. *Scientific Reports* 5 (2015), 1–6.

[133] Martin Helm, Kilian Hagel, Werner Pfeffer, Stefan Hiebler, and Christian Schweigler. 2014. Solar heating and cooling system with absorption chiller and latent heat storage–a research project summary. *Energy Procedia* 48 (2014), 837–849.

[134] L. M. Ayompe and Aidan Duffy. 2013. Thermal performance analysis of a solar water heating system with heat pipe evacuated tube collector using data from a field trial. *Solar Energy* 90 (2013), 17–28.

[135] Anton Beloglazov, Rajkumar Buyya, Young Choon Lee, and Albert Zomaya. A taxonomy and survey of energy-efficient datacenters and cloud computing systems. *Advances in Computers* 82, 2 (2011), 47–111.

[136] Fahimeh Alizadeh Moghaddam, Patricia Lago, and Paola Grosso. 2015. Energy-efficient networking solutions in cloud-based environments: A systematic literature review. *ACM Computing Surveys (CSUR)* 47, 4, 1–32.

[137] Mehiar Dabbagh, Bechir Hamdaoui, Ammar Rayes, and Mohsen Guizani. 2017. Shaving datacenter power demand peaks through energy storage and workload shifting control. *IEEE Transactions on Cloud Computing* (2017). DOI: https://doi.org/10.1109/TCC.2017.2744623

[138] Fredy Juarez, Jorge Ejarque, and Rosa M. Badia. 2018. Dynamic energy-aware scheduling for parallel task-based application in cloud computing. *Future Generation Computer Systems* 78 (2018), 257–271.

[139] Anik Mukherjee, R. P. Sundarraj, and Kaushik Dutta. 2017. Users' time preference based stochastic resource allocation in cloud spot market: cloud provider's perspective. In *Proceedings of the 12th International Conference on Design Science Research in Information Systems and Technology (DESRIST'17)*. 30 May-1 Jun. Karlsruher Institut für Technologie (KIT), Karlsruhe, Germany

[140] Suleiman Onimisi Aliyu, Feng Chen, Ying He, and Hongji Yang. 2017. A Game-theoretic based QoS-Aware capacity management for real-time edgeiot applications. In *Proceedings of the IEEE International Conference on Software Quality, Reliability and Security (QRS)*. 386–397.

[141] D. Kanapram, R. Rapuzzi, G. Lamanna, and M. Repetto. 2017. A framework to correlate power consumption and resource usage in cloud infrastructures. In *Proceedings of the IEEE International Conference on Network Softwarization (NetSoft'17)*. 1–5.

[142] Chao Jin, Bronis R. de Supinski, David Abramson, Heidi Poxon, Luiz DeRose, Minh Ngoc Dinh, Mark Endrei, and Elizabeth R. Jessup. 2016. A survey on software methods to improve the energy efficiency of parallel computing. *The International Journal of High Performance Computing Applications* 31, 6 (2016), 517–549.

[143] Sukhpal Singh, Inderveer Chana, 2013. Consistency verification and quality assurance (CVQA) traceability framework for SaaS. In *Proceedings of the 3rd IEEE International Advance Computing Conference (IACC'13)*. India.

[144] Junaid Shuja, Kashif Bilal, Sajjad Ahmad Madani, and Samee U. Khan. 2014. Data center energy efficient resource scheduling. *Cluster Computing* 17, 4 (2014), 1265–1277.

[145] Junaid Shuja, Raja Wasim Ahmad, Abdullah Gani, Abdelmuttlib Ibrahim Abdalla Ahmed, Aisha Siddiqa, Kashif Nisar, Samee U. Khan, and Albert Y. Zomaya. 2017. Greening emerging IT technologies: techniques and practices. *Journal of Internet Services and Applications* 8, 1, 1–11.

[146] Ignacio Aransay, Marina Zapater, Patricia Arroba, and José M. Moya. 2015. A trust and reputation system for energy optimization in cloud data centers. In *Proceedings of the IEEE 8th International Conference on Cloud Computing (CLOUD'15)*. 138–145.

[147] Eduard Oró, Ricard Allepuz, Ingrid Martorell, and Jaume Salom. 2018. Design and economic analysis of liquid cooled data centres for waste heat recovery: A case study for an indoor swimming pool. *Sustainable Cities and Society* 36 (2018), 185–203.

[148] Atefeh Khosravi and Rajkumar Buyya. 2018. Short-term prediction model to maximize renewable energy usage in cloud data centers. In *Sustainable Cloud and Energy Services*. Springer, Cham, 203–218.

[149] Charalampos P. Triantafyllidis, Rembrandt H. E. M. Koppelaar, Xiaonan Wang, Koen H. van Dam, and Nilay Shah. 2018. An integrated optimization platform for sustainable resource and infrastructure planning. *Environmental Modelling & Software* 101 (2018), 146–168.

[150] Theodore A. Ndukaife and A. G. Agwu Nnanna. 2018. Optimization of water consumption in hybrid evaporative cooling air conditioning systems for data center cooling applications. *Heat Transfer Engineering*, 1–15. DOI: https://doi.org/10.1080/01457632.2018.1436418

[151] Jiahong Wu, Yuan Jin, and Jianguo Yao. 2018. EC 3: Cutting cooling energy consumption through weather-aware geo-scheduling across multiple datacenters. *IEEE Access* 6 (2018), 2028–2038.

[152] Sudipta Sahana, Rajesh Bose, and Debabrata Sarddar. 2018. Server utilization-based smart temperature monitoring system for cloud data center. In *Industry Interactive Innovations in Science, Engineering and Technology*, S. Bhattacharyya, S. Sen, M. Dutta, P. Biswas, and H. Chattopadhyay (Eds.). Springer, Singapore, 309–319.

[153] Morito Matsuoka, Kazuhiro Matsuda, and Hideo Kubo. 2017. Liquid immersion cooling technology with natural convection in data center. In *Proceedings of the IEEE 6th International Conference on Cloud Networking (CloudNet'17)*. 1–7.

[154] Qiang Liu, Yujun Ma, Musaed Alhussein, Yin Zhang, and Limei Peng. 2016. Green data center with IoT sensing and cloud-assisted smart temperature control system. *Computer Networks* 101 (2016), 104–112.

[155] Ioannis Manousakis, Íñigo Goiri, Sriram Sankar, Thu D. Nguyen, and Ricardo Bianchini. 2015. Coolprovision: Underprovisioning datacenter cooling. In *Proceedings of the 6th ACM Symposium on Cloud Computing*. ACM, 356–367.

[156] Sukhpal Singh Gill and Rajkumar Buyya. 2018. Resource provisioning based scheduling framework for execution of heterogeneous and clustered workloads in clouds: From fundamental to autonomic offering. *Journal of Grid Computing* (2018), 1–33. DOI: https://doi.org/10.1007/s10723-017-9424-0

[157] Sathya Chinnathambi, Agilan Santhanam, Jeyarani Rajarathinam, and M. Senthilkumar. 2018. Scheduling and checkpointing optimization algorithm for Byzantine fault tolerance in cloud clusters. *Cluster Computing* (2018), 1–14.

[158] Stelios Sotiriadis, Nik Bessis, and Rajkumar Buyya. 2018. Self-managed virtual machine scheduling in Cloud systems. *Information Sciences* 433–434 (2018), 381–400.

[159] Milad Ranjbari and Javad Akbari Torkestani. 2018. A learning automata-based algorithm for energy and SLA efficient consolidation of virtual machines in cloud data centers. *Journal of Parallel and Distributed Computing* 113 (2018), 55–62.

[160] Adnan Ashraf and Ivan Porres. 2018. Multi-objective dynamic virtual machine consolidation in the cloud using ant colony system. *International Journal of Parallel, Emergent and Distributed Systems* 33, 1 (2018), 103–120.

[161] Naresh Kumar Reddy Beechu, Vasantha Moodabettu Harishchandra, and Nithin Kumar Yernad Balachandra. 2017. High-performance and energy-efficient fault-tolerance core mapping in NoC. *Sustainable Computing: Informatics and Systems* 16 (2017), 1–10.

[162] C. Dastagiraiah, V. Krishna Reddy, and K. V. Pandurangarao. 2018. Dynamic load balancing environment in cloud computing based on VM ware off-loading. In *Data Engineering and Intelligent Computing*, S. C. Satapathy, V. Bhateja, K. S. Raju, and B. Janakiramaiah (Eds.). Springer, Singapore, 483–492.

[163] Yahya Al-Dhuraibi, Fawaz Paraiso, Nabil Djarallah, and Philippe Merle. 2017. Autonomic vertical elasticity of docker containers with elasticdocker. In *Proceedings of the IEEE 10th International Conference on Cloud Computing (CLOUD'17)*. 472–479.

[164] Yahya Al-Dhuraibi, Faiez Zalila, Nabil Djarallah, and Philippe Merle. 2018. Coordinating vertical elasticity of both containers and virtual machines. In *Proceedings of the 8th International Conference on Cloud Computing and Services (CLOSER'18)*. 1–8.

[165] Sukhpal Singh and Inderveer Chana. 2016. A survey on resource scheduling in cloud computing: Issues and challenges. *Journal of Grid Computing* 14, 2 (2016), 217–264.

[166] Eduardo Felipe Zambom Santana, Ana Paula Chaves, Marco Aurelio Gerosa, Fabio Kon, and Dejan S. Milojicic. 2017. Software platforms for smart cities: Concepts, requirements, challenges, and a unified reference architecture. *ACM Computing Surveys* 50, 6 (2017), 78.

Received January 2018; revised May 2018; accepted June 2018

Online Appendix to: A Taxonomy and Future Directions for Sustainable Cloud Computing: 360 Degree View

SUKHPAL SINGH GILL and RAJKUMAR BUYYA, The University of Melbourne, Australia

A BACKGROUND

Figure 16 shows the consumption of energy by different components of an idle server—such as the processor, storage, memory, network, and cooling—as reported in [3, 71, 95].

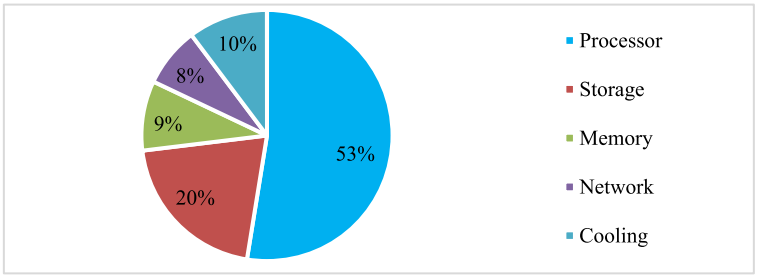


Fig. 16. Energy consumption of the different components of a CDC (Data Source: [3, 71, 95]).

The design of sustainable systems is one of the greatest challenges of the 21st century, the ecological transition coupled with the digital transformation. It is scientifically difficult to decide on the benefit of a technique for improving the sustainability of a system. "Sustainability" by definition involves four areas of study, that is, environmental, social, technological, and economic spheres [7, 15, 82] as shown in Figure 17. There is a need to identify the different components of a sustainable CDC to enable sustainable cloud computing economically (energy/electricity cost), socially (laws and regulations to establish a cloud data center) and technologically (data preservation, protection and retention) and environmentally (energy consumption/carbon footprints/greenhouse emissions).

This systematic review deals mostly with energy in the environmental and economic spheres and by covering issues concerning how to run CDCs efficiently with minimal energy (minimum cost of energy) by harnessing renewable energy–powered resources through holistic management of workloads and resources. The main objectives of sustainable cloud computing are to reduce energy consumption at datacenters and to dispose of hardware devices after their useful life. Cloud computing accelerates our economy rapidly through the use of remote servers via the Internet instead of local servers. Due to the availability of a large number of datacenters, the user's data is stored, managed, and processed efficiently and swiftly but increases carbon footprints, which affects sustainability [135]. Cloud computing is growing very rapidly to fulfill user demand. Initially, there were only a few investment deals and cloud computing accounted for almost $26 million in 2005 [63], as shown in Figure 18. Investment has reached $375 million in 2009 and $40.7 billion in 2010 [43]. In 2013, it reached $150 billion and climbed to $210 billion in 2015 [49]. In 2020,

© 2018 Association for Computing Machinery.
0360-0300/2018/12-ART104 $15.00
https://doi.org/10.1145/3241038

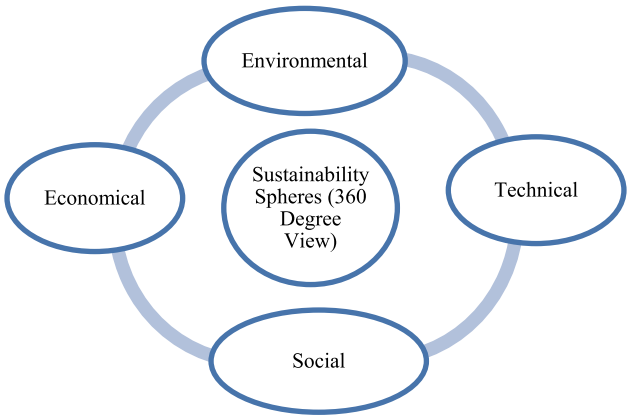


Fig. 17. Types of sustainability spheres.

the investment in cloud computing will be $241 billion and $250 billion in 2022 according to the Verdantix company [136]. Cloud-based solutions are also used by a large number of small and medium-sized corporations, mostly in North America [9, 10, 11]. Further, physical resource (bare metal as service)–based services have been replaced by virtual services, which makes the environment much more sustainable. Cloud computing provides an efficient management of resources, which saves energy by minimizing the number of physical servers. Further, it reduces the maintenance cost and provides more flexibility and scalability for business expansion. The cloud provides the platform to conduct meetings online, which also saves time compared to face-to-face meetings and reduces costs up to 50% [3]. Due to continuous use of virtualization, 31% of the energy consumption has been reduced [4]. The multi-tenant architecture of cloud computing allows for more efficient cloud services. Energy consumption is the still main challenge of cloud computing, which is responsible for producing large carbon footprints and environmental hazards. Researchers are trying to reduce carbon emissions as much as possible; by 2020, servers will avoid 85.7 million metric tons of CO_2 [15].

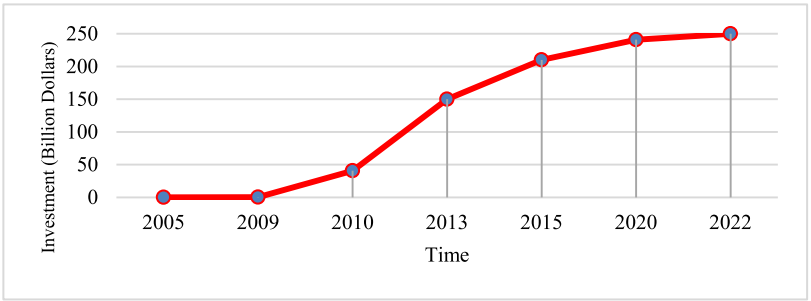


Fig. 18. Investment on cloud computing.

Energy consumption will be increased to 60 billion watts in 2020 [16, 17], which will be equivalent to the energy consumption of 200,000 homes in Tokyo, Japan [12]. Moreover, existing energy-aware techniques mainly focus on reducing the energy consumption of the servers [3, 6, 71]. The other components (networks, storage, memory, processor, and cooling systems) of CDCs are consuming a huge amount of energy. Consequently, there is a need for holistic management of cloud resources to improve the energy efficiency of CDCs. The identification and consideration

of the important aspects of sustainable CDCs (application design, sustainability metrics, capacity planning, energy management, virtualization, thermal-aware scheduling, cooling management, renewable energy, and waste heat utilization) is also required to enable sustainable computing economically (energy/electricity cost) as well as environmentally (energy consumption/carbon footprints/greenhouse emissions).

A.1 Related Surveys and Our Work

A few works have conducted a survey on energy management techniques, but as the research has persistently grown in the field of cloud computing, there is a need for a systematic review to assess, update, and combine the existing literature. Mastelic et al. [2] mainly focus only on energy consumption of datacenters. Shuja et al. [3] present a survey of CDCs only. Chaudhry et al. [14] offer a survey of existing thermal-aware scheduling techniques developed for efficient management of green datacenters. Piraghaj et al. [81] review energy-efficient resource management techniques at the platform level. A broad review of energy-efficient datacenters is presented by Beloglazov et al. [135]. Moghaddam et al. [136] introduce a survey on energy-efficient networking solutions in cloud-based environments. Basmadjian et al. [47] review energy-efficient techniques for cloud datacenters. Orgerie et al. [57] present a survey on improving the energy efficiency of large-scale distributed systems. A broad review of energy efficiency in information systems at the hardware and software levels is offered by Vitali and Pernici [58]. Shuja et al. [71] review the techniques and architectures for designing energy-efficient datacenters and presented an overview of server cooling and storage components. Garg and Buyya [80] present a survey on energy-efficient techniques for cloud datacenters for green cloud computing. A broad review of energy efficiency of the cooling and power supply subsystems is presented by Guitart [87]. Hameed et al. [46] review the energy-efficient resource allocation techniques for cloud computing systems. Shuja et al. [145] present a review of green computing techniques among the emerging IT technologies such as big data and the IoT. Our survey is the first review that covers all of the main characteristics (360-degree view holistically) of sustainable cloud computing. This research augments the previous surveys and presents a fresh systematic review to assess and identify the latest research issues. Table 1 shows the comparison of our survey with other relevant survey articles based on different criteria.

Our Focus: Our systematic review focuses on the types and levels of consumption of energy used by cloud datacenters and identifies the reasons for large amounts of energy consumption and suggests possible solutions to reduce carbon footprints and environmental problems in future. The components of CDCs—such as networks, storage, memory, and cooling systems—are consuming huge

Table 1. Comparison of Our Survey with Other Survey Articles

Criteria		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Reviewed up to		2018	2014	2016	2014	2010	2013	2015	2014	2011	2012	2013	2011	2015	2012	2016
Taxonomy	Application Design	✔(∗)	✔	✔		✔										
	Sustainability Metrics	✔														
	Energy Management	✔(∗)	✔	✔		✔	✔		✔	✔	✔	✔	✔	✔(+)	✔	✔(+)
	Virtualization	✔(∗)		✔	✔	✔		✔								
	Thermal-Aware Scheduling	✔		✔	✔											
	Capacity Planning	✔														
	Cooling Management	✔						✔(+)				✔(+)	✔(+)	✔(+)		
	Renewable Energy	✔(∗)						✔								
	Waste Heat Utilization	✔	✔					✔								

(Continued)

Table 1. Continued

Criteria		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Reviewed up to		2018	2014	2016	2014	2010	2013	2015	2014	2011	2012	2013	2011	2015	2012	2016
Comparison	Based on Evolution and Focus of Study	✔(∗)														
	Based on Taxonomy Mapping	✔								✔					✔	
	Based on Objective	✔			✔	✔	✔								✔(+)	
	Based on Optimization Parameters	✔				✔	✔	✔	✔						✔(+)	
	Based on Demerits (Open Issues)	✔														
Future Research Directions for Every Technique#		✔(∗)														
Emerging Trends and Their Impact		✔(∗)														
Proposed a Conceptual Model for Holistic Management		✔(∗)												✔(+)		

1—Our Survey (This Paper), 2—Mastelic et al. [2], 3—Piraghaj et al. [81], 4—Chaudhry et al. [14], 5—Beloglazov et al. [135], 6—Moghaddam et al. [136], 7—Shuja et al. [3], 8—Basmadjian et al. [47], 9—Orgerie et al. [57], 10—Vitali and Pernici [58], 11—Shuja et al. [71], 12—Garg and Buyya [80], 13—Guitart [87], 14—Hameed et al. [46] and 15—Shuja et al. [145]. *Note:* ∗: Comprehensive Discussion; +: Just an Overview; and #: Summarized Open Challenges.

amounts of energy. To improve energy efficiency of CDCs, there is a need to review energy-aware resource management techniques for management of all resources (including servers, storage, memory, networks, and cooling systems) in a holistic manner and identify the relationship of energy management with other related aspects of sustainable cloud computing, such as application design, sustainability metrics, capacity planning, virtualization, thermal-aware scheduling, cooling management, renewable energy and waste heat utilization. The holistic management of cloud computing resources makes cloud services more energy efficient and sustainable.

B REVIEW METHODOLOGY

This systematic literature review comprises different stages, including formation of review framework, executing the survey, examining the review outcomes, management of review outcomes, and investigation of open issues. The list of research questions used to plan the systematic review is provided in Table 2.

Table 2. Research Questions for Different Categories

Category	Research Questions
Application Design	1 What is the current status of an application design?
	2 What are the QoS requirements of different applications for sustainable cloud computing?
	3 How does one design an application architecture that can reduce coupling among different components of an application?
	4 What is the need for green Information and Communications Technology (ICT)-based innovative applications?
	5 What are the main challenges of an application design for recent technological developments such as the IoT?
	6 What are the different types of application design models for effective energy management?
	7 How does one decrease execution cost and meet deadlines simultaneously?

(Continued)

Table 2. Continued

Category		Research Questions
Sustainability Metrics	1	What are the different sustainability metrics for CDCs?
	2	What is the evolution of sustainability metrics?
	3	How are different sustainability metrics related to each other?
	4	How does one measure the performance of CDCs holistically?
Capacity Planning	1	What are the conditions to change the SLA with respect to time?
	2	What are the criteria for compensation and penalties if a CDC service provider violates the SLA?
	3	What are the different types of components for which capacity planning is required?
	4	How does one recognize and categorize the numerous workloads to design a CDC successfully?
Energy Management	1	How does one reduce energy consumption and its impact on the environment?
	2	What is the difference between static and dynamic energy management techniques?
	3	How does one develop an energy-aware resource management technique that proficiently schedules the provisioned resources without SLA violation?
	4	What are the configurable components for energy management?
	5	What is the trade-off between energy consumption and execution time?
	6	How much energy is consumed by various components (cooling, network, memory, storage, and processor) of the idle server?
Virtualization	1	What is the trade-off between time and energy cost for Virtual Machine (VM) migration?
	2	What are different issues with VM migration in geographically distributed CDCs?
	3	What type of technology is available for VM migration?
	4	What are the optimization criteria for VM migration?
	5	What are the different types of mechanisms for VM elasticity?
	6	What is the difference between resource-aware and performance-aware in VM load-balancing techniques?
	7	What are the co-location criteria for VM consolidation?
	8	What are the different types of recovery techniques for VM fault tolerance?
	9	What are the different types of VM scheduling mechanisms?
Thermal-Aware Scheduling	1	What are the different architectures for thermal-aware scheduling?
	2	What are the different heat modeling techniques?
	3	What different types of thermometers are being used to measure temperature?
	4	What are the different monitoring and awareness techniques?
	5	What simulation tools are used to generate thermal gadgets?
Cooling Management	1	How does one reduce cooling costs without performance degradation?
	2	What are the different types of cooling techniques?
	3	What are the different temperature ranges for different mediums (water and air) in a cooling plant?
	4	What are the different types of heat rejection systems for cooling management?
Renewable Energy	1	What are the main sources of renewable energy?
	2	What are the different objectives of renewable energy-aware techniques?
	3	What are the different storage devices to store renewable energy?
Waste Heat Utilization	1	What are the different techniques for utilization of waste heat?
	2	What are the different types heat-transfer methods?
	3	What are the different types of cooling methods for waste heat utilization?

B.1 Sources of Information

We followed CRD guidelines [45] to perform electronic database search and manual search using different search strings, as mentioned in Table 3, which retrieved 470 research articles. The following electronic databases have been used for searching:

- ACM Digital Library (<www.acm.org/dl>)
- HPC (<www.hpcsage.com>)
- Wiley Interscience (<www.Interscience.wiley.com>)
- Taylor & Francis Online (<www.tandfonline.com >)
- IEEE eXplore (<www.ieeexplore.ieee.org>)
- Google Scholar (<www.scholar.google.co.in>)
- ScienceDirect (<www.sciencedirect.com>)
- Springer (<www.springerlink.com>)

Table 3. Search String

Sr. No.	Keywords	Synonyms	Dates	Content Type
1	Sustainability	Sustainable Cloud Computing, Sustainable Computing	2010–2018	Journal, Conference, Symposium, Workshop, Book Chapter, PhD Thesis, Magazine, White Paper, and Transactions
2	Applications for Sustainable Cloud Computing	QoS, Models, Workloads, Architecture		
3	Energy-aware Sustainable CDC	Static, Dynamic, Resource Management, Consolidation, Power, Configurable Components, Carbon Emission, DVFS, C-States		
4	Capacity Planning for Sustainable CDC	IT Device, Cooling, Power Infrastructure, Autoscaling		
5	Virtualization for Sustainable Computing	VM Migration, VM Technology, VM Elasticity, VM Load Balancing, VM Consolidation, VM Fault Tolerance, VM Scheduling, Checkpoint, Recovery, Failure Masking, Container Management		
8	Metrics for Sustainable Cloud Computing	Sustainability Metrics, CDC Metrics, Sustainability Parameters		
9	Thermal-aware Sustainable Cloud	Architecture, Heat Modeling, Thermometer, Scheduling, Monitoring and Awareness, Simulator		
10	Cooling in Sustainable Cloud Computing	Cooling Plants, Temperature, Heat Rejection System, Location, Mechanical Equipment, Type of Cooling		
11	Renewable Energy and Sustainable Cloud Computing	Workload Scheduling, Energy Sources, Location of Sources, Storage Devices		
12	Waste Heat Utilization for Sustainable Cloud Computing	Heat Model, Heat Transfer Method, Cooling Technique, Location-aware		
13	Holistic Management for Cloud	Energy efficiency, Cloud Resources, Sustainability, Sustainable Cloud Computing		

The review technique used in this systematic review is described in Figure 19.

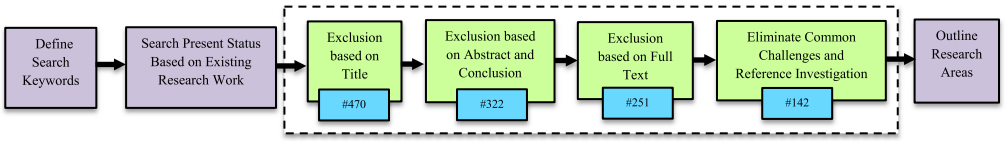


Fig. 19. Review technique used in this systematic review.

B.2 Search Criteria

The keyword "sustainable cloud computing'' is found in the abstract of every research article in every search. Table 3 describes the different search strings used to conduct this detailed survey. Our

article contains both qualitative and quantitative research papers from 2010 to 2018. The literature reported that the basic research started in the area of sustainable cloud computing in 2008. This systematic review contains research work from magazines, white papers (technical reports as well as industry research work), workshops, symposiums, conferences, and journals as described in Appendix D (see Table 26). To cross-check the e-search, an individual search has been applied on some journals of Science Direct, Taylor & Franciş IOS Press, Wiley, ACM, IEEE, and Springer. Figure 19 shows the exclusion criteria used at different stages of this systematic review. Initially, 470 research papers were selected based on their titles, which were reduced to 322 based on their abstracts and conclusion and 251 research papers were retrieved based on their full text. Further, research papers with common challenges (based on exclusion and inclusion criteria) have been eliminated and the references of 251 research papers have been investigated thoroughly to identify a final set of 142 research papers.

B.3 Quality Assessment

To find suitable articles for this systematic review, the criteria of inclusion and exclusion are used to implement the quality assessment on the outstanding research papers. We have investigated several different conferences and journals related to sustainable cloud computing. Further, we used CRD guidelines given by [45] to explore the internal and external validation of results to find high-quality sustainable cloud computing research papers.

B.4 Data Extraction

Table 4 contains the data extraction guidelines, which resulted in 142 research articles in this systematic review.

Table 4. Data Items Extracted from All Papers

Data item	Description
Bibliographic data	Author, year, title, source of research paper
Type of article	Conference, workshop, symposium, journal
Study context	What are the research focus and its aim?
Study plan	Classification of sustainable cloud computing techniques, evolution, taxonomy, comparison based on taxonomy
What is sustainable cloud computing?	It explicitly refers to sustainable cloud computing and its categories.
How was comparison carried out?	Compare various traits, such as objectives, metrics, optimization parameters, and the like.
Data collection	How was the data of sustainable cloud computing collected?
Data analysis	How does one analyze data and extracted research challenges?
Simulation tool	It refers to the tool used for validation.
Research challenges	Open challenges in the area of sustainable cloud computing.

Further, research questions have been designed for different categories of sustainable cloud computing. When the systematic review commenced, we faced a number of difficulties, such as extraction of suitable data. To find out the in-depth knowledge of research work (142 papers), we have contacted various authors. The data extraction procedure used in this systematic review is described below:

- After in-depth review, the first author extracted data from 142 research papers.
- The second author cross-checked the review results using random samples.
- A meeting was called to resolve the conflict during cross-checking.

B.5 Acronyms

A glossary of important acronyms used in this systematic review can be found in Table 5. *Note:* The abbreviations for different techniques are mentioned in their corresponding category.

Table 5. List of Important Acronyms

Acronym	Definition
PDU	Power Distribution Unit
QoS	Quality of Service
SLA	Service Level Agreement
CDC	Cloud Datacenters
IoT	Internet of Things
TWh	Terawatt Hours
CUE	Carbon Usage Efficiency
PUE	Power Usage Efficiency
ICT	Information and Communications Technology
DVFS	Dynamic Voltage and Frequency Scaling
HPC	High-Performance Computing
HTC	High-Throughput Computing
DVS	Dynamic Voltage Scaling
DFS	Dynamic Frequency Scaling
RC	Resistor–Capacitor
CFD	Computational Fluid Dynamics
VM	Virtual Machine
OSs	Operating Systems
LAN	Local Area Network
WAN	Wide Area Network
CPU	Central Processing Unit
IT	Information Technology
HFC	Hydrogen Fuel Cells
CoP	Coefficient of Performance
SaaS	Software as a Service
PaaS	Platform as a Service
IaaS	Infrastructure as a Service
PMU	Power Management Unit
DRAM	Dynamic Random-Access Memory
ATS	Automatic Transfer Switch

C ELEMENTS, EVOLUTION, COMPARISON AND OPEN RESEARCH ISSUES

Figure 20 contains various elements that impact or support sustainable cloud computing (360-Degree View), which have been categorized into nine categories: application design, sustainability metrics, capacity planning, energy management, virtualization, thermal-aware scheduling, cooling management, renewable energy, and waste heat utilization based on the existing literature.

Table 6 contains the mapping of aspects of sustainable CDCs to types of sustainability spheres based on Figures 17 and 20. ***Note:*** The abbreviation of every technique is defined in their corresponding Appendix.

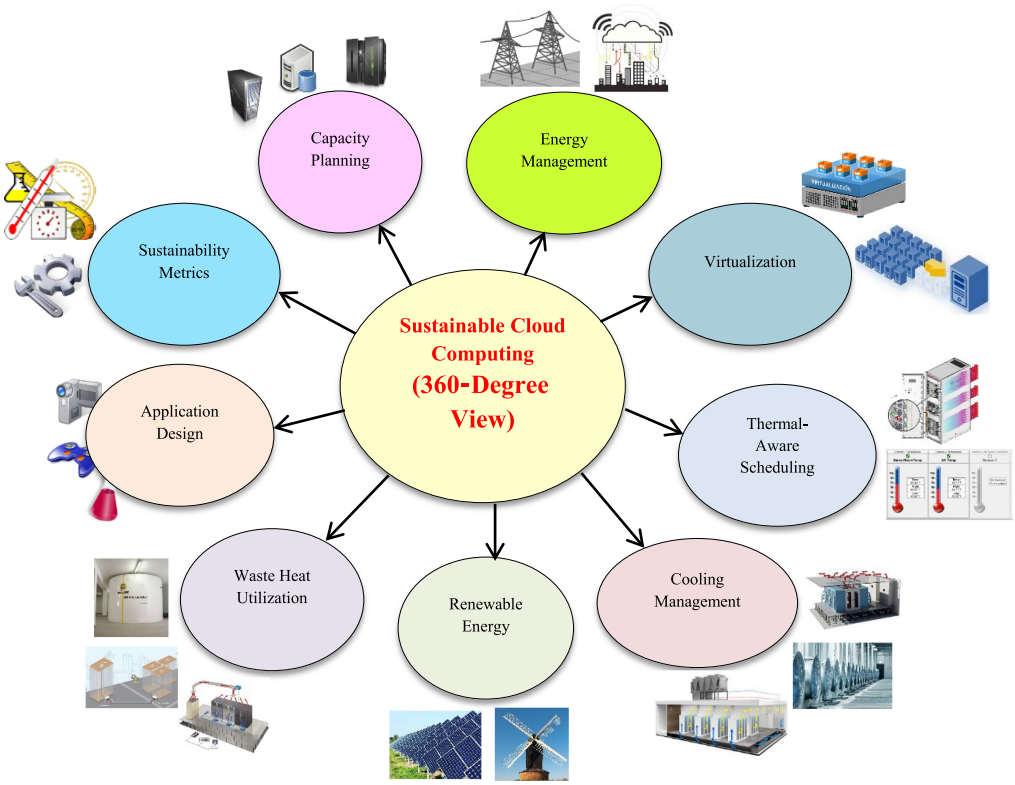


Fig. 20. Taxonomy of sustainable cloud computing (360-Degree view).

Table 6. Mapping of Aspects of Sustainable CDCs to Types of Sustainability Spheres

Aspects of Sustainable CDCs	Types of Sustainability Spheres			
	Economic	Environmental	Social	Technical
Application Design	✔		✔	✔
Sustainability Metrics				✔
Capacity Planning	✔	✔		✔
Energy Management	✔		✔	
Virtualization	✔		✔	
Thermal-Aware Scheduling		✔		
Cooling Management	✔	✔		
Renewable Energy		✔		
Waste Heat Utilization	✔	✔	✔	

C.1 Application Design

Bifulco et al. [20] discuss the concept of ICT-based smart sustainable cities to decrease energy consumption while providing services to households. In this research work, the Bag of Tasks (BoT) application model–driven Component-Based Lifecycle (CBL) is proposed to design an

application that estimates energy consumption for household activities. Further, it is recommended that a smart sustainable city can be designed by optimizing power using the Dynamic Voltage and Frequency Scaling (DVFS) technique, which can save power while devices are idle. Moreover, natural resources can be managed easily by locating datacenters at proper places so that thay have minimum impact on the environment. Cappiello et al. [26] propose a Green Computing–Based Model (GCBM) using the BoT application model, which distributes the user data into interrelated tasks and improves the use of resources and reduces energy consumption and CO_2 emissions during the deployment of applications. Park et al. [42] developed the Cloud-Based Clustering Simulator (CBCS) for desktop resource virtualization to choose a cluster for efficient and sustainable computing. It uses the BoT application model to design an application for selection of an energy-efficient cluster. However, utilization of resources is done based only on network infrastructure and time without considering storage, memory, and CPU.

Bossche et al. [25] designed a Data Mining–Based Architecture (DMBA) application, which uses the Map-Reduce model to extract useful information from unstructured data by eliminating useless data. Further, it helps to form sustainable clusters by reducing the execution time of processing. Fu et al. [78] designed and implemented the Agricultural Information Service (AIS) application to manage data regarding fresh products in a knowledge base and enable communication among different components using the Map-Reduce model. Further, the Hadoop cloud computing platform is used to analyze the application of agriculture for economic cooperation and energy efficiency. Bradley et al. [24] propose an IoT-based Application Design (IoT-AD) using a graph processing model to reduce the cost of data management. Further, a machine-learning method is used to generate sustainable value, which improves environmental sustainability and energy efficiency. NoviFlow [35] designed a Green-Software Defined Network (G-SDN)–based application using a graph processing model to provide sustainable solutions, which further reduces carbon emissions for making network infrastructure efficient. The green-SDN-based network reduces energy consumption of CDCs, which improves sustainability. NoviFlow [35] reports that existing models are mainly focused on cost and QoS to deliver sustainable services.

Pesch et al. [38] propose a task-based Thermal-Aware Scheduling (TAS) architecture, which schedules resources to execute user applications by optimizing energy use to improve sustainability of CDCs. TAS saves energy consumption up to 40%. Juarez et al. [138] propose a Dynamic Energy-Aware Scheduling (DEAS) technique for execution of task-based applications in parallel to estimate energy consumption. To provide energy-efficient and sustainable cloud service, the DEAS technique generates an appropriate energy consumption profile automatically. Further, a trade-off between performance and energy savings is presented. Gill et al. [56] propose an IoT-based Agriculture Service (Agri-Info) model to manage agriculture data in an efficient manner, which is coming from different preconfigured devices. Moreover, Agri-Info uses a task-based application model to design an application and a simulated cloud environment is used to validate the Agri-Info model in terms of energy efficiency and other QoS parameters. Dabbagh et al. [137] have designed the Energy Efficient Technique (EET)–based application using a thread model to reduce monthly expenses of datacenters by delaying the non-urgent workloads, which also decreases the execution of workload. Further, an efficient peak control policy has been designed to predict the demands of datacenters, such as storage, power, and memory, for coming requests and real traces of a Google datacenter are used to validate the EET. Garg et al. [54] proposed the Environment Conscious Application Scheduling (ECAS) framework to design a thread-based application model for execution of High Performance Computing (HPC) applications on cloud resources. Further, performance parameters such as cost, carbon emission rate, and CPU power efficiency have been optimized during execution of applications.

Gmach et al. [44] propose a stream processing model–based Power Profiling Technique (PPT) for datacenters to improve energy use and its effect on the environment; both demand and supply of power should be co-managed to reduce consumption of water and greenhouse gas emissions. Based on the availability of power, user data is processed without compromising the QoS and energy efficiency, which provides sustainability. Park and Cho [79] propose a Mobile Building Information Modeling Platform (MBIMP)–based tracking system using a stream processing–based application model, which processes user data using different streams to improve energy efficiency. Based on a virtual BIM view and communication, the MBIMP system tracks the required real-time information and improves the coordination among different components during data extraction. Charr et al. [29] proposed an Online Frequency Selecting (OFS) algorithm for heterogenous environment, which uses DVFS for message-passing iterative applications to reduce energy consumption.

Figure 21 shows the evolution of application design techniques along with their Focus of Study (FoS) across various years.

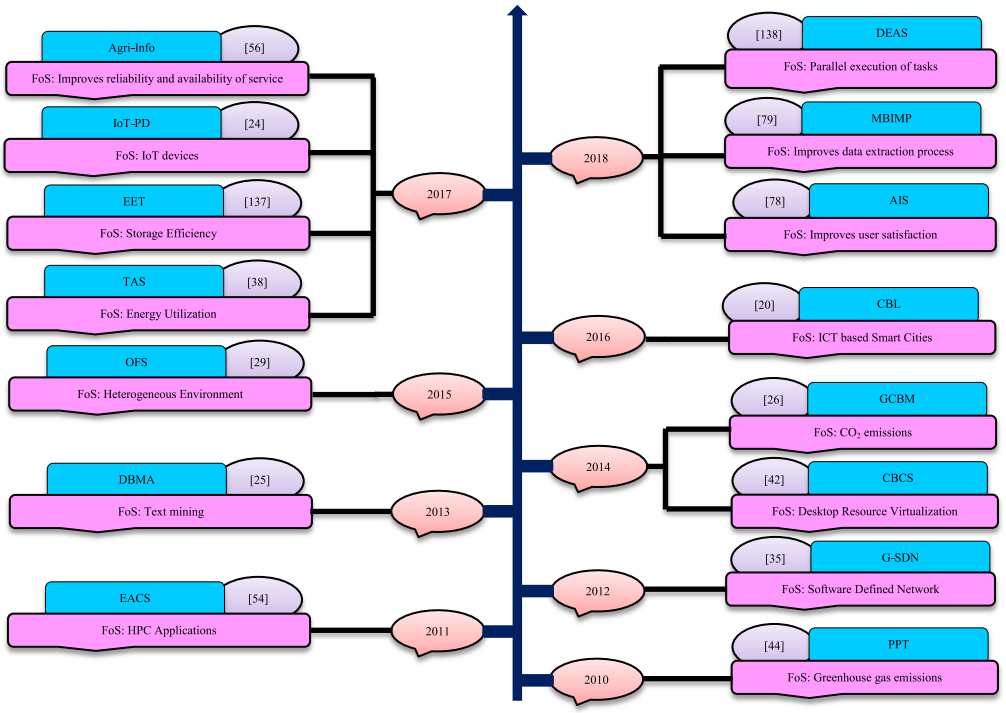


Fig. 21. Evolution of application design techniques.

A summary of these related techniques (application design) and their comparison based on objectives, optimization parameters, and metrics along with open research challenges, is given in Table 7.

Table 7. Comparison of Existing Techniques (Application Design) and Open Research Challenges

Technique	Organization	Objective	Metric	Optimization Parameter	Citations	Open Research Challenges
OFS [29]	University of Franche-Comté, France	Reduce energy consumption	Energy	Energy consumption	7	1. Does not include power consumed by memory and network resources. 2. Secure communication is required.
DEAS [138]	Workflows and Distributed Computing Group, Spain	Improve energy efficiency	Energy efficiency	Energy consumption	10	Trade-off between energy saving and reliability is an open research area.
MBIMP [79]	Georgia Institute of Technology, USA	Improve real-time information tracking	Correctness	Data accuracy	1	High response time and low workload utilization level
AIS [78]	Huazhong Agricultural University, China	Improve throughput	User satisfaction	Memory Usage	-	Security and privacy of cloud service is an open issue.
Agri-Info [56]	The University of Melbourne, Australia	Improve throughput	Throughput, response time, latency, and cost	Execution time, cost, latency, and security	6	Need autonomic management of cloud resources.
CBL [20]	University of Naples, Italy	Reduce energy consumption	Security and energy	Energy consumption	38	Power is required to be managed automatically.
IoT-AD [24]	University of Kentucky, USA	Reduce cost	Cost	Execution cost	3	Secure communication is required.
DMBA [25]	University of Antwerp, Belgium	Reduce execution time	Response time	Execution time	128	Longer response time and no QoS guarantee
GCBM [26]	Polytechnic University of Milan, Italy	Improve throughput	Energy and cost	Energy Use	9	Need to improve reliability of applications
G-SDN [35]	NoviFlow Inc, Canada	Reduce execution time	Energy	Energy use	-	No cost optimization involved
TAS [38]	Cork Institute of Technology, Ireland		Correctness	Data accuracy	1	Not feasible for non-time-sensitive systems
CBCS [42]	Seoul National University of Science and Technology, Korea	Improve throughput	Availability	Availability	12	Difficult to obtain optimal scheduling decisions with dynamic workloads
PPT [44]	HP Lab, USA	Reduce energy consumption	Energy	Energy consumption	43	Suitable for homogeneous CDCs only
EET [137]	Oregon State University, USA	Reduce cost			1	Underutilization of resources
ECAS [54]	The University of Melbourne, Australia	Improve CPU power efficiency	Application transaction rate	Response time	314	Lacks applicability in a virtualized cloud environment

The comparison of existing techniques based on our application design taxonomy is given in Table 8.

Table 8. Comparison of Existing Techniques Based on Taxonomy of Application Design

Technique	Author	Application Model	QoS Parameter	Workload Type	Architecture
OFS	Charr et al. [29]	Message-Passing Interface	Energy	Critical Interactive	Centralized
DEAS	Juarez et al. [138]	Task-Based	Energy	Batch Style	Centralized
MBIMP	Park and Cho [79]	Stream Processing	Cost	Batch Style	Centralized
AIS	Fu et al. [78]	Map-Reduce	Time	Batch Style	Decentralized
Agri-Info	Gill et al. [56]	Task-Based	Execution time, cost, and security	Critical Interactive	Decentralized
IoT-AD	Bradley et al. [24]	Graph Processing	Cost	Critical Interactive	Centralized
EET	Dabbagh et al. [137]	Thread-Based	Cost	Batch Style	Decentralized
TAS	Pesch et al. [38]	Task-Based	Energy	Critical Interactive	Centralized
CBL	Bifulco et al. [20]	Bag-of-Tasks or Parameter Sweep Tasks	Security	Batch Style	Centralized
GCBM	Cappiello et al. [26]	Bag-of-Tasks or Parameter Sweep Tasks	Time	Critical Interactive	Centralized
CBCS	Park et al. [42]	Bag-of-Tasks or Parameter Sweep Tasks	Throughput	Batch Style	Decentralized
DMBA	Bossche et al. [25]	Map-Reduce Tasks	Time	Batch Style	Centralized
G-SDN	NoviFlow [35]	Graph Processing	Energy	Batch Style	Decentralized
ECAS	Garg et al. [54]	Thread-Based	Cost and energy	Batch Style	Decentralized
PPT	Gmach et al. [44]	Stream Processing	Energy	Critical Interactive	Centralized

C.2 Sustainability Metrics

Table 9 shows the use of sustainability metrics by year in different categories of sustainable cloud computing to measure the performance of numerous infrastructure components of CDCs.

Table 9. Use of Sustainability Metrics in Different Categories by year

Year	Sustainability Metrics
2010	(32)
2011	(40)
2012	(11) (20) (32) (36)
2013	(3) (10) (15) (22) (30) (32) (34) (42)
2014	(5) (9) (18) (24) (29) (35) (41)
2015	(2) (14) (16) (21) (27) (28) (33) (39)
2016	(1) (4) (13) (17) (22) (26) (31) (32) (34) (43)
2017	(6) (8) (12) (15) (19) (23) (27) (30) (36) (38)
2018	(7) (11) (18) (21) (25) (26) (27) (28) (32) (35) (36) (37)

Table 10 presents a brief definition of sustainability metrics as identified from the literature.

Table 10. Sustainability Metrics and Their Definitions

Metric	Definition
Execution Time	Execution time is the amount of time required to execute an application successfully.
Energy Cost	The combination of monetary and non-monetary costs related to generation, transmission, and use of energy.
Network Bandwidth	The number of bits transferred/received in 1 second.
VM Co-Location Cost	The total cost of VM migration from one CDC to another.
Resource Utilization	The ratio of execution time of a workload executed by a particular resource to total uptime of that resource.
Network Power Usage	The amount of electricity use on the network.
Latency	The delay before the transfer of user request for processing.
Storage Throughput	The amount of time taken for the storage system to execute the required operation per second.
Total Cost of Ownership	The addition of direct cost (purchase cost of a CDC) and indirect cost (operational cost of CDC).
Return on Investment	The ratio of net profit to the cost of investment for a CDC.
Capital Expenditure	The amount of money used to obtain, upgrade, and maintain physical components related to a CDC.
Capacity	The capability of an application to execute user tasks using available resources such as power infrastructure, IT devices, and the like.
Memory Usage	The total usage of main memory (RAM) to execute user tasks.
Storage Usage	The total usage of secondary memory (hard disk) to execute user tasks.
Carbon Usage Efficiency	The ratio of total CO2 emissions produced by total CDC energy to energy of IT equipment.
Water Usage Efficiency	The ratio of annual water usage to energy of IT equipment.
Energy Reuse Effectiveness	The ratio of energy (reused) consumed by cooling, lighting, and IT devices to the total energy consumed by IT devices.
Green Energy Coefficient	The ratio of green energy consumed by a CDC to total energy consumption of that CDC.
The Green Index	It is used to measure the economic growth of a company with the environmental consequences of that growth.
Energy Proportionality	The relationship between power consumed and resource utilization.
Coefficient of Performance	It is calculated by dividing the quantity of removed heat to total work done for removal of heat.
Recirculation Ratio	The amount of waste-water that flows through the advanced pretreatment component divided by the amount of waste water that is sent to the final treatment and dispersal component.
Return Heat Index	The measure of the net level of recirculation air in a datacenter.
Supply Heat Index	The measure of suppling heat from outside for recirculation.
Recirculation Index	It is used to measure the amount of water saved while circulating hot water to the water heating system.
Water Economizer Utilization Factor	The percentage of hours in a year that the water side economizer system is used to provide the required cooling to the CDC.
Datacenter Cooling System Efficiency	The amount of cooling capacity per unit of energy that it consumes to maintain the working of a CDC.
Airflow Efficiency	The amount of airflow that a ceiling fan can produce per minute.
Datacenter Temperature	The operating temperature of a CDC.
Thermal Correlation Index	It is used to measure the relation between the heat flux and the thermodynamic driving force for the flow of heat.
Thermodynamic Efficiency	The amount of heat used by a heat utilization system based on the amount of heat received.
Energy Consumption	The amount of electricity expended by a resource to complete the execution of an application.
Energy efficiency	The ratio of the number of workloads executed to total energy consumed by a CDC to execute those workloads.
Average Datacenter Efficiency	The ratio of IT equipment power to total facility power.

(Continued)

Table 10. Continued

Metric	Definition
Computation Power Consumption	The amount of energy consumed during the execution of compute-intensive tasks.
Power Usage Effectiveness	The ratio of the energy consumed by ICT devices to the energy consumed by all the devices, including ICT and cooling devices.
User Satisfaction	It is used to measure how cloud services of a cloud provider fulfill user QoS requirements.
Response Time	The length of time taken for a system to react to a user request
Correctness	The degree to which the cloud service will be provided accurately to the cloud customers.
Reliability	The capability of an application to sustain and produce correct results in case of network-, hardware-, or software-related faults.
Availability	The amount of time (hours) a specific application will be available for use per day.
Privacy	The parameter through which user and provider can store their information privately using authorization and authentication.
Throughput	The ratio of total number of tasks of an application to the amount of time required to execute the tasks.
Application Transaction Rate	The ratio of number of applications coming for processing per unit time.
Security	The ability of the computing system to protect the system from malicious attacks.
Cost	The total money that can be spent in 1 hour to execute the application successfully.
Makespan	The time difference between the start and finish of a sequence of tasks of an application.

C.3 Capacity Planning

Ghosh et al. [109] propose a Stochastic Model–based Capacity Planning (SMCP) technique for virtual infrastructure to serve user requests and execute their workloads within a specified deadline and budget. To improve the capacity of the CDC, an optical network is deployed for enabling sustainability in CDCs to save energy consumption. Kouki and Ledoux [110] designed an SLA-aware Technique (SLAT) for improving capacity planning for cloud-based applications and their QoS requirements. Further, the trade-off between user satisfaction and profit has been identified using a queueing network to plan the configuration of the required CDC autonomically. Jaing et al. [111] designed Cloud Analytics for Immediate Provisioning (CAIP) of VMs and planning capacity for CDCs. Predication error is calculated based on asymmetric and heterogeneous metrics. This error value is used for effective capacity planning, which improves the quality of cloud services and reduces the emissions of carbon dioxide. This technique uses IBM data traces and is effective in reducing VM provisioning time and reducing overhead to improve energy efficiency. Sousa et al. [112] aimed to develop an approach for Capacity Planning for Cloud Infrastructure (CPCI) by considering two important parameters, cost and dependability. A stochastic model generator is used to identify the dependability of servers while developing the cost-effective capacity plan. Further, Moodle [3] hosted on a Eucalyptus-based environment to test CPCI, which shows that CPCI is a cost- and energy-efficient capacity planning technique.

Kong and Liu [113] propose a methodology for Selection of Optimal Energy Source (SOES) for effective capacity planning to design an energy-efficient green datacenter. This research focuses on (i) criteria to choose an energy source, (ii) planning the capacity for cloud infrastructures with minimum energy cost and carbon emissions, and (iii) finding and optimizing the requirements of datacenters to improve service availability. SOES is effective in reducing lifetime total cost, which includes operational as well as capital costs. Carvalho et al. [114] propose a Capacity Planning Framework (CPF) for the cloud market using different price schema, such as on-demand, reservation, and spot, to serve user requests. The CPF helps to identify the sustainable price schema required to execute current workloads. It has been concluded that the spot price schema [139] is cost-effective without degrading the quality of cloud service. Menasce et al. [115] performed

experiments to test the capacity planning model [113] using three different cloud providers: Microsoft Azure, Google's App Engine, and Amazon EC2. SLA is defined based on four QoS parameters [140]: throughput, response time, availability, and cost. Dorsch and Häckel [116] investigated the correlation between environmental sustainability and economic efficiency to enable exchange of extra capacity between different cloud infrastructures to execute cloud workloads. This method can improve resource utilization and reduces cost of energy and response time. Triantafyllidis et al. [149] developed an Integrated Optimization Platform (IOP) to evaluate the effectiveness of capacity planning for infrastructure and resources. The cost-optimal solutions are identified using mixed-integer linear programming in an IOP based on the optimization of objective functions. The performance of the proposed platform is evaluated with respect to energy, system cost, network topology, and emission flow.

Figure 22 shows the evolution of capacity planning techniques along with their Focus of Study (FoS) across the various years.

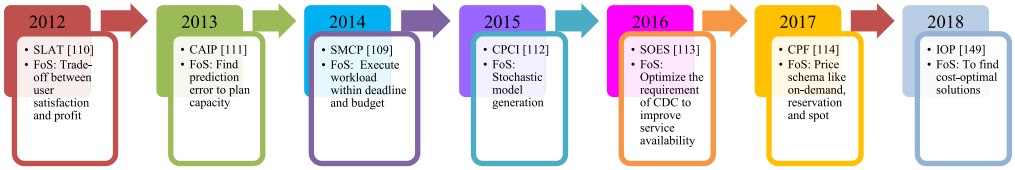


Fig. 22. Evolution of capacity planning techniques.

A summary of these related techniques (capacity planning) and their comparison based on objectives, optimization parameters and metrics along with open research challenges is given in Table 11.

Table 11. Comparison of Existing Techniques (Capacity Planning) and Open Research Challenges

Technique	Organization	Objective	Metric	Optimization Parameter	Citations	Open Research Challenges
SMCP [109]	Duke University, USA	Improve energy efficiency	Capital expenditure	Energy consumption	35	Combining applications improves resource utilization and reduces capacity cost. Cloud workloads should be analyzed before execution to finish their execution on time because some workloads are deadline oriented.
SLAT [110]	INRIA, France	Reduce SLA violation	Total cost of ownership	Energy cost	20	VM migration should be provided for migration of workloads or machines to successfully complete the execution of workloads with minimum use of resources, which improves the energy efficiency of CDCs.
CAIP [111]	Florida International University, USA	Reduce CO_2 emissions	Return on investment	Datacenter Cost	51	An effective data management policy is needed to store data effectively at lower cost and so that data can be accessed easily for modification, deletion, etc.
CPCI [112]	Federal Rural University, Brazil	Identify the dependability of server	Capacity	Cost per dependability	19	Configuration of cloud should be examined for future execution of applications with or without migration of data.
SOES [113]	McGill University, Canada	Design energy-efficient green CDC	Storage usage	Storage cost	10	A strong plan is needed in case of disaster management so that data can be recovered successfully at a later stage with minimum cost.

(Continued)

Table 11. Continued

Technique	Organization	Objective	Metric	Optimization Parameter	Citations	Open Research Challenges
CPF [114]	Federal University of Campina Grande, Brazil	Improve quality of service	Memory usage	Memory cost	1	User requirements should be considered clearly while preparing technical design to achieve maximum user satisfaction.
IOP [149]	Imperial College London, UK	Improve resilience to deal with climate change and poverty	Return on investment and capital expenditure	Energy, system cost, network topology, and emission flow	2	IOP can be extended to support uncertainty in the modeling formulation for other parameters, such as magnitude of demand in resources or capital and operational expenditures.

The comparison of existing techniques based on our capacity planning taxonomy is given in Table 12.

Table 12. Comparison of Existing Techniques Based on Taxonomy of Capacity Planning

Technique	Author	Component	IT Workload	Application Model	Autoscaling	Utility Function
IOP	Triantafyllidis et al. [149]	Power infrastructure and cooling	Batch style	Configuration	Proactive	Cost (energy)
CPF	Carvalho et al. [114]	Power infrastructure	Interactive critical	SLA	Proactive	Cost
SOES	Kong and Liu [113]	IT device	Interactive critical	SLA	Proactive	Latency
CPCI	Sousa et al. [112]	Power infrastructure	Batch style	Configuration	Proactive	Latency
SMCP	Ghosh et al. [109]	IT device	Batch style	SLA	Reactive	Latency
CAIP	Jaing et al. [111]	Power infrastructure	Batch style	Configuration	Reactive	Cost
SLAT	Kouki and Ledoux [110]	Cooling	Interactive critical	SLA	Proactive	Cost

C.4 Energy Management

Domdouzis [15] studied the existing technologies of green computing and its effects on sustainable cloud computing. Further, QoS-based energy-aware scheduling techniques have been discussed and identified the impact of cloud infrastructure on the environment. Moreover, technologies related to sustainable cloud computing—such as web services, capacity planning, and application design—have been discussed. Ficco and Rak [5] stated that sustainable cloud computing optimizes the performance of different applications—such as education, health care, and agriculture—in terms of power consumption. Abbasi [16] studied existing literature on sustainable cloud computing and stated that sustainability and energy efficiency of CDCs is certain to reduce the impact on the environment. To solve this problem, an architecture has been proposed for management of workloads through a multi-layer server that consists of upper and lower levels. The lower level focuses on the management of energy aspects such as energy consumption, cooling power efficiency, and server consolidation, while the upper level focuses on carbon footprints and utility cost [32]. The proposed architecture reduced carbon footprints as well as energy costs [5]. This research considers only homogeneous datacenters without considering the important QoS parameters, such as time, throughput, and security.

Accenture [17] suggests the ecological advantages of moving to the cloud in terms of performance and efficiency of service, which can improve reliability and sustainability of services. This research identified important factors to reduce energy consumption as well as carbon emissions,

such as (i) decreasing power consumption by efficient cooling management (datacenter management), (ii) improving utilization of servers by avoiding underutilization and overutilization of resources (server utilization), (iii) executing a large number of requests using shared infrastructures (multi-tenancy), and (iv) efficient matching of workload and resource for execution (dynamic provisioning). Moreover, the business community can benefit from cloud computing through large numbers of users of cloud services, including mobile applications, online gaming, social media, and email. Thus, there is a need to make cloud services more energy efficient and sustainable, which can fulfill user demands in a timely manner without affecting the environment [6]. This study concluded that the amount of carbon footprint is largely dependent on deployment size. Microsoft deployed 60% large-sized deployments, 30% medium-sized deployments, and 10% small-sized deployments [7].

Gill et al. [60] developed a Particle Swarm Optimization (PSO)–based resource scheduling algorithm (BULLET) for effective management of energy consumption. Initially, user workloads were classified based on QoS requirements for provisioning resources. This PSO-based scheduling algorithm is used for scheduling provisioned resources and workloads are executed successfully using Dynamic Voltage Scaling (DVS). The proposed algorithm is effective in improving energy use, time, and cost of resources in datacenters and enables sustainable computing. Brown et al. [61] propose a Method for the Assessment and Analysis (MAA) of sustainability for energy-efficient cloud services. The MAA has been tested using the case study of different multi-story buildings in Sweden. Further, a Swedish environmental rating tool has been used to calculate the life-cycle cost. It has been concluded that 25% more cost is required to develop an energy-efficient and sustainable cloud environment, which further improves the indoor environmental quality. Hsu et al. [62] propose the Energy Efficient and Sustainable Model (EESM) for management of a transport system, which (i) provides an effective management of data (video and audio) using available storage, (ii) improves the energy use and reliability of services, and (iii) identifies effective route and status based on the analysis of traffic conditions such as traffic lights and traffic flow. The EESM helps to reduce energy consumption and fuel consumption, which enables sustainable eco-driving. Uddin and Rahman [64] have investigated large amounts of energy required to run CDCs in an efficient manner and have identified the following issues that affect global warming: backup and recovery, low carbon emissions, and huge energy consumption. Further, a framework is proposed to decrease carbon emissions by dividing datacenters into different resource pools for efficient management of energy consumption and green metrics that is, PUE [68], which is used to measure the effectiveness of a datacenter. Singh et al. [108] analyze the QoS-aware autonomic computing systems and have identified that energy is one of the important parameters, which requires optimization to enable sustainable cloud computing.

Giacobbe et al. [65] review the cost-effective energy-aware resource management technique in cloud federation, finding that cost saving is an important factor for sustainable computing. Most energy-saving techniques are based on cloud federation, and use of resources in datacenters is also affected by large energy consumption. It has been reported that the dynamic migration of computational resources reduces energy consumption during the interaction of IoT devices [73]. Kramers et al. [66] proposes the genetic algorithm–based Energy-Aware Resource Allocation (EARA) technique for allocation of VMs on heterogenous CDCs. Further, a new metric named POWERMARK has been designed to find the energy efficiency of CDCs and reduce VM co-location cost and bandwidth cost to fulfill cloud user requirements. Gill et al. [67] propose a QoS-aware scalable resource management policy (CHOPPER) for effective scheduling of resources by considering self-optimization for improving energy utilization, self-protection against cyber-attacks, self-configuration of energy-efficient cloud resources, and self-healing by managing unexpected failures. Further, energy consumption of resources has been reduced using an autonomic system

to execute workloads in sustainable datacenters. Chen et al. [69] investigate the trade-off between throughput and energy in cloud-based radio access networks. In this network, renewable energy is used to provide the power to multiple base stations to exchange information between senders and receivers. It has been concluded that energy consumption can be reduced by selecting optimal paths to transfer data in the network. Further, data is managed efficiently at the sender as well as receiver side using a cloud repository, which improves the throughput and sustainability of datacenters [70]. Singh and Chana [83] have developed an energy-aware resource scheduling technique (EARTH) through resource consolidation, which considers fuzzy logic to make scheduling decisions for the execution of cloud workloads. A Dynamic Frequency Scaling (DFS)–based proposed technique [77] is effective in improving energy efficiency and resource use due to the timely decisions of the resource scheduler.

Dandres et al. [74] propose an Energy-based Resource Management (ERM) approach to reduce greenhouse emissions of datacenters and it has been identified that server load migrations affect the energy consumption of datacenters. Xu et al. [75] identified that DVFS manages the datacenter effectively but these techniques failed in the case of overloaded data. Authors used brownout to determine the overloads effectively and managed the load by deactivating the free resources to save energy, which improves sustainability of CDCs. Wang et al. [76] recognize that data coming toward a cloud in a short time are difficult to manage; this issue is solved by incorporating multiple mobile sinks for data gathering. Further, a time adaptive schedule policy is used to reduce latency triggered by arbitrary allocation of tasks. The optimization of latency and the effective gathering of data reduce energy consumption, which creates a sustainable cloud. Garg and Buyya [80] propose the Green Cloud Framework (GCF) to minimize carbon footprints and to measure the impact of carbon emissions to the environment. In this research work, it is mentioned that there is a need for holistic management of energy to make CDCs more sustainable by minimizing their overall power consumption. Mardani et al. [82] investigate multi-objective-based decision-making techniques for sustainable and renewable energy to identify the parameters (social, political, environmental, technical, and economic) that are affecting the environment. It has been suggested that energy should be managed efficiently at different levels—social, political, environmental, technical, and economic—to make clouds more sustainable in the future. Singh et al. [125] propose an SLA-aware resource allocation mechanism (STAR) for efficient management of resources by considering scalable components: processor, storage, and memory. This technique uses DVFS for power management of cloud resources in sustainable CDCs. Experimental results prove that STAR is effective in decreasing energy consumption of resources, which can effectively provide the sustainable cloud environment without violation of the SLA. Battistelli et al. [41] propose a Cloud-based Energy-aware Service Automation (CESA) technique for a sustainable cloud datacenter to reduce energy consumption during the execution of cloud resources. The authors used a MAPE-k loop to offer automation of cloud service, using an efficient communication path to transfer data from source to destination with the maximum value of energy efficiency. The experimental results show that the CESA technique is effective in providing the required cloud service with a maximum value of energy efficiency.

Figure 23 shows the evolution of energy management techniques along with their FoS across various years.

A summary of related techniques (Energy Management) and their comparison based on objectives, optimization parameters, and metricz along with open research challenges is presented in Table 13.

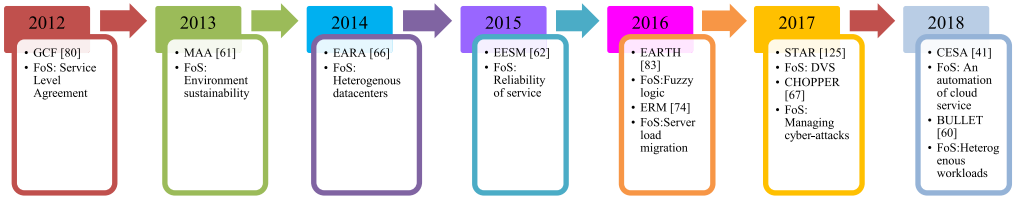


Fig. 23. Evolution of energy management techniques.

Table 13. Comparison of Existing Techniques (Energy Management) and Open Research Challenges

Technique	Organization	Objective	Optimization Parameter	Metric	Citations	Open Research Challenges
CESA [41]	RWTH Aachen University, Germany	Improve energy efficiency	Energy consumption	Energy consumption	1	Trade-off between energy consumption and reliability is an open research challenge.
BULLET [60]	Thapar University, India	Reduce energy consumption	Execution time	Energy consumption	3	Resource utilization of CDCs is affected during workload execution.
MAA [61]	Royal Institute of Technology, Sweden	Improve energy utilization	Energy cost	Energy consumption	53	Switching resources between high-scaling and low-scaling modes increases response time.
EESM [62]	Yuan Ze University, Taiwan	Improve energy efficiency	Energy consumption	Energy efficiency	34	Cannot handle the dynamic nature of tasks and SLA violation can be reduced.
EARA [66]	Royal Institute of Technology, Sweden	Improve energy efficiency	VM co-location cost and bandwidth	Computation power consumption	16	Putting servers in sleeping mode or turning on/off servers affects the reliability of the storage component.
CHOPPER [67]	Thapar University, India	Improve energy utilization	Execution time and cost	Energy efficiency	6	Resources are reserved in advance, but resource requirement is less than resources available, which increases cost.
EARTH [83]	Thapar University, India	Improve resource utilization	Energy	Energy consumption	25	A large amount of clock speed is wasted while waiting for the data because of the speed gap between processor and main memory.
ERM [74]	University of Montreal, Canada	Reduce SLA violation	Energy consumption	Average datacenter efficiency	1	Energy consumption can be saved by reducing processor frequency through the manipulation of supply voltage.
GCF [80]	The University of Melbourne, Australia	Reduce carbon footprints	Power cost	Power usage efficiency	133	A large number of workloads are waiting for execution owing to unavailability of a sufficient amount of resources.
STAR [125]	Thapar University, India	Reduce greenhouse emissions	Execution cost	Power usage efficiency	17	Switching resources between high-scaling and low-scaling modes increases service delay.

Energy cost and CO_2 emission for static and dynamic energy management techniques [122, 144, 145] are shown in Figure 24.

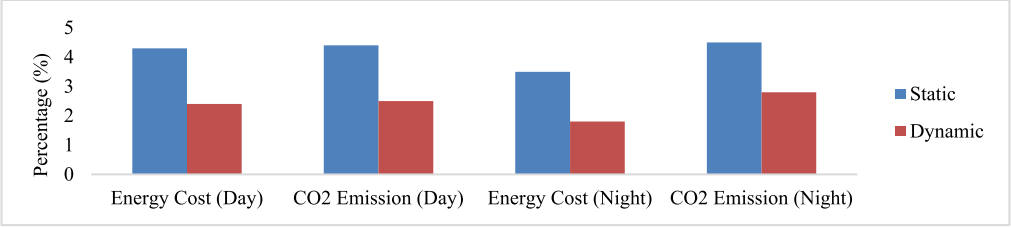


Fig. 24. Energy cost and CO_2 emission for static and dynamic energy management techniques (Data Sources: [122, 144, 145]).

It is clearly shown that static energy management techniques consume more energy and produce more CO_2 emissions as compared to dynamic energy management techniques.

The comparison of existing techniques based on our energy management taxonomy is given in Table 14.

Table 14. Comparison of Existing Techniques Based on Taxonomy of Energy Management

Technique	Author	Resource Management	Processor	Storage	Memory Requirement	Resource Consolidation	Cooling
CESA	Battistelli et al. [41]	Proactive	DVFS	Caching	Large	Yes	Internal
BULLET	Gill et al. [60]	Proactive	DVS	Caching	Large	Yes	Internal
CHOPPER	Gill et al. [67]	Proactive and Reactive	DVFS	Caching	Large	No	Internal
STAR	Singh et al. [125]	Reactive	DVS	Caching	Large	Yes	Internal
EARTH	Singh et al. [83]	Reactive	DFS	Replication	Large	Yes	Internal
ERM	Dandres et al. [74]	Proactive	DFS	Replication	Small	No	Internal
EESM	Hsu et al. [62]	Reactive	DFS	Replication	Small	Yes	Internal
EARA	Kramers et al. [66]	Reactive	DVS	Caching	Large	No	Internal and External
MAA	Brown et al. [61]	Proactive	DVS	Replication	Small	No	Internal
GCF	Garg and Buyya [80]	Reactive	DVFS	Caching	Small	No	Internal and External

C.4.1 DVFS. DVFS is an energy optimization power management technique, which is basically the adjustment of frequency settings of computing devices in order to optimize the resource allotment for tasks. If resources are not required, then DVFS minimizes the CPU frequency by supplying lower voltage to the CPU, which maximizes the power savings [60]. The DVFS technique is used for VMs hosted by physical machines along with the algorithm or scheduling mechanism to reduce the energy. The DVFS system and workload planning can be joined in two ways: (1) workload scheduling and (2) slack reclamation. In schedule generation, DVFS-empowered processors are used to (re)schedule tasks of the graph in a worldwide cost function including both makespan (execution time) and energy saving to meet both time and energy limitations in the meantime [3]. In slack reclamation, which fills in as a post-preparing method on the yield of

planning calculations, the DVFS mechanism is used to limit the power utilization of undertakings in a timetable created by a different scheduler [25]. The current strategies, in light of the DVFS strategy, have two noteworthy inadequacies: (1) a large portion of them center around schedule age and do not adopt enough of the *slack reclamation* strategies into record to spare extra energy, and (2) the current slack reclamation techniques use just a single frequency for every task among all distinct arrangements of the processor's frequencies. Using one frequency generally brings about revealed slack time in which the processor and different devices just waste energy [66]. DVFS-based energy management techniques reduce energy consumption, but response time and service delay are increased due to switching resources between high-scaling and low-scaling modes.

C.4.2 C-states or C-modes. In order to save energy when the CPU is idle, the CPU can be transferred into a low-power mode, which is called C-states or C-modes [61, 62]. The main motive of C-states is to cut the clock signal and power from idle components inside the Central Processing Unit (CPU). The more units stop (by cutting the clock), voltage is needed or even the CPU completely shuts down, the more energy can be saved, but more time is required for the CPU to "wake up" and be 100% operational again [66, 75, 76]. There are main seven cases: (i) C0 mode (Operating State)—the CPU is fully turned on and it consumes the maximum amount of energy; (ii) C1 mode (Halt)—the CPU main internal clocks are stopped via software and it consumes less energy as compared to C0 mode but needs a little time to make the CPU 100% operational; (iii) C2 mode (Stop Grant and Clock)—CPU main internal and external clocks are stopped via hardware and it consumes less energy as compared to C1 mode but needs more time to make the CPU 100% operational as compared to C1 mode; (iv) C3 mode (Sleep)—Stops all internal and external clocks of the CPU; (v) C4 mode (Deeper Sleep)—Stops all internal clocks of the CPU and decreases voltage of the CPU; (vi) C5 mode (Enhanced Deeper Sleep)—Decreases voltage of the CPU even more and turns off the memory cache; and (vii) C6 mode (Deep Power Down)—Decreases the internal voltage of the CPU to any value, including 0V, and this state consumes the minimum amount of energy (depends on the voltage value), but this state needs the maximum amount of time to make the CPU 100% operational.

C.5 Virtualization

VM migration is the process of migrating VMs (a VM is the software implementation of the computer that runs a number of applications and the operating system) to another physical server without interrupting the running application operation. A virtualization takes place when the server is underutilized or overutilized or in the case of temperature overpass, which improves energy efficiency of the CDC. Also, virtualization reduces the carbon emission by moving the workload to a location with a renewable energy supply. Wang et al. [99] proposed a Green-Aware VM migration (GVM) policy for the efficient use of energy coming from grid sources and optimizing the power consumption of cooling and IT devices. A statistical searching approach is used to find the destination of migration, and post-copy technique is used in GVM. Wang et al. [88] proposed an extended version of GVM (E-GVM) by replacing grid energy with renewable energy to make datacenters sustainable, which further reduces power cost and carbon emissions. Further, VM consolidation-based resource allocation is improved using IT-enabled virtualization [100], which measures the variation of energy requirements during pre-copy and post-copy of data from one server to another. Bolla et al. [103] analyzed the migration time of live migration of a number of VMs between physical machines. The migration time is the amount of data copied from one physical machine to another. It also estimates the interference effects during live migration. The Kernel-based Virtual Machine (KVM) is used to test the proposed technique to prove its effectiveness to calculate

migration time more accurately. Khosravi et al. [106] developed a technique to improve the use of renewable energy for Online Virtual Machine Migration (OVMM) from one physical server to another to enable sustainable cloud computing. In this research work, an optimal offline algorithm and an online algorithm are proposed for VM migration [141]. The offline algorithm is effectively working when future information regarding renewable energy is known a priori and the online algorithm is used when future information is unavailable.

Ranjbari and Torkestani [159] propose a Learning Automata-based VM Consolidation (LAVMC) algorithm to reduce energy consumption and SLA violation rate. The LAVMC algorithm decreases the number of migrations and shuts down idle servers to decrease the energy consumption of the CDC. Similarly, Ashraf and Porres [160] propose an Ant Colony Optimization (ACO)–based VM Consolidation (ACOVMC) technique to reduce the number of VM migrations. Dabbagh et al. [101] propose the VM Predication and Migration (VMPM) mechanism for overcommitted clouds to reduce the use of physical machines through predication of VMs and run the datacenter sustainably, thus improving the energy efficiency of CDCs. Further, a load-balancing mechanism distributes the load effectively on physical machines. This technique uses Google workload traces and is effective in reducing overload, reducing energy consumption, and improving resource utilization. Dastagiraiah et al. [162] propose a VMware Off-Loading (VMOL)–based dynamic load-balancing technique to distribute the load effectively on virtual nodes to minimize energy consumption of resources. Giacobbe et al. [102] developed a VM-based Resource Allocation (VMRA) technique (VM scheduling), which schedules virtual resources to decrease carbon footprints efficiently. In this technique, the best green destination is identified to migrate a VM from one server to another with minimum emissions of carbon dioxide. Rybina et al. [105] propose a VM-based Assessment Technique (VMAT) for the investigation of network functions during VM scheduling that need to be optimized for efficient consumption of energy and reduction of carbon footprints. In this research work, virtual Evolved Packet Core [83] is used to estimate the processing latency of VM migration with minimum emissions of carbon footprints. A large amount of energy consumption reduces energy efficiency, which affects the environment and increases power cost [104]. Further, there is a need to define the sustainability level of environmental impact to improve sustainability of CDCs.

Zhao et al. [37] propose a Performance-guaranteed and Power-aware VM Placement (PPVMP) technique to investigate the relationship between CPU use and power consumption to develop a fault tolerance mechanism using a forward error recovery technique. An ACO-based power-aware technique is proposed for VM placement with minimum power consumption. Chinnathambi et al. [157] propose a Scheduling and Checkpointing Optimization (SCO) algorithm for efficient management of VMs for mission-critical applications. The SCO algorithm reduces energy consumption and crash failure errors during execution of applications. Sotiriadis et al. [158] propose a Self-managed VM Scheduling (SVMS) technique for virtual resource management using a bin-packing algorithm [74], which increases infrastructure capacity to enable VM elasticity and maximizes a VM's real CPU use during workload execution. Beechu et al. [161] propose an Energy-efficient VM Fault Tolerance (EVMFT) technique, which focuses on core mapping based on the application core graph and reduces failure using a restart recovery mechanism.

Mishra et al. [40] propose a Task-based VM-Placement (TVMP) algorithm to perform mapping of tasks to VMs and VMs to physical machines for optimization of energy consumption. This VM elasticity–based TVMP algorithm performs searching of the optimal VM and physical machine to fulfill the resource requirement for execution of user tasks. The proposed algorithm is implemented using the CloudSim toolkit [61] and reduces the task rejection rate and makespan. Al-Dhuraibi et al. [163] propose a Docker Containers–based VM Elasticity (DCVME) technique, which scales up and down both CPU and memory assigned to each container according to the application

workload autonomously. Experimental results show that the DCVME technique performs better than the Kubernetes elasticity mechanism [164] in terms of energy efficiency and resource utilization. Metzger et al. [107] suggest that the use of virtualization technology in educational universities can reduce energy consumption as well as carbon footprints. For an efficient resource utilization of CDC, the hypervisor multiplexes VMware-based Operating Systems (OS) to the primary hardware resources (storage, memory, and processor). Nevertheless, a co-hosted VM interference reduces the performance of an application if VMs are isolated poorly. Due to an increase in requirement of datacenters, energy consumption is also increasing, which affects the sustainability of CDCs. To solve this problem, virtualization technology can be used to transfer the data from one server to other via VM migration and consolidation.

Figure 25 shows the evolution of virtualization technology along with its FoS across various years.

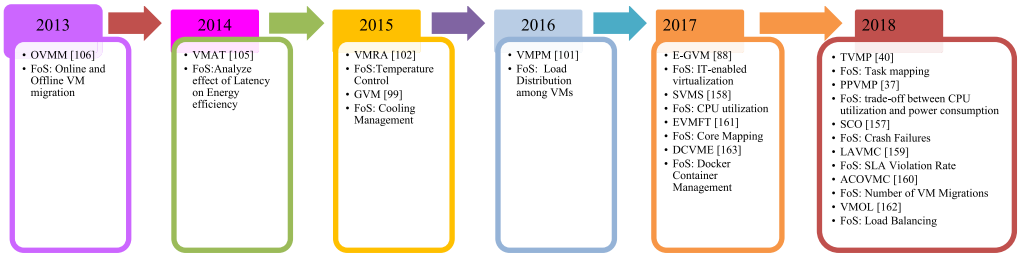


Fig. 25. Evolution of virtualization technology.

A summary of these related techniques (virtualization technology) and their comparison based on objectives, optimization parameters and metrics along with open research challenges is given in Table 15.

Table 15. Comparison of Existing Techniques (Virtualization Technology) and Open Research Challenges

Technique	Organization	Objective	Optimization Parameter	Metric	Citations	Open Research Challenges
TVMP [40]	NIT, Rourkela, India	Optimal searching of VM	Task rejection rate and makespan	Execution time	1	During the execution of workloads, VM fault tolerance is required.
LAVMC [159]	Islamic Azad University, Iran	Reduce energy consumption	SLA violation rate	Energy cost	2	Dynamic energy-aware technique can be adopted to reduce energy consumption in case of idle server.
ACOVMC [160]	Abo Akademi University, Finland	Reduce no. of VM migrations	VM migration cost	VM co-location cost	6	VM fault tolerance is required to deal with different types of faults.
VMOL [162]	K L University, India	Improve energy utilization	Power consumption	Energy cost	2	VM load-balancing mechanism can be improved using nature or bio-inspired optimization algorithms.
SCO [157]	Coimbatore Institute of Technology, India	Reduce number of failures	Number of VM migration	Network bandwidth	-	Replication technique can improve the VM fault tolerance mechanism.

(Continued)

Table 15. Continued

Technique	Organization	Objective	Optimization Parameter	Metric	Citations	Open Research Challenges
SVMS [158]	University of Toronto, Canada	Increase infrastructure capacity	Resource utilization	Resource utilization and energy cost	1	VM consolidation mechanism is required to reduce number of VM migrations.
EVMFT [161]	NIT, Goa, India	Improve fault tolerance	Number of failures per VM migration	Reliability	-	VM load-balancing mechanism can be improved using resource prediction technique.
DCVME [163]	University of Lille, Spain	Improve resource utilization and energy efficiency	CPU utilization and energy consumption	Resource utilization and energy cost	-	Container startup can be accelerated by optimizing the storage driver.
PPVMP [37]	Xidian University, China	Reduce power consumption	CPU utilization	Resource utilization and energy cost	1	VM scheduling mechanism is required for effective management of virtual resources.
GVM [99]	Tsinghua University, China	Improve energy utilization	Power consumption	Energy cost	7	During the execution of workloads, VM load-balancing mechanism is required to balance the load effectively due to decentralized CDCs and renewable energy resources.
E-GVM [88]	Qinghai University, China	Reduce carbon emissions	Power cost	Network power usage	1	To balance the workload demand and renewable energy, VM-based workload migration and consolidation techniques provide virtual resources using few physical servers.
VMPM [101]	Oregon State University, USA	Analyze variation in energy consumption	Resource utilization	Resource utilization and energy cost	13	It is the great challenge for VM elasticity techniques to control the cost in terms of energy consumption and network delay while migrating workloads between distributed resources geographically.
VMRA [102]	University of Messina, Italy	Reduce carbon emissions	Energy consumption	VM co-location cost	15	Increasing the size of VMs is creating another challenge for VM consolidation, which consumes more energy and results in service delay.
VMAT [105]	Technical University, Germany	Improve energy efficiency	Latency	Network bandwidth and latency	20	WAN-based VM migration requires storage migration, which can be an overhead for cost-effective migration.
OVMM [106]	The University of Melbourne, Australia	Improve renewable energy utilization	Throughput	Storage throughput	4	Mainly focuses on minimizing the carbon footprint and ignores performance metrics.

C.5.1 Issues with VM Migration in Geographically Distributed CDCs. In virtualized cloud environments, VM migration is the mechanism used most often to achieve energy efficiency at runtime [88]. The value of migration time rapidly increases as the value of available network bandwidth reduces as does the size of the VM; but migration time can be decreased by consolidation of VMs, which improves resource utilization, and effective use of resources can reduce energy consumption [101]. Furthermore, the energy consumption depends on the size of the VM, which is migrated. Remarkably, for a similar set of VMs, different orders of migrations lead to different migration-time based on VM size [102, 106]. Thus, the trade-off between energy cost and migration time of VMs needs to be investigated, The following are the important issues with VM migration in geographically distributed datacenters [88, 99, 105, 106]:

- It is challenging to transfer VMs on shared bandwidth while maintaining the SLA of an application because the memory size of VMs is dynamic, which varies from 1GB to 50+GBs.
- WAN-based VM migration has many issues, such as higher packet drop ratio, larger communication distances, heterogeneous network architecture design, unpredictable network behavior, greater latencies, and limited network bandwidth, which can increase the possibility of SLA violation.
- VM migration complexity increases with the increase of WAN-based storage migration because live storage migration needs asynchronous and synchronous communication modes to transmit storage blocks from one CDC to another.
- Moreover, existing VM migration techniques are not able to distribute loads dynamically in coordinated manner among various CDCs.
- Due to longer communication distances, secure VM migration on heterogeneous CDCs is a challenging task. Consequently, hijackers acquire hardware states, application-sensitive data, currently hosted applications, and OS kernel states for malicious activities.

C.5.2 Container as a Service (CaaS). Container as a Service (CaaS), such as Kubernetes and Docker, uses resource isolation features of Linux kernel such as Control groups (Cgroups) and kernel namespaces to allow independent containers to run within a single Linux instance to avoid substantial overhead of starting virtual machines on hypervisors [156]. Shifting to container-based deployments can reduce overhead related to deployment of containers. Further, CaaS can increase realization by supporting real-time workloads [40]. Container-based deployment outperforms VM transition-based deployment for following reasons [37, 99]: (a) containers start up very quickly and their launch time is less than 1 second and (b) containers have a tiny memory footprint and consume a very small amount of resources. Moreover, containers permit the host to support more instances simultaneously as compared to a VM-based cloud testbed.

The comparison of existing techniques based on our virtualization taxonomy is given in Table 16 (VM Migration, VM Elasticity, and VM Load Balancing) and Table 17 (VM Consolidation, VM Fault Tolerance, and VM Scheduling).

Table 16. Comparison of Existing Techniques Based on Taxonomy of VM Migration, VM Elasticity and VM Load Balancing

		VM Migration					VM Elasticity				VM Load Balancing	
Technique	Author	Technique	VM Technology	Optimization Criteria	Network Topology	Storage Migration	Scope	Policy	Objective	Mechanism	Performance-aware	Resource-aware
TVMP	Mishra et al. [40]	NA	VMware	NA	NA	NA	IaaS/ PaaS	Non-autonomic	Improve performance	Migration	QoS-based	Bin-Packing
LAVMC	Ranjbari and Torkestani [159]	NA	VMware	NA	NA	NA	NA	NA	NA	NA	QoS-based	NA
ACOVMC	Ashraf and Porres [160]	Pre-copy	VMware	Write Throttling	LAN	Shared Storage	NA	NA	NA	NA	Adaptive	NA
VMOL	Dastagiraiah et al. [162]	Post-copy and Pre-copy	Xen	Write Throttling	LAN	Shared Storage	NA	NA	NA	NA	Adaptive	Dynamic Cluster-based
SCO	Chinnathambi et al. [157]	NA	KVM	NA	NA	NA	SaaS	Autonomic	Reduce energy	Migration/ Replication	NA	NA
SVMS	Sotiriadis et al. [158]	NA	VMware	NA	NA	NA	SaaS	Non-autonomic	Increase in-frastructure capacity	Migration	QoS-based	Bin-Packing
EVMFT	Beechu et al. [161]	NA	KVM	NA	NA	NA	NA	NA	NA	NA	NA	Agent-based
DCVME	Al-Dhuraibi et al. [163]	NA	KVM	NA	NA	NA	IaaS/ PaaS	Non-autonomic	Reduce energy	Migration/ Replication	QoS-based	Dynamic Cluster-based
PPVMP	Zhao et al. [37]	Post-copy and Pre-copy	VMware	Compression	WAN	Shared Storage	SaaS	Autonomic	Reduce energy	Migration/ Replication	Adaptive	Dynamic Cluster-based
E-GVM	Wang et al. [88]	Post-copy and Pre-copy	Xen	Write Throttling	LAN	Shared Storage	NA	NA	NA	NA	NA	NA
VMPM	Dabbagh et al. [101]	Post-copy	VMware	Write Throttling	WAN	Shared Storage	IaaS/ PaaS	Autonomic	Improve performance	Replication	Adaptive	Dynamic Cluster-based
GVM	Wang et al. [99]	Post-copy	KVM	Compression	LAN	Not-Shared Storage	NA	NA	NA	NA	QoS-based	Agent-based
VMRA	Giacobbe et al. [102]	NA	NA	NA	NA	NA	IaaS/ PaaS	Autonomic	Improve performance	Migration	Adaptive/QoS-based	Dynamic Cluster-based
VMAT	Rybina et al. [105]	Pre-copy	KVM	Compression	WAN	Shared Storage	IaaS/ PaaS	Autonomic	Reduce energy	Replication	Adaptive	Dynamic Cluster-based
OVMM	Khosravi et al. [106]	Post-copy and Pre-copy	KVM	Compression	LAN	Not-Shared Storage	NA	NA	NA	NA	QoS-based	Agent-based

Note: NA: Means Not Applicable.

Table 17. Comparison of Existing Techniques Based on Taxonomy of VM Consolidation, VM Fault Tolerance and VM Scheduling

Technique	VM Consolidation				VM Fault Tolerance				VM Scheduling		
	Resource Assignment Policy	Architecture	Co-location Criteria	Migration Triggering Point	Redundancy	Failure Semantics	Failure Masking	Recovery	Application Type	Operational Environment	Objective Function
TVMP [40]	Static	Centralized	Resource Availability	Historic Data	NA	NA	NA	NA	NA	NA	NA
LAVMC [159]	Static	Centralized	Resource Availability	Heuristic	Software	Arbitrary Errors	Hierarchical Group Masking	-Rollback --Checkpoint ---User Level ----Patch	Workload (Heterogenous)	Dynamic	To reduce power consumption
ACOVMC [160]	Autonomic	Centralized	Resource Availability	Heuristic	NA	NA	NA	NA	NA	NA	NA
VMOL [162]	NA	NA	NA	NA	NA	NA	NA	NA	Workload (Homogenous)	Distributed	To reduce energy cost
SCO [157]	NA	NA	NA	NA	Hardware	Crash Failure Errors	Flat Group Masking	-Rollback --Restart Recovery	Workload (Homogenous)	Distributed	To reduce energy cost
SVMS [158]	NA	NA	NA	NA	NA	NA	NA	NA	Workflow	Dynamic and Dynamic	To reduce power consumption
EVMFT [161]	NA	NA	NA	NA	Hardware	Crash Failure Errors	Flat Group Masking	-Rollback --Checkpoint ---System Level ----Hardware Level	Workload (Homogenous)	Distributed	To reduce energy cost
DCVME [163]	NA	NA	NA	NA	Hardware	Crash Failure Errors	Flat Group Masking	-Rollback --Checkpoint ---User Level ----Library	NA	NA	NA
PPVMP [37]	Autonomic	Centralized	Power	Heuristic	Software	Arbitrary Errors	Hierarchical Group Masking	Forward Error Recovery	NA	NA	NA
E-GVM [88]	Static	Decentralized	Power	Historic Data	NA	NA	NA	NA	Workload (Heterogenous)	Dynamic	To reduce energy cost
VMPM [101]	NA	NA	NA	NA	Hardware	Crash Failure Errors	Flat Group Masking	-Rollback --Checkpoint ---Application Level ----Single-Thread	Workload (Heterogenous)	Dynamic	To reduce power consumption
GVM [99]	NA	NA	NA	NA	NA	NA	NA	NA	Workload (Heterogenous)	Distributed	To reduce power consumption
VMRA [102]	NA	NA	NA	NA	Hardware	Crash Failure Errors	Flat Group Masking	-Rollback --Checkpoint ---Application Level ----Multi-Thread	Workflow	Dynamic	To reduce power consumption
VMAT [105]	Autonomic	Decentralized	Power	Historic Data	NA	NA	NA	NA	Workload (Homogenous)	Dynamic	To reduce energy cost
OVMM [106]	Static	Centralized	Resource Availability	Heuristic	Software	Arbitrary Errors	Hierarchical Group Masking	-Rollback --Checkpoint ---System Level ----Kernel Level	Workload (Homogenous)	Distributed	To reduce power consumption

Note: NA: Means Not Applicable.

C.6 Thermal-Aware Scheduling

Chaudhry et al. [14] explore the existing thermal-aware scheduling techniques developed for efficient management of green datacenters. They stated that the microprocessor is the most important part of the server, which consumes large amounts of electricity and produces heat continuously. Further, efficient cooling management is required to avoid overheating of servers. Moreover, thermal monitoring and profiling approaches provide different techniques for measuring both lower level (microprocessor) and upper level (datacenter) temperatures to reduce generation of heat [142]. The authors did not identify the effect of thermal-aware scheduling on sustainability and energy efficiency of CDCs. Oxley et al. [84] have determined that a sufficient amount of cooling is required to maintain the temperature for smooth working of sustainable datacenters. Further, cloud workloads should be executed before deadline using available electricity but sharing resources creates problems for the execution of different tasks on different cores of available resources [95]. Co-location, power, and thermal-aware resource allocation techniques have been proposed to execute a number of tasks. Experimental results proved that this approach is effective in executing tasks before a deadline and under a temperature constraint.

Cupertino et al. [85] propose the Energy-Efficient Workload Management (EEWM) approach for thermal-aware CDCs, in which workloads and applications are managed effectively by creating their profile. Moreover, a power-aware resource scheduling mechanism has been proposed to execute the workloads, while optimizing energy consumption. Sun et al. [86] extend EEWM by adding a Heat Distribution Matrix (EEWM-HDM) to control the energy effect of servers on other servers in a CDC. Fuzzy-based priority policy is used to make a trade-off between power and cooling and to execute user workloads in a sustainable cloud environment. Guitart [87] discusses a thermal-aware resource management framework to minimize carbon footprints of CDCs. Guitart states that power usage in CDCs is increasing with the increase in number of IT devices. Guo et al. [89] propose the Thermal Storage-based Power and Network (TSPN)–aware workload management framework, which provides integration with green energy and reduces bandwidth costs between CDCs and cloud users. Stochastic cost reduction procedure uses the Lyapunov optimization [14] to create a trade-off between workload delay and energy cost. Fu et al. [92] propose a Temperature-Aware Resource Management (TARM) mechanism by extending TSPN to reduce energy consumption in CDCs. TARM focuses on the reliability of cloud services using a soft server temperature constraint. Shamalizadeh et al. [90] propose the Thermal-Aware Workload Distribution (TAWD) model to reduce heat recirculation in sustainable datacenters. The cooling and computing power requirement is assessed to control the servers in the cloud for the execution of user workloads without violation of SLAs.

Han and Shu [91] propose the DVFS-based Thermal-Aware Energy-Efficient (TAEE) resource scheduling policy to execute workloads while focusing on the reduction of AC and computation energy use. TAEE assumes a linear relationship between CPU frequency and computation energy use to determine the thermal correlation among the servers of CDCs. TAEE improves the energy efficiency of sustainable CDCs. Dou et al. [93] propose the Carbon-Aware Resource Management (CARM) approach, which focuses on cost reduction of electricity to run sustainable CDCs. Further, time-varying system states have been analyzed to identify the trade-off between workload delay and electricity use to run CDCs. Singh et al. [94] extend EARTH [83] and propose a thermal-aware autonomic resource (SOCCER) management policy for the execution of heterogeneous cloud workloads that improve energy efficiency. Zapater et al. [97] propose a DVFS-based Dynamic Consolidation for Allocation of Resources (DCAR) to execute cloud workloads and Garcia proposes a thermal-aware VM allocation technique [96] that is designed to control the temperature of a cloud datacenter during the execution of cloud resources. Chien and Chang [98] propose

Thermal-Aware Scheduling of Resources (TASR) for multi-core architectures, which divides applications into small threads using the concept of dynamic programming and executes those threads using different cores of the processor. The temperature of the datacenter is controlled using both proactive and reactive scheduling procedures to analyze the variation of temperature with increasing number of threads. Oxley et al. [53] propose a Rate-based Thermal-aware Resource Management (RTRM) technique for heterogenous CDCs to execute user-different workloads within their respective deadlines. RTRM technique satisfies power and temperature constraints while a linear regression technique–based co-location interference model co-locates the tasks with the minimum amount of energy consumption. Experimental results show that the proposed technique performs effectively in terms of temperature and power as compared to a greedy and genetic algorithm. Van Damme et al. [50] propose an Optimized Thermal-aware Job Scheduling (OTJS) technique to analyze the CDCs and reduce consumption of energy. The OTJS technique schedules the jobs effectively on cloud resources so that it maintains the temperature of the system below its threshold value, which reduces the required amount of cooling. The performance of the OTJS technique is tested under varying workload conditions; experimental results show that the proposed technique reduces the temperature of CDCs.

Figure 26 shows the evolution of thermal-aware scheduling techniques along with their FoS across various years.

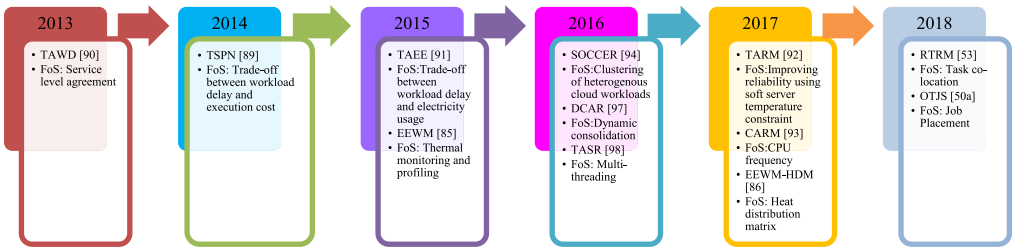


Fig. 26. Evolution of thermal-aware scheduling techniques.

A summary of these related techniques (thermal-aware scheduling) and their comparison based on objectives, optimization parameters, and metrics along with open research challenges is given in Table 18.

Table 18. Comparison of Existing Techniques (Thermal-aware Scheduling) and Open Research Challenges

Technique	Organization	Objective	Optimization Parameter	Metric	Citations	Open Research Challenges
RTRM [53]	Colorado State University, USA	Execute workloads within their deadline	Temperature and power	Datacenter temperature and energy efficiency	5	The relationship between varying optimal temperature distributions and computational capacity can be developed to reduce the response time of jobs.
OTJS [50]	University of Groningen, Netherlands	Reduce cooling cost and energy consumption	Temperature	Datacenter temperature	1	Server consolidation can reduce the number of active racks, which can further reduce consumption of energy.
EEWM [85]	University of Toulouse, France	Improve energy efficiency	Energy consumption	Energy efficiency and datacenter temperature	24	If scheduling is performed based on different thermal aspects such as inlet temperature and heat contribution, then admission control mechanisms at the processor level and server level contradict each other.
EEWM-HDM [86]	University of Toulouse, France	Improve cooling	Power	Energy efficiency and thermodynamic efficiency	21	Datacenter temperature needs to be optimized.
TSPN [89]	University of Florida, USA	Reduce cost	Bandwidth	Network bandwidth and Thermal Correlation Index	61	Large amounts of heat concentration and dissipation affect the performance of CDCs.
TARM [92]	Inner Mongolia University of Technology, China	Reduce energy consumption	Temperature	Datacenter temperature and Thermal Correlation Index	1	A large variation of temperatures also increases the complexity of scheduling and monitoring.
TAWD [90]	University of Aveiro, Portugal	Improve heat recirculation	Energy	Airflow efficiency and energy efficiency	12	Dynamic thermal profiles are required for continuous updating of temperature.
TAEE [91]	Oakland University, USA	Improve energy efficiency	Computation of power consumption	Energy efficiency	4	TAEE technique focused on reducing datacenter temperature, but reduction in temperature may not improve airflow efficiency.
CARM [93]	Xi'an Jiaotong University, China	Reduce electricity cost	Energy cost	Energy efficiency and Thermal Correlation Index	3	There is a need of recirculation coefficient matrix to identify the heat circulation values for every node.
SOCCER [94]	Thapar University, India	Improve energy efficiency	Energy consumption	Energy efficiency	8	It requires the coordination between processor level and server level workload schedulers.
DCAR [97]	University of Madrid, Spain	Reduce energy consumption	Execution time	Energy efficiency and Thermal Correlation Index	2	The energy consumption of CDCs can be minimized by activating those servers adjacent to each other in rack or chassis but power density increases, which creates heat concentration.
TASR [98]	National Chung Cheng University, Taiwan	Analyze variation of temperature	Temperature	Datacenter temperature	1	Reducing cost makes hardware reliability is an open challenge.

The comparison of existing techniques based on our thermal-aware scheduling taxonomy is given in Table 19.

Table 19. Comparison of Existing Techniques Based on Taxonomy of Thermal-Aware Scheduling

Technique	Author	Architecture	Heat Model	Scheduling	Thermometer	Monitoring and Awareness	Simulator
RTRM	Oxley et al. [53]	Multi-core	Thermodynamics model	Proactive	Digital	Thermal Data Filtering and Predictions	FloVent
OTJS	Van Damme et al. [50]	Multi-core	Heat recirculation	Optimized	Digital	Thermal Gadgets	CFD
TARM	Fu et al. [92]	Single core	Heat recirculation	QoS	Digital	Thermal Gadgets	CFD
CARM	Dou et al. [93]	Multi-core	Thermodynamics model	Reactive and Proactive	Analog	Thermal Data Filtering and Predictions	HotSpot
EEWM-HDM	Sun et al. [86]	Multi-core	RC model	Reactive and Proactive	Digital	Manual Profiling and Monitoring	FloVent
SOCCER	Singh et al. [94]	Multi-core	Heat recirculation	Optimized	Analog	Thermal Gadgets	HotSpot
DCAR	Zapater et al. [97]	Multi-core	Thermodynamics model	Proactive	Digital	Thermal Data Filtering and Predictions	CFD
TASR	Chien and Chang [98]	Multi-core	Thermal network	QoS	Analog	Thermal Gadgets	FloVent
TAEE	Han and Shu [91]	Single core	Heat recirculation	Reactive and Proactive	Digital	Manual Profiling and Monitoring	HotSpot
EEWM	Cupertino et al. [85]	Single core	Thermodynamics model	QoS	Digital	Manual Profiling and Monitoring	HotSpot
TSPN	Guo et al. [89]	Single core	Thermal network	Optimized	Analog	Thermal Gadgets	FloVent
TAWD	Shamalizadeh et al. [90]	Multi-core	Thermal network	QoS	Digital	Manual Profiling and Monitoring	CFD

C.7 Cooling Management

Ndukaife and Nnanna [150] propose a Split Air-Conditioning System (SACS)–based cooling management technique, which investigates the characteristics of water consumption to test performance. The authors vary the thickness of the cooling pad to investigate its effect on the coefficient of performance. The SACS technique is efficient in maintaining an adequate quantity of humidity with the same amount of water consumption while cooling the CDC. The SACS technique helps to reduce energy consumption using this cooling process. Wu et al. [151] propose the Weather-Aware Geo-Scheduling (WAGS)–based cooling management technique, which reduces the energy consumption for cooling while distributing the load of end users among different datacenters. A workload distribution model is designed that primarily focuses on the SLA constraints during workload execution. Moreover, trace-driven experiments have been conducted on real clouds to test the performance of the WAGS technique; experimental results show that the proposed technique performs effectively in terms of energy consumption and latency. Sahana et al. [152] propose a Server Utilization-based Smart Temperature Monitoring (SUSTM) technique, which maintains the cooling of CDCs. In this technique, the concept of Mean Utilization Factor is used to find and control the amount of cool air to maintain the operating temperature in and around the servers within a CDC.

Matsuoka et al. [153] propose a Natural Convection-based Liquid Immersion (NCLI) cooling technology for saving energy consumption and space. A CFD simulator [86] is used to test the performance of the NCLI technique; the experimental results show that this technique is effective in improving cooling efficiency, which further improves the value of PUE. Liu et al. [154] propose a Cloud-Assisted Smart Temperature Control (CASTC) system, which uses IoT-sensing technology

to enable green datacenter air conditioning for cooling. The CASTC system has two subcomponents: (i) a cloud management platform, which offers support to the application layer and manages data effectively; and (ii) a datacenter air conditioning system, which includes ventilation and temperature control, air conditioning, and environment monitoring. The CASTC system effectively reduces the energy consumption of CDCs without affecting cooling management. Manousakis et al. [155] propose an Underprovisioning Datacenter Cooling (UDC) technique to reduce the cost of cooling by underprovisioning the infrastructure of CDCs. The authors developed a trade-off between cooling and QoS and they suggested that the processing capacity can be reduced in the case of a relaxed deadline by selecting the cheapest available provisioning. The experimental results show that the UDC technique is capable of reducing cooling costs.

Figure 27 shows the evolution of cooling management techniques along with their FoS across various years.

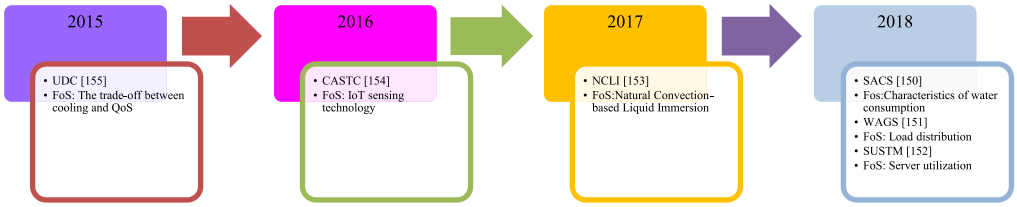


Fig. 27. Evolution of cooling management techniques.

A summary of these related techniques (cooling management) and their comparison based on objectives, optimization parameters, and metrics along with open research challenges is given in Table 20.

Table 20. Comparison of Existing Techniques (Cooling Management) and Open Research Challenges

Technique	Organization	Objective	Optimization Parameter	Metric	Citations	Open Research Challenges
SACS [150]	Purdue University, USA	Reduce energy consumption	Water consumption	Water Economizer Utilization Factor	1	The trade-off between water consumption and energy consumption can be developed.
WAGS [151]	Shanghai Jiao Tong University, China	Reduce cooling cost	Energy consumption and latency	Datacenter Cooling System Efficiency and Latency	1	The nature or bio-inspired load distribution policy can be adopted for further reduction of SLA violation rate.
SUSTM [152]	JIS College of Engineering, India	Maintain operating temperature	Temperature	Recirculation Index	1	The cost of cooling can be reduced by adopting underprovisioning of infrastructure, which can further reduce the temperature of CDCs.
NCLI [153]	Osaka University, Japan	Improve cooling efficiency	Power utilization efficiency	Datacenter Cooling System Efficiency	1	The trade-off between PUE and energy consumption can be developed.
CASTC [154]	Beijing Jiaotong University, China	Reduce energy consumption	Energy efficiency	Water Economizer Utilization Factor	26	Thermal-aware scheduling can improve temperature control.
UDC [155]	Rutgers University, USA	Reduce cooling cost	Cooling cost	Datacenter Cooling System Efficiency	10	Underprovisioning of resources can violate the SLA, which affects the QoS.

The comparison of existing techniques based on our cooling management taxonomy is given in Table 21.

Table 21. Comparison of Existing Techniques Based on Taxonomy of Cooling Management

Technique	Cooling Plant							Cooling Management Techniques
	Medium	Mechanical Equipment	Heat Rejection System	Location	Temperature Range (°C)		Type	
					Supply	Return		
SACS [150]	Air	CRAC	Dry cooler	CRAH	15-25	30-40	DE	Outside air cooling
WAGS [151]	Water	Chiller	Cooling tower	Chiller	10-13	15-18	Chilled water	Chilled water cooling
SUSTM [152]	Water	Chiller	Cooling tower	CRAH	10-13	15-19	Chilled water	Chilled water cooling
NCLI [153]	Air	CRAC	Dry cooler	CRAH	15-25	30-40	DE	Free cooling
CASTC [154]	Air	CRAC	Dry cooler	Rack	17-27	35-40	DE Glycol	Outside air cooling
UDC [155]	Water	Chiller	Cooling tower	Chiller	10-13	15-18	Chilled water	Chilled water cooling

C.8 Renewable Energy

Running CDCs using renewable sources of energy—such as wind, solar, and water—instead of energy generated from fossil fuels (grid electricity) results in lower carbon emissions and lower operational costs. Renewable energy results in an almost zero carbon footprint as compared with brown energy, which results in a reduction of carbon in metric tons. Grid electricity contributes to high operational costs and high energy use. Balasooriya et al. [18] identified that business operations of different cloud providers such as Microsoft, Google, Facebook, Intel, and Amazon are consuming large amounts of electricity continuously, which are increasing the environmental impact on society. To overcome this impact, green cloud computing is required, which can provide more sustainable cloud services. Renewable resources of energy (wind or solar) can be used to produce electricity, which reduces cost due to grid electricity [8]. To make an effective use of renewable energy, datacenters should be located nearer to the source of energy to adopt green cloud computing. Even the use of mixed energy (renewable and non-renewable) generated will increase cost by 13%. Accenture [17] identified that datacenters produced 116 million tons carbon dioxide equivalent ($mtCO_2e$), telecom services produced 307 $mtCO_2e$, and IT devices produced 407 $mtCO_2e$. Owing to the intermittent and unpredictable nature of the renewable power supply, mechanisms should be adopted for making the renewable supply constant as most workloads are dynamic and time sensitive. The two categories of managing renewable energy are (i) dynamic load balancing and (ii) renewable energy-based workload migration.

C.8.1 Dynamic Load-Balancing Technique. In this technique, a CDC is powered by two sources of energy: green and grid [118]. There is a server rack powered by a renewable energy supply and a server rack powered by grid electricity; switching the workload between these two racks occurs based on the energy supply. There is a dynamic load balancing between green electricity and grid electricity according to (i) renewable supply based on weather data and (ii) workload demand based on workload traces [119]. Renewable energy supply is exploited if it is available even for deadline-sensitive workloads. In the case of non-availability of renewable energy, workloads are served by grid supply instead of bringing the server into the low-power state [124]. The design implements a hybrid grid-renewable supply design, where CDCs are powered by either of the energies and VM migration is used to shift the workload between the servers [121].

C.8.2 Renewable Energy–based Workload Migration. The workload is migrated between the geographically located CDCs according to the availability of renewable energy [119, 126]. This

approach requires the availability of a number of CDCs at different locations. Thus, the maximum amount of renewable energy is exploited according to the availability. A dynamic request routing mechanism is used to route the user request or workload toward the datacenter with abundant supply.

Raza et al. [117] propose the Renewable Energy-Aware (REA) approach, which chooses an energy source (hydrogen fuel cell, lithium polymer battery, and lead acid cell) to store power. This study identified that hydrogen fuel cell is effective in providing long life to stored energy. Further, a sustainability index is designed based on environmental, biological, technical, and economic factors to identify the energy source based on user requirements to make it more eco-friendly. Pierie et al. [118] proposed an Industrial Metabolism Approach (IMA) to find green gas production pathways for effective use of renewable energy to generate power for the smooth working of CDCs in a sustainable manner. A decentralized energy system is simulated using Material Flow Analysis [18] to calculate indirect material and energy requirements. Temporal dynamics identify energy consumption, carbon footprint, and environmental impact to find the sustainable pathway. Toosi et al. [119] proposed a Renewable-Aware Technique (RAT) for sustainable datacenters to perform load balancing of web applications geographically. Based on the availability of renewable resources, load balancing of web application requests is distributed geographically and processed at different sites of resources. Real traffic of Wikipedia is used as a workload to test the proposed technique; experimental results prove that RAT is effective in utilization of green resources. Petinrin and Shaaban [120] analyze available renewable energy-aware approaches for the sustainability of CDCs available in Malaysia and identify the available renewable energy sources. It has been concluded that the use of renewable energy sources reduces carbon emissions and creates an eco-friendly environment.

Stein [121] proposes the Multi-Criteria Model (MCM) to evaluate the feasibility of renewable energy, considering four types of energy sources: wind energy, solar energy, hydropower, and biomass. Economic, social, and ecological aspects of sustainability are considered for development of renewable energy. Andrae and Edler [122] analyze the use of renewable and electrical energy for communication technology such as social networking websites, mobile applications, and banking websites. This study states that 4G devices consume more energy as compared to 3G and 2G devices. It is challenging to quantify the sustainability of renewable energy resources. To measure sustainability, gray rational analysis based on an indicator has been developed [123] by considering economic, social, and ecological aspects of sustainability. Liu et al. [124] propose a Cooling and Renewable Aware (CRA) approach for a sustainable cloud infrastructure to manage cloud workloads effectively. In this research work, three important aspects of sustainable computing are considered such as—IT workload, power infrastructure, and cooling—for the execution of user requests and IT workload that can be batch style or critically interactive. CRA manages workloads effectively and reduces power consumption using a cooling-aware technique, which leads to sustainable cloud infrastructures. Mardani et al. [126] propose a Multi-Criteria Decision Making (MCDM) technique for sustainable and renewable energy, finding that biomass and hydropower are much less effective energy sources than solar and wind energy sources owing to use of extra land. Xu et al. [127] examined the assessment of synchronization of renewable energy and grid energy to enable a sustainable cloud infrastructure. Three important factors—power generation, power distribution, and power transmission—have been assessed, along with economic operation, stability, and security of power systems, to generate a sustainability index. Khosravi and Buyya [148] propose a Short-Term Prediction (STM) technique using a Gaussian mixture model, which aids in forecasting future energy levels. Experimental results show that the STM technique is able to predict 15 minutes ahead with 98% accuracy.

Figure 28 shows the evolution of renewable energy techniques along with their FoS across various years.

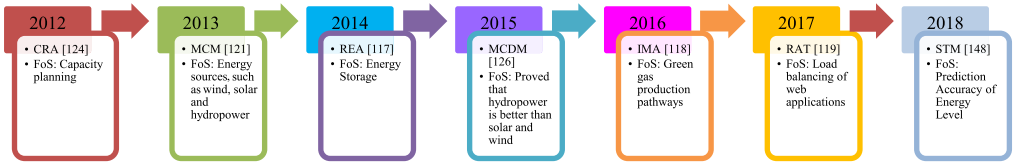


Fig. 28. Evolution of renewable energy techniques.

A summary of these related techniques (renewable energy) and their comparison based on objectives, optimization parameters, and metrics along with open research challenges is given in Table 22.

Table 22. Comparison of Existing Techniques (Renewable Energy) and Open Research Challenges

Technique	Organization	Objective	Metric	Optimization Parameter	Citations	Open Research Challenges
REA [117]	Masdar Institute of Science and Technology, UAE	Choose effective energy source	Carbon Usage Efficiency and Water Usage Efficiency	Sustainability Index	31	Larger capital cost of green resources.
IMA [118]	Hanze Research Centre Energy, Netherlands	Reduce carbon footprints	Green Energy Coefficient	Energy consumption	19	There is a need for hybrid design of energy generation that uses energy from renewable resources and grid resources.
RAT [119]	The University of Melbourne, Australia	Improve resource utilization of green resources	Energy Proportionality Coefficient	Energy utilization	8	A huge of amount of power is lost during transmission from renewable source to CDC site.
MCM [121]	Great Valley School of Graduate Professional Studies, USA	Evaluate the feasibility of energy sources	The Green Index	Energy cost	112	CUE can be optimized by adding renewable energy resources.
CRA [124]	California Institute of Technology, USA	Improve cooling management	Energy Reuse Effectiveness	Power usage	294	There is need to investigate capacity planning in terms of energy cost to cover other important aspects of sustainable computing.
MCDM [126]	Universiti Teknologi Malaysia, Malaysia	Reduce carbon footprints	Carbon Usage Efficiency and The Green Index	Energy utilization	53	The issue of unpredictability in supply of renewable energy and demand of CDCs needs to be addressed effectively.
STM [148]	The University of Melbourne, Australia	Predict energy level	Green Energy Coefficient	Energy prediction	1	Cloud providers can use STM technique to perform online VM migrations.

The comparison of existing techniques based on our renewable energy taxonomy is given in Table 23.

Table 23. Comparison of Existing Techniques Based on Taxonomy of Renewable Energy

Technique	Author	Workload Scheduling	Focus	Source of Energy	Location-aware	Storage Device
STM	Khosravi and Buyya [148]	Dynamic load balancing	Improve accuracy of prediction	Solar and wind	On-site	Batteries
RAT	Toosi et al. [119]	Dynamic load balancing	Reduce energy consumption	Wind	On-site	Batteries
IMA	Pierie et al. [118]	Dynamic load balancing	Reduce energy consumption	Solar	Off-site	Batteries
MCDM	Mardani et al. [126]	Dynamic load balancing	Meet deadline	Hydrogen Fuel Cells (HFC)	On-site	Net-metering
REA	Raza et al. [117]	Dynamic load balancing	Meet deadline	Solar and Water	Co-location	Net-metering
MCM	Stein [121]	Power-preserving	Improve renewable energy utilization	HFC	On-site	Net-metering
CRA	Liu et al. [124]	Power-preserving	Improve renewable energy utilization	Solar and HFC	On-site	Net-metering

C.9 Waste Heat Utilization

Waste heat dissipates from various electronic components (servers use about 40% to 50% of the energy to cool down in CDCs) [132]. Currently, a vast amount of waste heat is generated by CDCs owing to rising demand of cloud-based services and performance. Waste heat utilization is implemented to capture and reuse the waste heat (to heat power plants), thus reducing the load on the cooling equipment that is used to cool the servers. It also contributes to reducing carbon emissions and operational costs. Waste heat recovery is better done on-site, as the heat generated is not of good quality and a large quantity of heat is wasted during the transfer to off-site heat recovery sites. Thus, on-site Waste Heat Utilization (WHU) is more efficient. Waste heat utilization techniques can be put into two broad categories: (i) air recirculation and (ii) power plant co-location.

***Air recirculation*:** This technique enables the reuse of waste heat in a CDC. In this technique, the servers are placed with their front ends facing each other (known as cold aisles) so that the back end of the servers face each other (known as hot aisles). The cool air from the CRAC (Computer Room Air Conditioning) unit is supplied to the cold aisles either through raised floor design or through a diffused ceiling [143]. The warm air produced in the hot aisles is captured and transferred to the CRAC unit. The chiller absorbs the heat in CRAC and outlets the cool air, which is then supplied back to the cold aisles [134].

Power plant co-location: One of the common techniques for waste heat utilization is to co-locate the datacenter with a power plant powered with fossil fuels. This helps in indirect power generation of the plants. The heat dissipated from the datacenter is used to heat the power plant and contributes in reduction of fossil fuels and carbon footprints [133]. The low-quality heat is used to preheat the boiler feed water. As a condenser, the counterflow heat exchanger transports the datacenter heat to the water of power plant. Power plant efficiency is improved and reduces the cost in terms of carbon taxes and coal use.

Waste heat utilization techniques convert the energy dissipated by network components into other useful types of energy and provide the following advantages: (i) revenue generation due to selling of waste heat, (ii) reduction in operational cost, (iii) heat that can be used for vapor-based cooling systems, (iv) reduction of load on cooling equipment because it is utilized effectively to heat power plants, and (v) reduction of carbon footprints. Waste heat utilization of CDCs can be measured by a metric called Energy Reuse Effectiveness (ERE), which is the ratio of *reused energy*

to *total energy* [132]. The quality of heat affects the amount of work, which can be done by utilizing heat.

Heat can be recovered from hot streams using an energy recovery heat exchanger to utilize this heat to generate energy, which can be used to run cloud infrastructures without affecting the environment [129]. Markides [63] analyzes the existing techniques of waste heat utilization to enable sustainable computing in the United Kingdom. Fossil fuel–based energy generators are compared with renewable energy sources and the role of power schemes, combining heat, and pumped heat is discussed. Load factor (heat-to-power demand ratio) calculates the efficiency of different WHU technologies, such as power schemes, combined heat, and pumped heat. It has been found that pumped heat is an effective way to transfer waste heat to useful work as compared to others. Chae et al. [128] designed a technique to assess the available techniques for the utilization of waste heat in an eco-industrial park to make it eco-friendly and economical. Due to the continuous increase in oil prices globally, it is challenging to fulfill multinational companies' energy consumption needs to run their CDCs. This study remarked that quantity of waste heat and energy cost can be decreased by developing more eco-industrial parks, which will be helpful to maintain sustainable environments. Karellas and Braimakis [130] propose a Biomass Co-location Aware (BCA) waste heat utilization technique to produce energy from waste heat. This technique works by utilizing computational fluid dynamics at high temperatures. Freeman et al. [131] propose a Solar Organic Rankine Cycle (SORC)–based indirect power generation technique for waste heat utilization. Thermal energy storage devices are used to store the generated heat and SORC has high electrical work output per unit storage volume as compared to BCA. Du et al. [132] propose a Piezoelectric and Thermoelectric–based Direct Power Generation (PTDPG) technique for waste heat utilization where a fabric device is used to store energy for future use. This technique uses a thermoelectric power generator, which is flexible and air-permeable and it can be effective in high-speed wind areas. Helm et al. [133] proposed the Absorption Chiller–based Solar Heating and Cooling (ACSHC) model to utilize waste heat and seasonal energy efficiency ratio. The proposed technique is effective as compared to PTDPG. Latent heat storage is used to generate energy at low temperatures. Ayompe and Duffy [134] propose a Water Heating System (WHS) using stand-alone wind turbine generators to produce energy. WHS reduces consumption of fuels by maintaining a constant temperature of the water flow of pumps. Experimental results prove that WHS performs better in energy saving and cost reduction. Oró et al. [147] propose a Dynamic Energy model for Heat Reuse (DEHR) analysis, which uses liquid-cooled CDCs for waste heat reuse. A case study of an indoor swimming pool is used to validate the proposed model and experimental results show that the DEHR model reduces operational expenses by 18%.

Figure 29 shows the evolution of waste heat utilization techniques along with their FoS across various years.

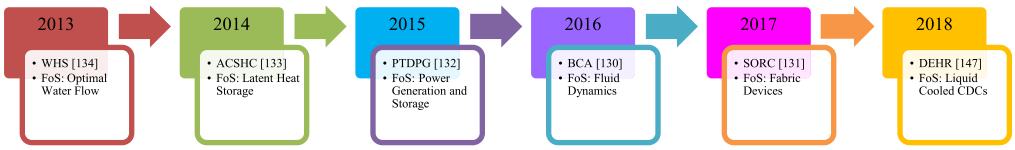


Fig. 29. Evolution of waste heat utilization techniques.

A summary of these related techniques (waste heat utilization) and their comparison based on objectives, optimization parameters, and metrics along with open research challenges is given in Table 24.

Table 24. Comparison of Existing Techniques (Waste Heat Utilization) and Open Research Challenges

Technique	Organization	Objective	Metric	Optimization Parameter	Citations	Open Research Challenges
BCA [130]	National Technical University of Athens, Greece	Improve waste heat utilization	Coefficient of Performance (CoP)	Energy utilization	60	Power densities of servers are increasing by using stacked and multi-core server designs, which further increases the cooling costs.
SORC [131]	Imperial College London, UK	Improve power storage	Return Heat Index, CoP	Storage capacity	4	The energy efficiency of CDCs can be improved by reducing the energy usage in cooling.
PTDPG [132]	Deakin University, Australia	Reduce power consumption	Recirculation Ratio, CoP	Heat transfer rate	59	Implantation of CDCs near free cooling resources reduces the cooling cost.
ACSHC [133]	Bavarian Center for Applied Energy Research (ZAE Bayern), Germany	Improve energy efficiency	Supply Heat Index, CoP	Energy saving	17	Due to consumption of large amounts of energy, CDCs are acting as a heat generator.
WHS [134]	Dublin Institute of Technology, Ireland	Reduce consumption of fuels	Recirculation Ratio	Temperature	73	Quality of air is degrading by shifting of air-based cooling systems to water-based.
DEHR [147]	Catalonia Institute for Energy Research, Spain	Improve heat reuse	Coefficient of Performance	Operational expenses	-	Waste heat reuse can be increased by reducing the use of fossil fuel, which further reduces urban CO2 emissions.

The comparison of existing techniques based on our waste heat utilization taxonomy is given in Table 25.

Table 25. Comparison of Existing Techniques Based on Taxonomy of Waste Heat Utilization

Technique	Author	Focus of Study	Location-aware	Heat Transfer Method	Cooling Method
DEHR	Oró et al. [147]	Allocate modes of CDC to heating model	On-site	Water to water	Using air and fluid
SORC	Freeman et al. [131]	Allocate modes of CDC to heating model	Off-site	Water to water	Using fluid
BCA	Karellas and Braimakis [130]	Drive heat for on-site cooling	Co-location	Water to water	Using air
PTDPG	Du et al. [132]	Drive heat for on-site cooling	On-site	Air to air	Using fluid
ACSHC	Helm et al. [133]	Drive heat for on-site cooling	On-site	Air to air	Using air and fluid
WHS	Ayompe et al. [134]	Allocate modes of CDC to heating model	Co-location	Air to air	Using air and fluid

D OUTCOMES

The main aim of this systematic review is to discover the existing research related to sustainable cloud computing as per the research questions specified in Table 2. We have considered 142 research papers; 55 articles out of 142 are published in leading journals and the remaining are published in prominent workshops, symposiums, and conferences on sustainable cloud computing. Table 26 lists the conferences, symposiums, books, and journals publishing the research mostly related to sustainable cloud computing, which includes the number of research articles that report sustainable cloud computing as main research from each source. It has been observed that

conferences such as International Conference on Cloud Computing, International Conference on Advance Computing, International Conference on Cloud Computing Technology and Science, International Symposium on Sustainable Systems and Technology, and Workshop on Dependable Systems and Networks contribute a major part of the research papers. Leading journals and IEEE Transactions such as Future Generation of Computer Systems, Transactions on Sustainable Computing, Transactions on Cloud Computing, Concurrency and Computation: Practice and Experience, Renewable and Sustainable Energy Reviews, Journal of Parallel and Distributed Computing, Transactions on Parallel and Distributed Systems, ACM Computing Surveys, Environmental Modelling & Software and Applied Energy contributed expressly toward sustainable cloud computing, which is our review area.

Table 26. Journals/Conferences/Books Reporting Most Sustainable Cloud Computing–Related Research

Publication Source	Publisher	Type	#	N
Future Generation Computer Systems	Elsevier	J	7	16
Concurrency and Computation: Practice and Experience	Wiley and Johns	J	1	4
ACM Computing Surveys	ACM	J	5	9
PhD Thesis	Arizona State University	PT	1	1
Renewable and Sustainable Energy Reviews	Elsevier	J	4	11
International Journal of Information Management	Elsevier	J	2	5
Transactions on Cloud Computing	IEEE	T	5	14
Parallel and Distributed Computing	Elsevier	J	4	8
Communications Magazine	IEEE	M	2	6
Cluster Computing	Springer	J	5	11
Computers & Industrial Engineering	Elsevier	J	1	3
Computers in Industry	Elsevier	J	1	4
Journal of Cleaner Production	Elsevier	J	2	4
International Symposium on Sustainable Systems and Technology	IEEE	S	2	5
International Symposium on Systems Engineering	IEEE	S	2	5
International Symposium on Cloud Computing	ACM	S	1	2
International Symposium on Advancement of Construction Management and Real Estate	Springer	S	1	2
Dependable Systems and Networks Workshops	IEEE	W	1	3
Recent Trends in Telecommunications Research Workshop	IEEE	W	1	3
White Paper	NoviFlow Inc.	WP	1	1
Journal of Manufacturing Science and Engineering	Elsevier	J	1	2
Mobile Networks and Applications	Elsevier	J	1	2
Journal of Software: Evolution and Process	Elsevier	J	1	4
Book Chapter of Large-Scale Distributed Systems and Energy Efficiency: A Holistic View	Wiley and Johns	BC	1	5
Transactions on Sustainable Computing	IEEE	T	5	8
Book Chapter of Intelligent Distributed Computing	Springer	BC	1	2
Journal of Network and Systems Management	Elsevier	J	1	2
Computer Network	Elsevier	J	2	3
Environmental Modelling & Software	Elsevier	J	2	3
Systems Journal	IEEE	J	2	1
International Conference on Network and Service Management	IEEE	C	1	2

(Continued)

Table 26. Continued

Publication Source	Publisher	Type	#	N
International Conference on Cyber, Physical and Social Computing	IEEE	C	1	4
International Conference on Computing, Networking and Communications	IEEE	C	1	5
International Conference on Communications	IEEE	C	1	5
International Conference on Cloud Computing Technology and Science	IEEE	C	1	2
International Conference on Cyber-Physical Systems	IEEE	C	1	2
International Conference on Software Quality, Reliability and Security	IEEE	C	1	3
International Conference on Network Softwarization	IEEE	C	1	4
International Conference on Advance Computing	IEEE	C	1	5
International Conference on Design Science Research in Information Systems and Technology	Karlsruhe Institute of Technology	C	1	5
International Conference on Cloud Computing	IEEE	C	1	2
International Conference on Cloud Networking	IEEE	C	1	2
Journal of Intelligent & Fuzzy Systems	IoS Press	J	1	3
Ad Hoc Networks	Elsevier	J	2	3
Sustainable Computing: Informatics and Systems	Elsevier	J	1	1
Transactions on Parallel and Distributed Systems	IEEE	J	3	5
PhD Thesis	University of Trento	PT	1	4
PhD Thesis	University of Madrid	PT	1	5
Journal of Systems Architecture	Elsevier	J	1	5
Simulation Modelling Practice and Theory	Elsevier	J	1	2
IT Professional	IEEE	M	1	2
Transactions on Services Computing	IEEE	T	1	3
Transactions on Network and Service Management	Elsevier	T	1	3
Applied Energy	Elsevier	J	3	6
International Journal of Energy Research	Elsevier	J	1	2
SIGMETRICS Performance Evaluation Review	ACM	M	2	4
Sustainability	MDPI	J	1	5
Energy	Elsevier	J	1	5
Energy Conversion and Management	Elsevier	J	1	2
Applied Thermal Engineering	Elsevier	J	2	2
Energy Procedia	Elsevier	J	1	3
Sustainable Cities and Society	Elsevier	J	1	3
Heat Transfer Engineering	Elsevier	J	1	4
Environmental Modelling & Software	Elsevier	J	2	5
IEEE Access	IEEE	J	1	5
Journal of Organizational and End User Computing	IGI Global	J	1	2
Computer Networks	Elsevier	J	2	2
Information and Software Technology	Elsevier	J	1	3
SIGOPS Operating Systems Review	ACM	J	1	3
Communications Surveys & Tutorials	IEEE	J	1	1
Computing	Springer	J	1	2

J—Journal, C—Conference, W—Workshop, S—Symposium, T—Transactions, WP- White Paper, M- Magazine, BC -Book Chapter, PT- PhD Thesis and N—Number of studies reporting sustainable cloud computing as prime study, #—Total number of articles investigated.

Figure 30 depicts the number of research articles discussing various categories of sustainable cloud computing from 2010 to 2018. Several drifts can be grasped for different categories of sustainable cloud computing. Research in the area of renewable energy has been consistent since 2012 and research on thermal-aware scheduling and cooling management increased sharply in 2016, now a hotspot area for sustainable cloud computing. The number of papers published in the area of application design rose sharply in 2014 and was a very important research area in 2017. The most research work has been done on important areas of energy management. On the other hand, the number of papers published in the area of virtualization, capacity planning, and waste heat utilization have been stable for years. Figure 31 shows the total number of publications on sustainable cloud computing. This figure clearly shows that research on sustainable cloud computing is growing exponentially.

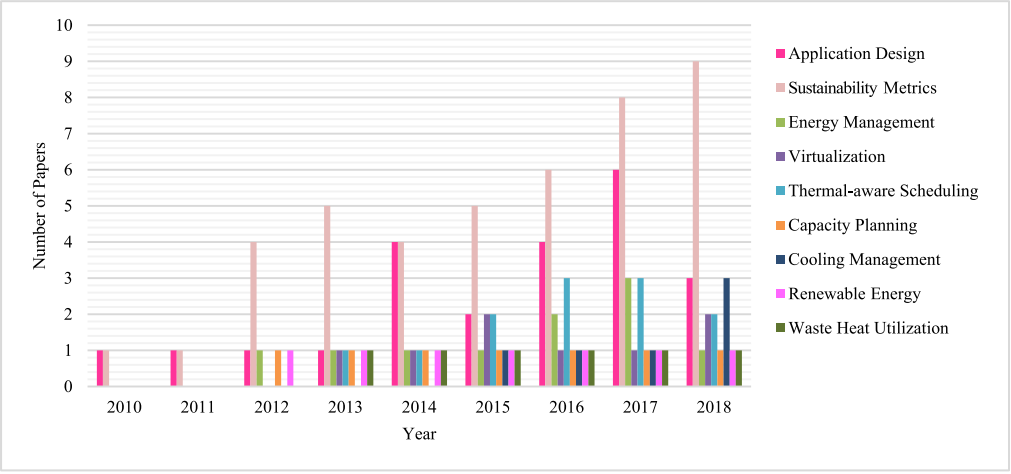


Fig. 30. Different categories of sustainable cloud computing.

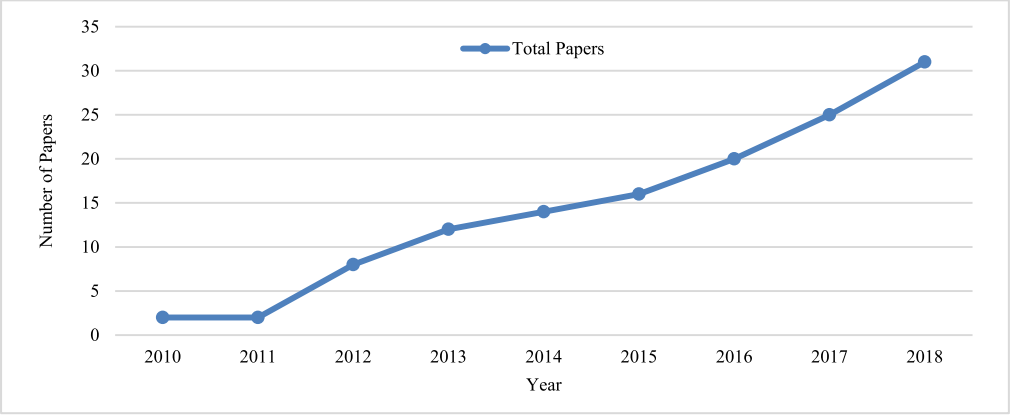


Fig. 31. Publications of sustainable cloud computing by year.

Figure 32 shows the number of citations discussing different categories of sustainable cloud computing from 2010 to 2018. From 2010 to 2011, the application design category was the hotspot. From 2017 to 2018, energy management, cooling management, thermal-aware scheduling, and

renewable energy were important research areas. Figure 33 shows that 28% of the research work appeared in conferences, 32% of the literature was published in journals, 21% of the studies appeared in IEEE Transactions, 10% of the literature was published in book chapters, 4% of the studies appeared in workshops, 3% of the literature was published in symposiums and 2% of the studies were published from various conferences. The largest amounts of publications came from journals (91 papers) followed by conferences (50 papers). Literature reported that there are seven different types of studies (introductory, review and survey, conceptual model, simulation, real testbed, and PhD thesis) in sustainable cloud computing. Figure 34 depicts that most of the research work (59%) has been on simulation-based environments. Figure 35 and Figure 36 depict the percentage of research articles focusing on various QoS parameters. We have identified different QoS parameters for the perspective of the user as well as cloud providers from 2010 to 2018, respectively.

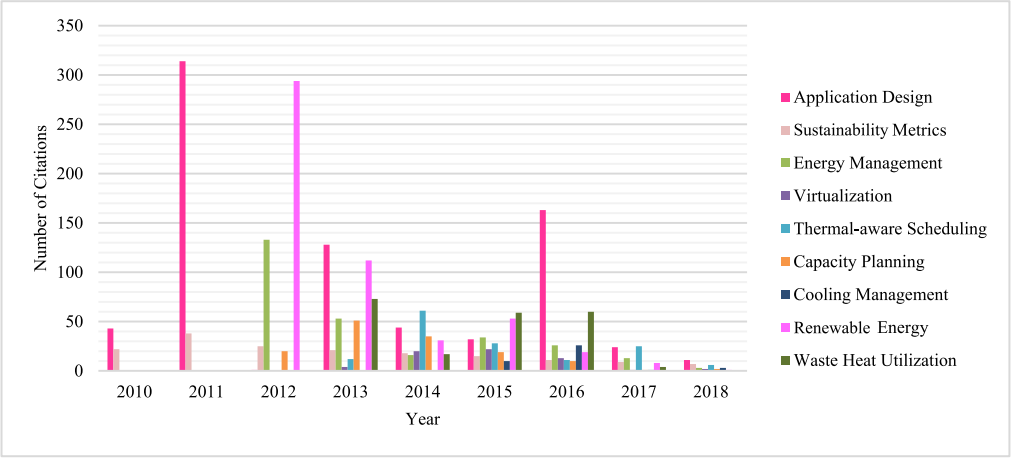


Fig. 32. Number of citations of different categories.

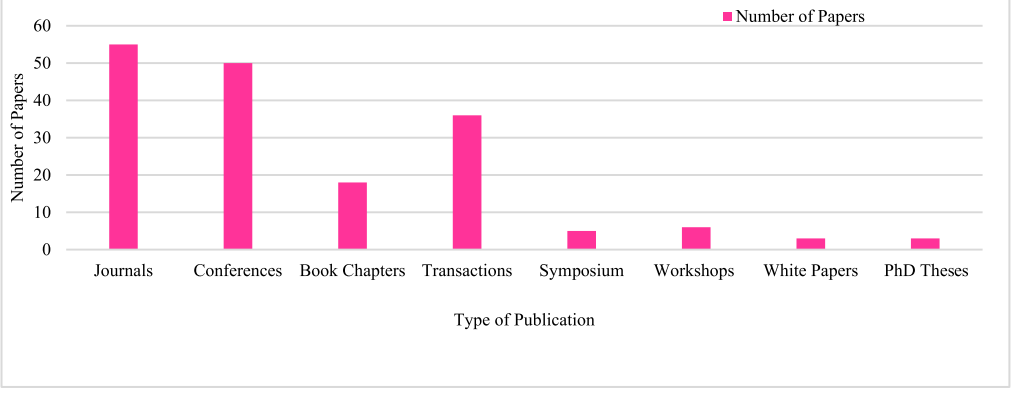


Fig. 33. Type of publications of sustainable cloud computing.

Response time (60%) is the most important QoS parameter for users while energy (40%) is the most important QoS parameter for cloud providers. We investigated a lack of research in security and reliability as a QoS parameter. We described a large number of research articles about cost, makespan, throughput, and energy efficiency as QoS parameters.

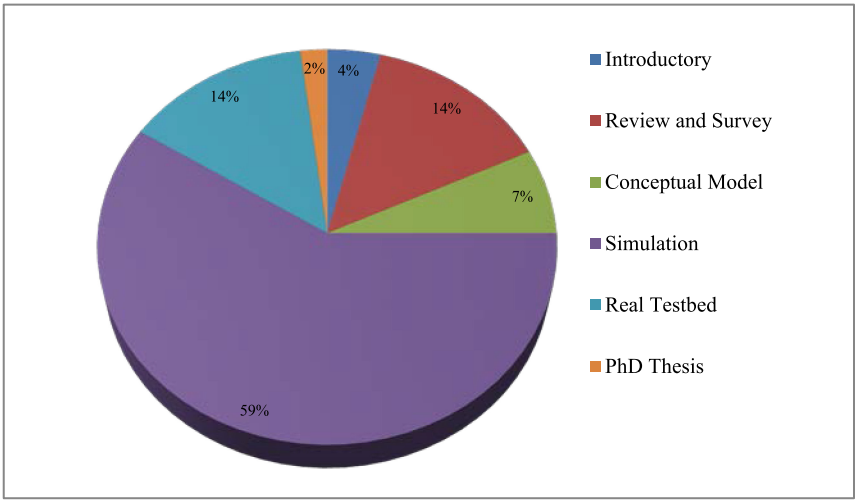


Fig. 34. Type of study of sustainable cloud computing.

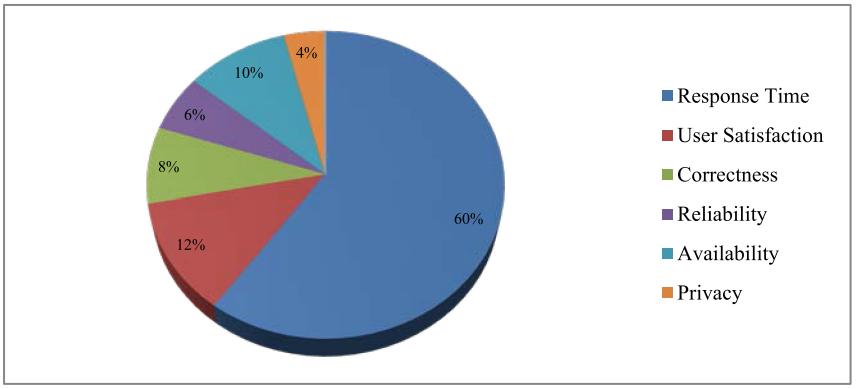


Fig. 35. QoS parameters for user's perspective.

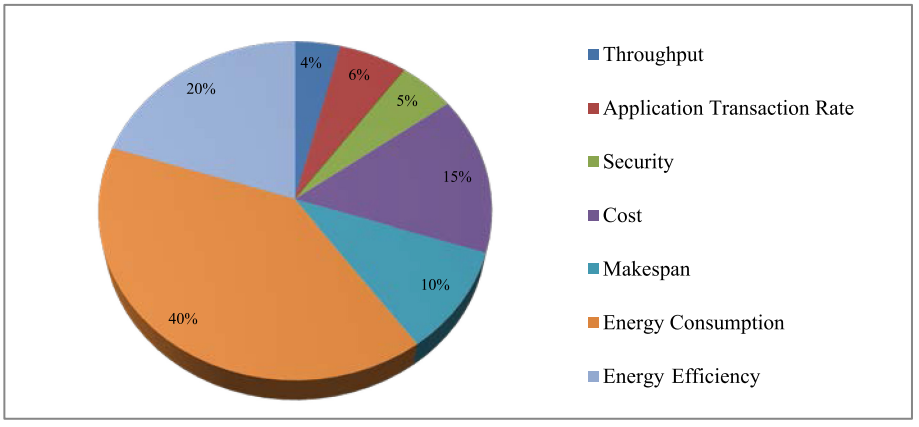


Fig. 36. QoS parameters for cloud provider's perspective.

E OPEN CHALLENGES AND FUTURE DIRECTIONS: A SUMMARY

We documented the research issues addressed and open challenges that are still unresolved in sustainable cloud computing.

E.1 Open Challenges

Though a lot of progress has been made in sustainable cloud computing, there are still many issues and challenges in this field that need to be addressed. Based on existing research, we have identified various open issues still pending in this area, described in Table 27.

Table 27. Open Challenges for Different Categories

Category	Open Challenges
Application Design	1 There is a need for effective communication among components of applications so that applications can scale easily during underloading or overloading of servers using a dynamic topology to add or remove resources automatically. 2 How does one add additional capacity to the application at runtime to scale individual services by avoiding the contention issues? 3 Which operations of the application could be handled asynchronously for effective load balancing at peak times? 4 What is the trade-off between scalability and availability during execution of an application? 5 How do we control access to application databases from other services running in parallel?
Sustainability Metrics	1 How does one measure the efficiency of a CDC by integrating different metrics for a particular context? 2 How does one determine the age and location of a CDC for overall comparison? 3 What are the acceptable levels of performance for every metric? 4 There is a need to define new metrics that can measure the security of a CDC directly because existing security metrics depend on compliance standards, SLA, and governance. 5 How does one measure the effect of IT load of the CDC on PUE significantly? 6 Existing metrics need seasonal benchmarking to capture region and change of the season to measure performance accurately. 7 How does one measure the energy consumption at subcomponent or lower level of a CDC? 8 How does one measure the cost of a CDC based on space and energy used by the CDC?
Capacity Planning	1 What application parameters are considered to merge different applications? 2 What is the trade-off between capacity cost and resource utilization during merging of applications? 3 How can migration of workloads or VMs affect power infrastructure capacity? 4 How does one plan effective capacity to perform data recovery in disaster management? 5 What are the most important user requirements that affect the capacity of a CDC during its technical design?
Energy Management	1 What is the trade-off between availability of an application and energy efficiency? 2 How does power consumption affect user satisfaction in terms of the SLA? 3 How does one reduce the SLA violation rate while transferring resources from high-scaling mode to low-scaling mode? 4 How does the size of the CDC affect its energy efficiency? 5 How is ICT mainly responsible for a large consumption of energy and carbon footprint generation? 6 How does memory contention affect the energy consumption of CDCs? 7 How can one improve energy efficiency based on traffic demands and QoS parameters? 8 How can on-demand switching affect energy consumption during starvation?

(Continued)

Table 27. Continued

Category	Open Challenges
Virtualization	1 What is the impact of VM consolidation and migration on preemption policy? 2 How does one track dynamic load variations during VM load balancing? 3 What is the trade-off between energy utilization and network delay during VM migration? 4 Increasing the size of a VM consumes more energy, which can increase service delay. 5 WAN-based VM migration requires storage migration, which can be overhead for cost-effective migration. 6 Security during VM migration is an important issue because a VM state can be hijacked during its migration. 7 How can one achieve VM migration and fault tolerance together in an efficient way?
Thermal-Aware Scheduling	1 There is a need for effective thermal-aware scheduling techniques that can execute workloads with minimum heat concentration and dissipation. 2 The complexity of scheduling and monitoring has increased due to temperature variations of CDC servers, which also causes vagueness in thermal profiling. To solve this problem, there is a need for dynamically updated thermal profiles instead of static profiles, which will be updated automatically and provide more accurate temperature values. 3 What is the trade-off between TCO and PUE? 4 If scheduling is performed based on different thermal aspects, such as inlet temperature and heat contribution, then admission control mechanisms at the processor level and server level contradict each other.
Cooling Management	1 How can one improve datacenter cooling efficiency without affecting the temperature of the CDC? 2 The consumption of large amounts of energy can challenge the cooling management system of CDCs but reductions in energy consumption can also affect the PUE of CDCs. 3 What is the trade-off between cooling efficiency and PUE? 4 There is a need to change the location of CDCs to reduce cooling costs. This can be done through placing the CDCs in areas that have availability of free cooling resources.
Renewable Energy	1 The main challenges of renewable energy are unpredictability and capital cost of green resources. 2 Mostly, sites of commercial CDCs are located away from abundant renewable energy resources. Consequently, movable CDCs are required to place them nearer to renewable energy sources to make them more cost-effective. 3 Adoption of renewable energy in CDCs is a research challenge of high capital cost. 4 The issue of unpredictability in supply of renewable energy and demand of CDCs must be addressed effectively. 5 The main reason for a large value of CUE is the total dependency on grid electricity, which is generated using fossil fuels. Cooling measures and other computing devices also have an impact on the value of PUE.
Waste Heat Utilization	1 The most important issues of waste heat utilization techniques are high capital cost and low heat quality. 2 The shifting of cooling systems from air based to water based creates new challenges: a. low quality of air due to mixing of exhaust air with cold air supply, b. smaller flow path in water-based cooling systems, and c. leakage of water into electronic equipment, which increases cost.

Note: The open research issues of different techniques have been discussed in their corresponding categories, as described in *Appendix C*.

E.2 Implications for Research and Practice

This systematic review has suggestions for prospective research scholars and practitioners who are already working in the area of sustainable cloud computing and looking for new ideas. A lot of research challenges are described for prospective researchers and professional experts. There is a need to integrate the different categories of sustainable cloud computing for better management of open issues related to CDCs. Figure 37 describes the interactions among different categories of sustainable cloud computing. We have identified two different types of interaction among different categories: weak interaction (if any category depends on the other category indirectly, then it is considered to be weak) and strong interaction (if one category depends on the other category directly, then it is considered to be strong). Based on the existing research work and their citations, we have identified the importance of an individual category along with its subcategories. Further, future hotspot areas have been identified among different categories. The 360-Degree View (global and complete view) of the taxonomy of sustainable cloud computing is provided in Appendix G.

E.3 Integrated: Sustainability vs. Reliability

Sustainable cloud services are attracting more cloud customers and making it more profitable. Improving energy utilization reduces electricity bills and operational costs, enabling sustainable cloud computing [14, 15]. On the other hand, to provide reliable cloud services, the business operations of cloud providers such as Microsoft, Google, and Amazon are replicating services, which need additional resources and increases energy consumption. To overcome this impact, a trade-off between energy consumption and reliability is required to provide cost-efficient cloud services. Existing energy-efficient resource management techniques consume a huge amount of energy while executing workloads, which decreases resources leased from CDCs [33]. DVFS-based energy management techniques have reduced energy consumption, but response time and service delays have increased due to the switching of resources between high-scaling and low-scaling modes [1]. Reliability of the system component is also affected by excessive turning on/off servers. Power modulation decreases the reliability of server components such as storage devices and memory. By reducing energy consumption of CDCs, we can improve the resource utilization, reliability, and performance of the server. Therefore, new energy-aware resource management techniques are needed to reduce power consumption without affecting the reliability of cloud services.

E.4 Emerging Trends and their Impact

Emerging trends can be observed in three different contexts: (1) applications (e.g., big data), (2) technologies (e.g., Software-Defined Network), and (3) techniques (e.g., deep learning). They either present a significant business opportunity for cloud service providers (to offer new SaaS using clouds by supporting emerging applications) in many interesting ways or help them in efficient management and utilization of their cloud infrastructures in a reliable and energy-efficient manner. With the enormous success of cloud computing, there are many emerging applications, such as big data, and smart cities [100] are harnessing it. These applications can be created using existing application/programming models such as those discussed in Section 3.1 (Application Design) or new ones to support the rapid creation of applications or ease of programming. When new application design/programming models are proposed, new approaches are needed for resource management and application scheduling to ensure the delivery of services of these cloud applications in an energy-efficient manner. All of these factors will impact cloud software platforms [3, 91].

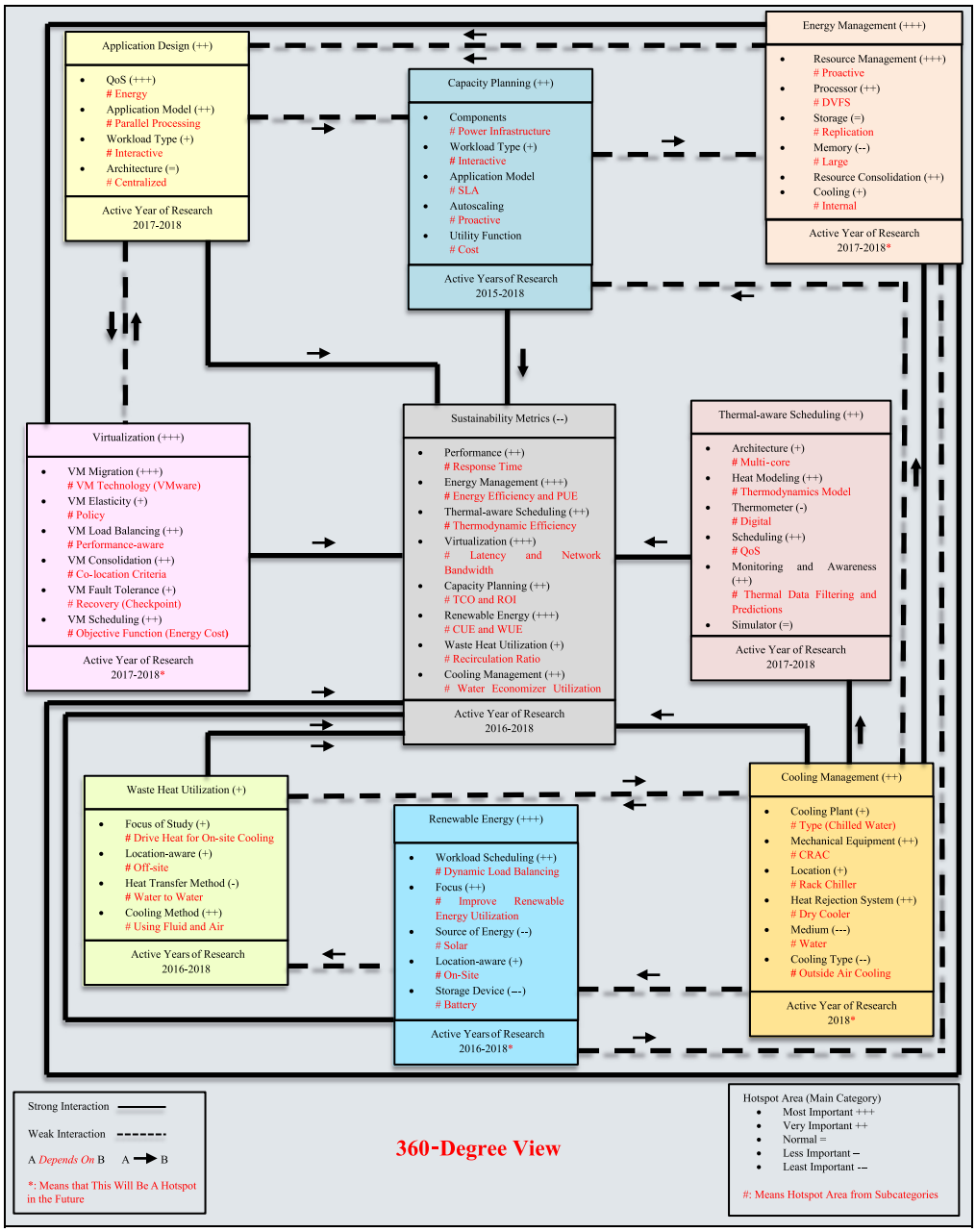


Fig. 37. Interactions among different categories of sustainable cloud computing (360-degree view).

Furthermore, new technological trends [161] such as Software-Defined Networks [165], the IoT, and Containers are presenting new opportunities for better management of datacenters to support execution of emerging applications in an energy-efficient manner. For example, IoT-based sensors and actuators help achieve better management of cooling systems of CDC infrastructures. The success of cloud computing is also leading to the creation of larger CDC infrastructures, which

is contributing to growing complexities in the management of CDCs. To better manage their resources and monitor application execution and resource status, works such as [160] are starting to use deep-learning approaches for detecting anomalies. Such deep-learning techniques need to be developed for better management of resources and workloads of emerging application models in a reliable and energy-efficient manner without affecting the quality of service delivered to users.

F SUSTAINABLE CLOUD COMPUTING ARCHITECTURE: A CONCEPTUAL MODEL

To resolve these challenges, cloud computing architectures is needed that can provide sustainable and reliable cloud services. Earlier models by Arlitt et al. [19] and Guitart [87] have been highly innovative, but as the research has persistently grown in the field of sustainable cloud computing, a new conceptual model to cover other important aspects of sustainability is needed. A model proposed by Arlitt et al. [19] is focused only on cooling management of CDCs and energy storage management, while a model proposed by Guitart [87] is focused on management of cooling, infrastructure, and workloads. Therefore, our model augments the previous aspects and covers all of the important components: (i) layered architecture, which comprises software (application management), platform (workload and VM/Resource management), and infrastructure management; (ii) cooling management; (iii) energy management (renewable and grid energy); and (iv) thermal-aware aspects of CDCs and describes the interactions among them. Figure 38 shows the conceptual model for sustainable cloud computing in the form of layered architecture, which offers holistic management of cloud computing resources to make cloud services more energy efficient, sustainable, and reliable.

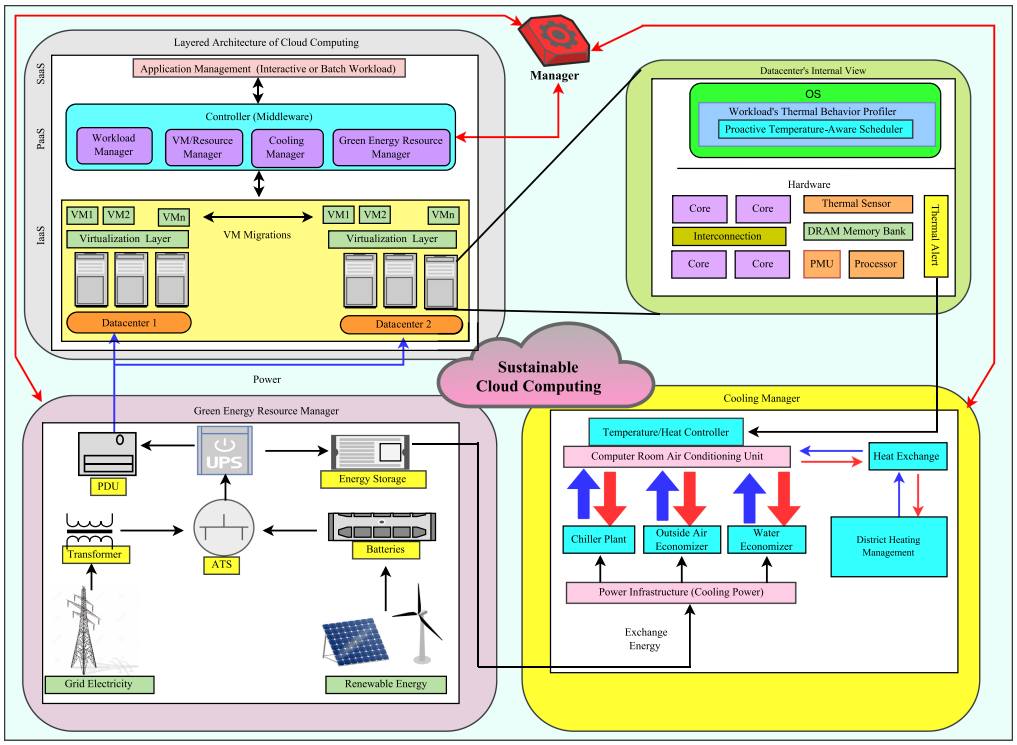


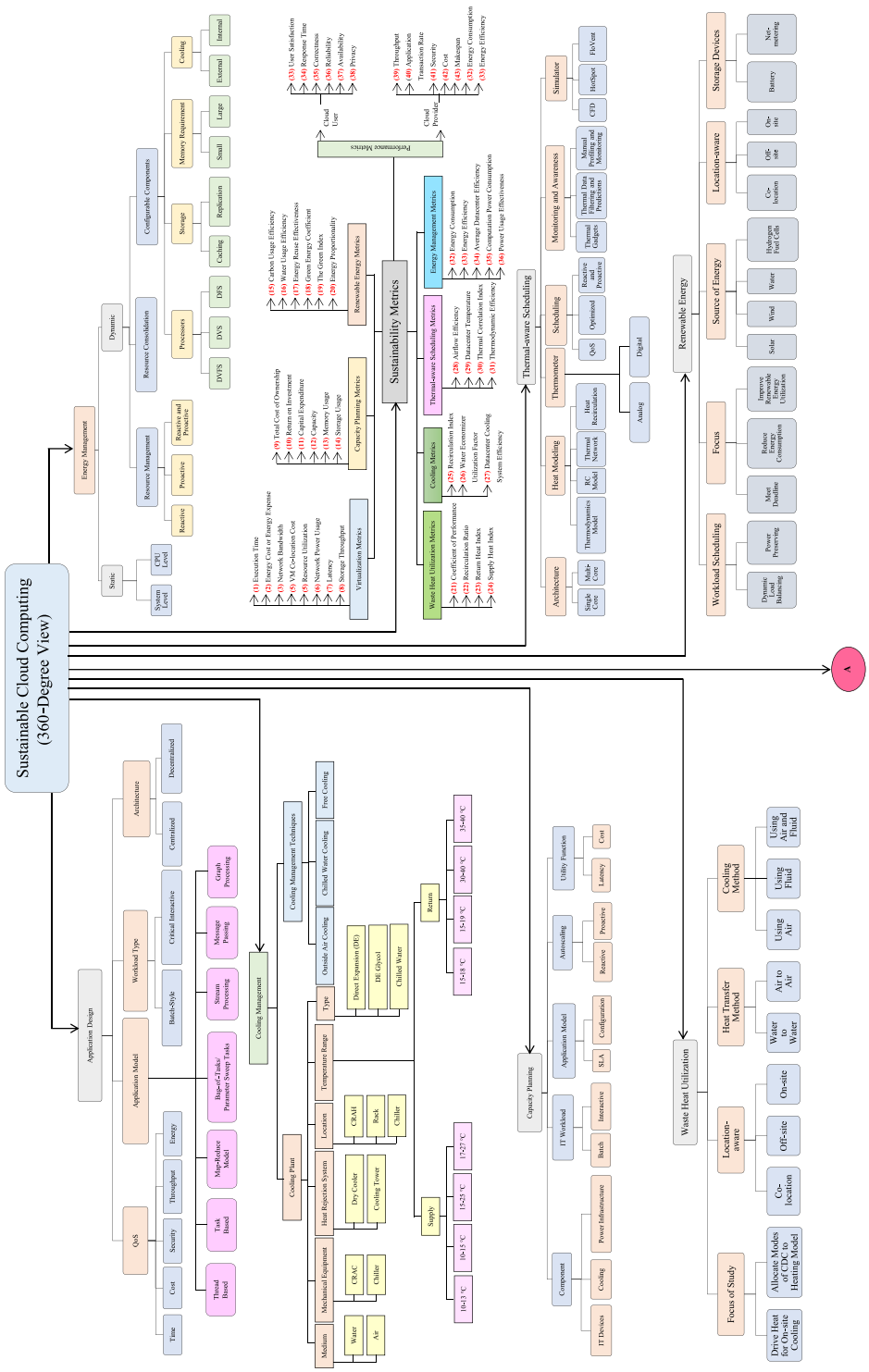
Fig. 38. Conceptual model for sustainable cloud computing.

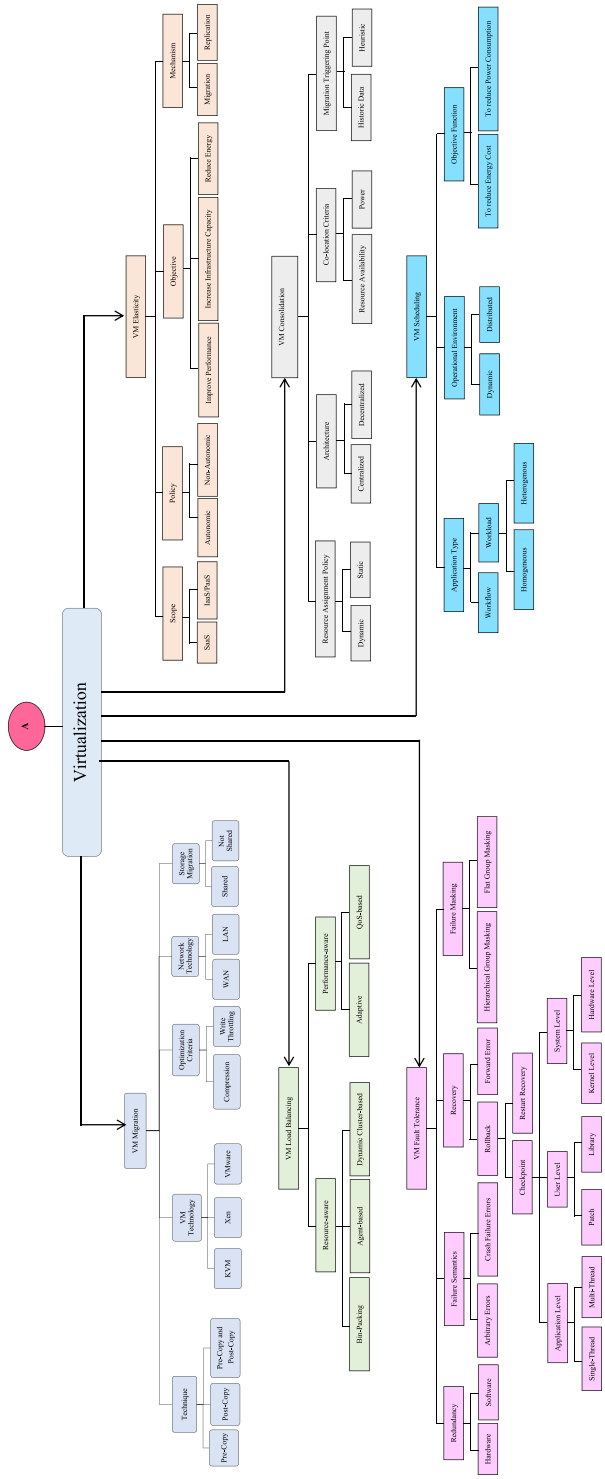
The three main components of the proposed architecture are as follows:

1. *Software as a Service (SaaS):* At this layer, the application manager is deployed to handle the incoming user workloads, which can be interactive or batch style, and transfer them to the workload manager for further action. At SaaS level, the QoS requirements of different applications can be defined in terms of the SLA. QoS requirements for an application can be deadline, budget, and the like. For example, different encoding techniques can be used for a video on-demand application based on the QoS requirements of the cloud users.
2. *Platform as a Service (PaaS)*: At this layer, the controller or middleware is deployed to control the important aspects of the system. The resource manager and scheduler follow different provisioning and scheduling policies for efficient management of cloud resources, which can improve energy efficiency and resource utilization to make CDCs more sustainable. There are three subunits of controller: workload manager, VM/resource manager, and manager. Their functions are described below:
   a) *Workload manager* manages the incoming workloads from the application manager and identifies the QoS requirement for every workload for successful execution and transfers the QoS information of the workload to the VM/resource manager.
   b) *VM/resource manager* provisions and schedules the cloud resources for workload execution based on the QoS requirements of the workload using physical machines or VMs.
   c) *Manager* controls the two modules of infrastructure: *Green energy resource manager* and *cooling manager*.
3. *Infrastructure as a Service (IaaS):* This layer contains the information about CDCs and VMs. VM migrations are performed to balance the load at the virtualization layer for efficient execution of workloads. The proactive temperature-aware scheduler is used to monitor the temperature variation of different VMs running at different cores. The Power Management Unit (PMU) is integrated to power all hardware executing the VMs. Dynamic Random-Access Memory (DRAM) stores the current states of VMs. Thermal profiling and monitoring techniques are used to analyze the temperature variations of CDCs based on the value of temperature as monitored by thermal sensors. Thermal alerts will be generated if the temperature is higher than the threshold value and the heat controller will take action to control the temperature without affecting the performance of the CDC. The electricity coming from Uninterruptible Power Supply (UPS) is used to run the cooling devices to control the temperature. District heating management is integrated, in which the temperature is controlled by using a chiller plant, outside air economizer, and water economizer. The *green energy resource manager* controls the power generated from renewable-energy resources and fossil fuels. To enable sustainable cloud environments, renewable energy is preferred over grid energy. If there is execution of deadline-oriented workloads, then grid energy can be used to maintain the reliability of cloud services. The sources of renewable energy are solar and wind power. Batteries are used to store the renewable energy. Automatic Transfer Switch (ATS) is used to manage the energy coming from both sources (renewable and non-renewable) and forward to UPS. Further, a Power Distribution Unit (PDU) is used to transfer the electricity to all the CDCs and cooling devices.

G 360-DEGREE VIEW OF TAXONOMY

The 360-Degree View (global and complete view) of the taxonomy of sustainable cloud computing is shown below.


Sustainable Cloud Computing (360-Degree View)
Application Design
Application Model
QoS
Time
Cost
Security
Throughput
Energy
Workload Type
Batch-Style
Critical Interactive
Architecture
Centralized
Decentralized
Thread Based
Task Based
Bags-of-Tasks/ Parameter Sweep Jobs
Map-Reduce Model
Message Passing
Stream Processing
Graph Processing
Cooling Management
Cooling Management Techniques
Free Cooling
Chilled Water Cooling
Outside Air Cooling
Chilled Water
Direct Expansion (DX)
DE Glycol
Type
Cooling Plant
Heat Rejection System
Mechanical Equipment
Medium
Water
Air
Chiller
CRAC
Dry Cooler
Cooling Tower
Chiller
CRAH
Rack
Chiller
Location
Temperature Range
Supply
Return
10-15° C
15-20° C
15-25° C
17-27° C
15-18 °C
18-24 °C
24-30 °C
35-40 °C
Capacity Planning
IT Workload
Batch
Interactive
Application Model
SLA
Configuration
Autoscaling
Reactive
Proactive
Utility Function
Latency
Cost
Component
IT Devices
Cooling
Power Infrastructure
Energy Management
Static
System Level
CPU Level
Dynamic
Resource Consolidation
Resource Management
Reactive
Proactive
Reactive and Proactive
Processors
DVFS
DVS
DFS
Caching
Storage
Replication
Configurable Components
Memory Requirement
Small
Large
Cooling
External
Internal
Sustainability Metrics
(1) Execution Time
(2) Energy Cost or Energy Expense
(3) Network Bandwidth
(4) VM Consolidation Cost
(5) Resource Utilization
(6) Network Power Usage
(7) Latency
(8) Storage Throughput
Virtualization Metrics
(9) Total Cost of Ownership
(10) Return on Investment
(11) Capital Expenditure
(12) Capacity
(13) Memory Usage
(14) Storage Usage
Capacity Planning Metrics
(15) Carbon Usage Efficiency
(16) Water Usage Efficiency
(17) Energy Reuse Effectiveness
(18) Green Energy Coefficient
(19) The Green Index
(20) Energy Proportionality
Renewable Energy Metrics
Waste Heat Utilization Metrics
(21) Coefficient of Performance
(22) Recirculation Ratio
(23) Return Heat Index
(24) Supply Heat Index
Cooling Metrics
(25) Recirculation Index
(26) Water Economizer
(27) Thermal Energy Cooling System Efficiency
Thermal-aware Scheduling Metrics
(28) Airflow Efficiency
(29) Datacenter Temperature
(30) Thermal Correlation Index
(31) Thermodynamic Efficiency
Energy Management Metrics
(32) Energy Consumption
(33) Energy Efficiency
(34) Average Datacenter Efficiency
(35) Computation Power Consumption
(36) Power Usage Effectiveness
Performance Metrics
Cloud User
(33) User Satisfaction
(34) Response Time
(35) Correctness
(36) Reliability
(37) Availability
(38) Privacy
Cloud Provider
(39) Throughput
(40) Application Transaction Rate
(41) Security
(42) Cost
(43) Makespan
(32) Energy Consumption
(33) Energy Efficiency
Thermal-aware Scheduling
Heat Modeling
Architecture
Single Core
Multi-Core
Thermodynamics Model
RC Model
Thermal Network
Heat Recirculation
Thermometer
Digital
Analog
Scheduling
QoS
Optimized
Reactive
Proactive
Monitoring and Awareness
Thermal Gadgets
Thermal Data Retrieval and Prediction
Manual Profiling and Monitoring
Simulator
CFD
HotSpot
HaVen
Renewable Energy
Workload Scheduling
Dynamic Load Balancing
Power Preserving
Meet Deadline
Focus
Reduce Energy Consumption
Power Renewable Energy Utilization
Source of Energy
Solar
Wind
Water
Hydrogen Fuel Cells
Storage Devices
Battery
Net-metering
Location-aware
Co-location
Off-site
On-site
A
Waste Heat Utilization
Focus of Study
Drive Heat Reuse Outside Cooling
Allocate Modes of CDCs
Heating Model
Location-aware
Co-location
Off-site
On-site
Heat Transfer Method
Water to Water
Air to Air
Cooling Method
Using Air
Using Fluid
Using Air and Fluid


A
Virtualization
VM Migration
Technique
Pre-Copy
Post-Copy
Pre-Copy and Post-Copy
Hardware
Software
VM Technology
KVM
Xen
VMware
Optimization Used
Compression
Write Throttling
Network Technology
WAN
LAN
Storage Migration
Shared
Not Shared
VM Load Balancing
Bin-Packing
Agent-based
Dynamic Cluster-based
Resource-aware
Performance-aware
Adaptive
QoS-Based
VM Fault Tolerance
Redundancy
Hardware
Software
Failure Semantics
Arbitrary Errors
Crash Failure Errors
Recovery
Rollback
Forward Error
Checkpoint
Restart Recovery
Failure Masking
Hierarchical Group Masking
Flat Group Masking
Application Level
Single-Thread
Multi-Thread
User Level
Patch
Library
System Level
Kernel Level
Hardware Level
VM Elasticity
Scope
Static
Dynamic
Policy
Autonomic
Non-Autonomic
Objective
Improve Performance
Increase Infrastructure Capacity
Reduce Energy
Mechanism
Migration
Replication
VM Consolidation
Resource Assignment Policy
Static
Dynamic
Architecture
Centralized
Decentralized
Co-location Criteria
Resource Availability
Power
Migration Triggering Point
History Data
Heuristic
VM Scheduling
Application Type
Workflow
Workload
Homogeneous
Heterogeneous
Operational Environment
Dynamic
Distributed
Objective Function
To reduce Energy Cost
To reduce Power Consumption