Computing (2016) 98:751–774
DOI 10.1007/s00607-014-0407-8

A survey and taxonomy on energy efficient resource allocation techniques for cloud computing systems

Abdul Hameed · Alireza Khoshkbarforoushha · Rajiv Ranjan · Prem Prakash Jayaraman · Joanna Kolodziej · Pavan Balaji · Sherali Zeadally · Qutaibah Marwan Malluhi · Nikos Tziritas · Abhinav Vishnu · Samee U. Khan · Albert Zomaya

Received: 23 June 2013 / Accepted: 29 April 2014 / Published online: 6 June 2014

Abstract In a cloud computing paradigm, energy efficient allocation of different virtualized ICT resources (servers, storage disks, and networks, and the like) is a complex problem due to the presence of heterogeneous application (e.g., content delivery networks, MapReduce, web applications, and the like) workloads having contentious

A. Hameed · S. U. Khan
North Dakota State University, Fargo, USA

A. Khoshkbarforoushha
Australian National University, Canberra, Australia

R. Ranjan (✉) · P. P. Jayaraman
CSIRO, Canberra, Australia
e-mail: rajiv.ranjan@csiro.au

J. Kolodziej
Cracow University of Technology, Kraków, Poland

P. Balaji
Argonne National Laboratory, Lemont, USA

S. Zeadally
University of the District of Columbia, Washington, USA

Q. M. Malluhi
Qatar University, Doha, Qatar

N. Tziritas
Chinese Academy of Sciences, Beijing, China

A. Vishnu
Pacific Northwest National Laboratory, Richland, USA

A. Zomaya
University of Sydney, Sydney, Australia

allocation requirements in terms of ICT resource capacities (e.g., network bandwidth, processing speed, response time, etc.). Several recent papers have tried to address the issue of improving energy efficiency in allocating cloud resources to applications with varying degree of success. However, to the best of our knowledge there is no published literature on this subject that clearly articulates the research problem and provides research taxonomy for succinct classification of existing techniques. Hence, the main aim of this paper is to identify open challenges associated with energy efficient resource allocation. In this regard, the study, first, outlines the problem and existing hardware and software-based techniques available for this purpose. Furthermore, available techniques already presented in the literature are summarized based on the energy-efficient research dimension taxonomy. The advantages and disadvantages of the existing techniques are comprehensively analyzed against the proposed research dimension taxonomy namely: resource adaption policy, objective function, allocation method, allocation operation, and interoperability.

Keywords Cloud computing · Energy efficiency · Energy efficient resource allocation · Energy consumption · Power management

Mathematics Subject Classification 68U01

1 Introduction

The most comprehensive, widely used and referred definition of cloud computing in the literature is presented in [1] that defines cloud computing as "A model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction". Cloud computing is a new computing model where a myriad of virtualized ICT resources are exposed as web utilities, which can be invoked and released in an on-demand fashion by application programs over the Internet [2–4]. The concept of cloud computing is an immediate extension of many well researched domains such as virtualization, distributed, utility, cluster, and grid computing. Cloud computing datacenters employ virtualization technologies that allow scheduling of workloads on smaller number of servers that may be kept better utilized, as different workloads may have different resource utilization footprints and may further differ in their temporal variations.

In the cloud computing paradigm, all ICT resources are "virtualized" as datacenter facilities and are operated by third party providers [5–7]. Multiple commercial clouds already alleviate different businesses from the burden of management and maintenance of different resources, and allow businesses to supplement their assets [8]. More and more companies are offering cloud computing services as evident by the development and expansion of commercial cloud infrastructures, such as Amazon, Microsoft, Gogrid, Flexiant, Layered Technologies, vCloud Express and ENKI Prima Cloud. From Google's point of view [9], the five key characteristics of cloud computing are task centric, user centric, intelligence, powerfulness, and programmability.

Moreover, cloud computing is based on a pay-as-you-go model, where the end-users pay only for the number and type of services they purchase [10]. In cloud

computing environments, increasing ICT resource capacity or removing an existing resource capacity, etc. can be done via invocation of an SOAP/Restful API. Cloud computing amortizes the ownership cost over distributed servers, shared system operators, and diversity of workloads and offers different services such as computation, backup, data access, software and hardware services to end-users. The cloud providers charge end-users based on service level agreements (SLAs) that account for the usage or reservation of data center resources. The cloud computing infrastructure has several important and unique key issues such as meeting performance constraints under uncertainties, dynamic scalability, standardization, fault-tolerance, debugging, reducing operational costs, *reducing carbon emission* (*focus of this paper*) [11] and ensuring security and privacy of hosted ICT resources and application data [12].

Reducing carbon emission by cloud computing datacenters has emerged as one the dominant research topics both in industry and academia. This is due to the fact that the energy required by the datacenters for its operation, illumination, power supply, and cooling, contribute significantly to the total operational costs [13]. Therefore, reducing the power consumption and energy dissipation had become important concerns for making cloud services environmentally sustainable [14].

The growing transition to datacenter cloud computing systems is fuelling an increasing concern about the growing demand for electricity and related carbon emissions that will be caused by vast datacenter that are being constructed at a fast pace. In 2010, electricity usage in global datacenter accounted for about 1.3 % of total electricity usage worldwide. In the U.S. alone this figure is about 2.0 %. According to a McKinsey report, the total estimated electricity bill for datacenter in 2010 was $11.5 billion. Energy costs in a typical datacentre doubles every 5 years. (Figure 1 shows the carbon emission from all datacenter worldwide estimated by Stanford University, McKinsey

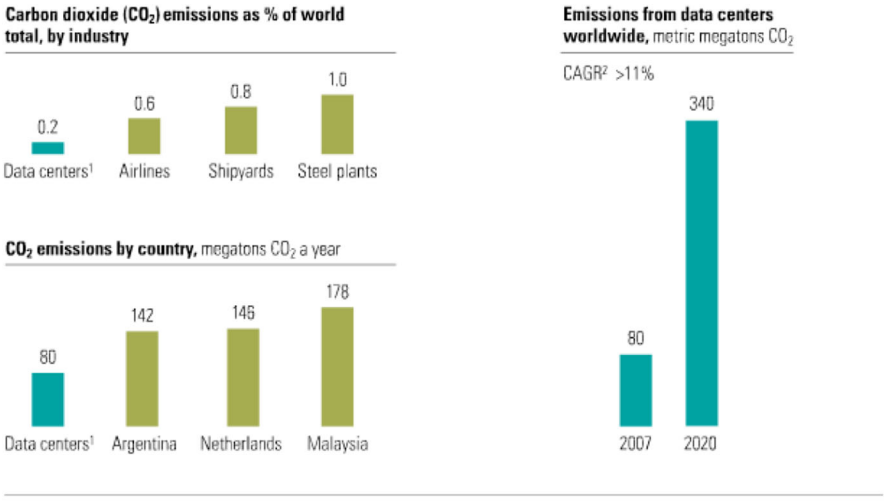

Fig. 1 Carbon footprint from datacenters worldwide

study and Gartner research). Without serious efforts to curb electricity demand and related emissions, current projections (see Fig. 1) show that by 2020 worldwide carbon emission from datacenter will quadruple (Commutative Average Growth Rate-CAGR >11 %). These figures are expected to rise with our growing reliance on datacenter.

One of the major causes of energy inefficiency in datacenters is the wastage of idle power when ICT resources such as servers providing computing and storage capacities run at low utilization. For instance, even at a very low server load of 10 % CPU utilization, the server consumes over 50 % of the peak power [15]. Similarly, if the disk, network, or any such ICT resource becomes the performance bottleneck, other ICT resources will consequently become idle and waste a lot of energy. Energy-efficient management of ICT resources in datacenters for cloud computing systems is an important concern for several reasons [16]. *First*, the electricity costs for powering ICT resources and cooling systems are starting to overtake the actual cost of purchasing the ICT hardware. *Second*, increased datacenter energy usage and related carbon emissions has prompted environmental concerns (leading governments across the world are now seeking to regulate datacenter power usage). *Finally*, increased energy usage and heat dissipation has negative impacts for density, reliability, and scalability of datacenter hardware.

We define the energy efficient resource allocation problem as "the problem of selecting, deploying, and run-time management of datacenter resources in a way that hosted application achieves their QoS constraints, while meeting providers' objectives—improved resource utilization with reduced financial and environmental costs". The foremost objective of cloud service providers is to have a cost effective and energy efficient solution for allocating virtualized ICT resources to end-users' application while meeting the QoS (Quality of Service) level as per SLA (Service Level Agreement). On the other hand, end-users are interested in minimizing the cloud investment while meeting QoS constraints such as application response time, availability, and throughput. Hence, the cloud datacenter service providers are eager to optimize the efficiency of resource allocation because the requested resources are handled from a shared pool of resources. By improving energy efficiency of datacenter, the service providers can reduce the size and cost of the energy sources needed to power and cool the ICT resources [17]. In the literature there are four well known approaches to design of an energy efficient cloud computing datacenters: (a) reducing the energy dissipated in an infrastructure through energy-efficient resource allocation and management of datacenters, (b) ensuring permanence of infrastructure to reduce the need for equipment replacement e.g., circumvent server breakdown by preserving safe operation temperature, (c) increased equipment utilization as computational load is geographically distributed to cater for the needs of end-users, and (d) minimizing self-management and flexibility as cost is spread across a number of datacenter [18].

To be more specific, some of the recent research works have investigated the optimization of energy utilization by monitoring the performance virtualized ICT resources (server) and hosted workload under variable CPU frequency [19,20]. Some other approaches have investigated techniques such as processor speed control, voltage adjustments, switching off a display monitor, hibernate or sleep mode [21–23]. However, applying aforementioned techniques targeted single personal computing (PC) devices; hence do not completely solve the problem of energy inefficiency in data-

center. This is because the energy saved by scaling down the CPU voltage is far less than powering off a physical server. An energy-saving approach for cloud datacenter is different from that for a single PC. The cloud computing is a prototype shift from the outdated uniprocessor computation approach of development to that of an accessible, multi-tenant, and global infrastructure.

Many recent research works have proposed energy-aware resource allocation computing methods for distributed and cloud computing [24–28]. Several surveys of resource allocation of cloud computing have also been reported [29–33]. However, none of them focus on the core challenges of energy efficient resource allocation in clouds. Further, none of the previous works have clearly addressed the energy efficient resource management problem from application engineering perspective.

In this paper we make following important contributions: (i) we clearly identify the open research challenges associated with energy efficient allocation of ICT resources in cloud computing datacenter; (ii) we present a novel research taxonomy (resource adaption policy, objective function, allocation method, allocation operation, and interoperability) for classifying existing literature; and (iii) we apply the proposed taxonomy for critically surveying that existing literature and discussing their core advantages and disadvantages.

Rest of this paper is organized as follows: Sect. 2 articulates the key concepts concerning the issue of energy efficient resource allocation problem; Sect. 3 presents the taxonomy based on research dimensions for classifying the existing literature in the concerned research area; Sect. 4 classifies the existing state-of-the-art based on proposed research taxonomy; Sect. 5 presents the summary and our plan for future work; paper ends with some conclusive remarks in Sect. 6.

2 Key concepts

This section explores concepts related to energy efficient allocation of cloud computing resources that forms the basics of our research work.

2.1 Energy efficiency

The fossil fuels are one of the major sources of energy generation and use of fossil fuels produce harmful carbon emissions. The issue of energy consumption across the ICT infrastructure such as datacenter is important and has received wide recognition in ICT sector [34–36]. A new scalable design for efficient cloud computing infrastructure is required that can support the reduction in Greenhouse Gas (GHG) transmissions in general, and energy consumption in particular [35–39]. The increase in ICT resource number and density has direct impact on the user expenditure as well as cooling and power management of datacenter infrastructures [40,41]. In cloud computing, two popular schemes: (a) sleep scheduling and (b) resource virtualization had helped in improvement of energy efficiency and power consumption within the datacenter [42,43].

A number of datacenter provider companies are persistent and dependent on the use of "old dirty energy" equipment. GreenPeace [44] analyzed the datacenter investments of top cloud vendors including Amazon, Azure, and the like; and the results of

their research revealed number of interesting facts, which are explained next. From their report it was clear that most of the vendors do not take major steps towards clean and renewable electricity. Only a few vendors have started to pay some attention to observing the dirty energy and carbon footprints of their ageing ICT resources. However, precisely measuring the energy consumption and GHG emissions seems impossible due to lack of transparency between energy consumption by ICT equipment (resource) and environmental impact of equipment. The two percentage measures: (a) coal intensity and (b) clean energy index were computed and compared with the maximum power demand of a datacenter. The coal intensity is an approximate measure of electricity produced by coal for datacenter operations. The clean energy index is computed based on the measure of electricity demand size and renewable energy used to power the datacenter.

The datacenter providers are now starting to realize the relationship between energy consumed by their ICT resources and GHG emissions. The three areas where energy is most consumed within a datacenter includes: (a) critical computational server providing CPU and storage functionalities; (b) cooling systems; and (c) power conversion units. In the recent Google's Green Data Centers report [9], three best practices and five step approaches for cooling and reducing energy usage within a datacenter is proposed. The best practices include (i) measuring performances such as ICT equipment (resource) energy and facility energy overhead; (ii) optimizing air flow by prevention of mixing hot and cold air and elimination of hot spots; (iii) turning up the thermostat by elevated cold aisle and raising the temperature. A part from best practices, the five steps of reducing energy usage are: (a) identifying critical monitoring points, (b) efficient arrangement of vent tiles, (c) increase humidity and temperature settings, (d) isolation of UPS, and (e) improving CRAC unit controller.

2.2 What is resource allocation problem?

Resource allocation is one of the challenges of cloud computing because end-users can access resources from anywhere and at any time. The resources in a cloud cannot be requested directly but can be accessed through SOAP/Restful web APIs that map requests for computations or storage are mapped to virtualized ICT resources (servers, blob storage, elastic IP, etc.). Since, cloud datacenter offer abundance of resources, the cloud computing model is able to support on-demand elastic resource allocation. However, such abundance also leads to non-optimal resource allocation.

In cloud computing paradigm, the key challenge is the allocation of resources among end-users having changing requests of resources based on their application usage patterns. The unpredictable and changing requests need to run on datacenter resources across the Internet. The aim of resource allocation for any particular cloud provider can be either optimize applications' QoS or improve resource utilization and energy efficiency. The main objective is to optimize QoS parameters (response time) that measures the efficiency of resource allocation regardless of the type of ICT resources allocated to end-users. The optimized QoS parameters can be any measure such as time, space, budget, and communication delay. Some of the challenges associated with energy efficient resource allocation policies that we have identified include:

(1) Choosing workload type and interference between different workloads, such as resource usage, performance, and power consumption.
(2) Provisioning of resource allocation and utilization at run time by evaluating the possibility of centralized, federated, and standardized datacenter resources.
(3) Improving asset utilization, network accessibility, power efficiency, and reduction in the time needed to recover from any failure.
(4) Improving cloud resources, topology, tools, and technologies by evaluating and fine tuning the cloud infrastructure layout.
(5) Increasing performance and the return on investment by assessing application inter-dependencies to facilitate resource consolidation.
(6) Supporting business security and flexibility for mission-critical applications through practical cloud infrastructure planning.

The definition of resource is very important as anything, such as CPU, memory, storage, bandwidth, and application can be termed an ICT resource in cloud computing landscape. The important characteristic of a resource unit is abstracted by the cost of operation and infrastructure. The problem of resource allocation is quite complex and needs some assumptions including: (a) set of workflow tasks for resource requirements, (b) set of operational servers, (c) task consolidation meeting SLA, and (d) reduction in power wastage and resource usage costs. The resource allocation problem involves the appropriate provisioning and efficient utilization of available resources for applications to meet the QoS performance goals as per SLA. For cloud computing infrastructures, the service providers also need to track the changes in resource demands. Moreover, a cloud service provider allocates system resources to CPUs, and determines whether to accept incoming requests according to resource availability. The factors, such as: (a) monitoring the availability of system resources, (b) tracking QoS requirements, (c) monitoring users' service requests, (d) pricing the usage of resources, (e) tracking execution improvement of the service requests, and (f) maintaining the actual usage of resources make the resource allocation complex and complicated task. Moreover, the resource allocation problem is also challenging due to the time-varying workloads that cause different resource demands from service providers for the cloud computing paradigm.

3 Taxonomy of resource allocation techniques

As we mentioned earlier, there are a number of resource allocation techniques that has not yet surveyed with the focus on energy efficient resource management problem in datacenter clouds. To do so, this section, makes a comparison based on the following dimensions: resource adaption policy, objective function, allocation method, and allocation operation.

3.1 Resource allocation adaption policy

This dimension refers to the degree to which an energy-aware resource allocator is able to adapt to dynamic or uncertain conditions. The uncertainties arise from a number

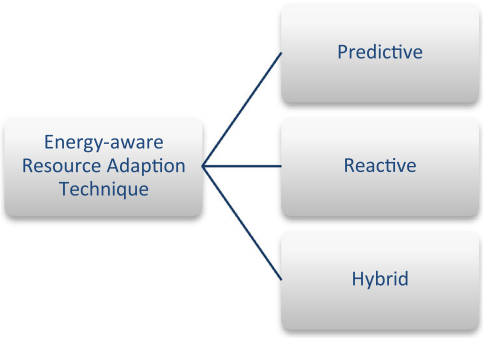

Fig. 2 Resource adaption policy taxonomy

of factors including resource capacity demand (e.g., bandwidth, memory, and storage space), failures (e.g., failure of a network link and failure of the CPU hosting application instance), and user workload pattern (e.g., number of users and location). In this paper, the resource adaption policy is classified into three categories: (a) predictive, (b) reactive, and (c) hybrid. Figure 2 depicts the pictorial representation of the proposed classification.

Monitoring the status of cloud-based hardware resources (e.g., virtual machine container or virtual server, storage, and network) and the software resources (e.g., web server, application server, database server, etc.) that make-up the application is integral to the functioning and implementation of the aforementioned resource adaptation policies. Monitoring activity [45] involves dynamically profiling the QoS parameters related to hardware and software resources, the physical resources they share, and the applications running on them or data hosted on them. Monitoring services can help resource allocator as regards to: (i) keeping the cloud resources and application operating at peak energy efficiency levels; (ii) detecting the variations in the energy efficiency of resources and QoS delivered by hosted applications; and (iii) tracking the failure of resources and applications.

Using past knowledge-driven machine learning techniques, predictive resource allocation policy can dynamically anticipate and capture the relationship between applications QoS targets, energy efficiency objective function, and current hardware resource allocation and user workload patterns in order to adjust the resource allocation. The past knowledge is derived from the monitoring service, which continuously profiles information in a searchable database (e.g., MySQL and NoSQL databases). The resource capacity planning is done at prior and allocations are approximated based on resource performance models and application workload models. Both of the aforementioned models leverage past monitoring knowledge for training machine learning techniques. The output of the machine learning techniques such as neural networks [46], genetic algorithms [47], reinforcement learning [48], etc. is the feedback to the resource allocator.

Workload prediction models forecast workload behavior across applications in terms of CPU, storage, I/O, and network bandwidth requirements. On the other hand, resource performance model predict the performance of CPU, storage, and network

resources based on their past performance history, and anticipated workload patterns. For example, the predictive technique proposed in [49] can tackle unprecedented sharp changes in application workloads. Predictive allocation handles the estimated base workload at coarse time scales (e.g., hours or days) maintaining long term workloads [50]. The predictive resource allocation suffers from limitation when there is no sufficient workload and resource performance data available to train the machine learning technique. For example, such a scenario can arise when deploying a new application on cloud resources, which does not have any past performance or workload history. Predictable approaches can also fail under situations when the workload and resource performance data do not have any specific distributions. This affects the accuracy of prediction. In addition, resource allocation predictions have been proven to be expensive in terms of storage cost (main memory processing) and processing time complexity [49].

Reactive techniques [51] rely on monitoring the state of cloud resources and triggering hard-coded, pre-configured corrective actions when some specific even occurs such as utilization of the CPU resource reaches certain threshold or energy consumption of a CPU resource goes beyond threshold. The efficiency of reactive allocation depends on the ability to detect fluctuations. Besides handling temporary changes of workload, the allocations are adjusted in a timely manner by minimizing the deviation from the QoS performance goals during and after the workload change [49]. Reactive resource allocation adapts to service requirements and resource availability, and optimize for long term resource allocation. The reactive approach changes allocation of resources in short intervals (e.g., every few minutes) in response to workload surges and is computationally attractive because no extensive knowledge of the application's resource and workload demands is needed. Reactive policies react quickly to changes in workload demand but have limited significance because the policies suffer from issues such as: (a) lack of predictability, (b) instability, and (c) high provisioning costs [50].

Pure reactive resource allocation delays workload and operates over time scale of a few minutes. Pure predictive resource allocation preserves long-term workload statistics besides envisaging and allocating for the next few hours. Therefore, hybrid resource allocation combines predictive with reactive allocation techniques and accomplishes substantial improvements in: (a) meeting SLAs, (b) conserving energy, and (c) reducing provisioning costs. In hybrid resource allocation approach, a predictive (reactive) allocation switches the underline workload (handles any excess demand) at granular (finer) time scales. A coordinated management among predictive and reactive approaches achieves a significant improvement in energy efficiency without sacrificing performance [52]. Hybrid allocation approaches outperforms predictive and reactive resource allocation strategies when performance, power consumption, and number of changes in resource allocation are considered [50].

3.2 Objective function

The objective function can be a mathematical expression, metric, or function that needs optimization with conditions subject to system constraints [50]. Energy optimization

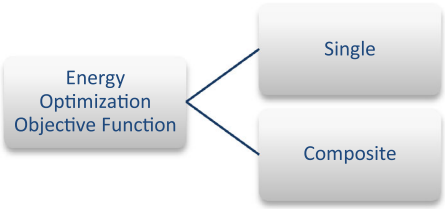

Fig. 3 Objective function taxonomy

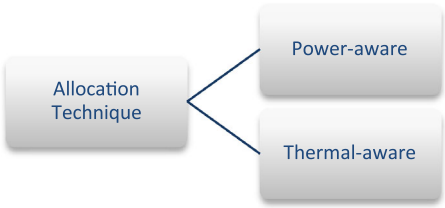

Fig. 4 Allocation method taxonomy

objective function, which is also referred as cost function and energy function in the literature, would be either single or composite respecting to the number of parameters considered for optimization. For instance, a cost function which aims at minimizing energy consumption is considered as single objective function; hence if it deals with minimizing both energy consumption and SLA violation, the cost model would be composite. Figure 3 represents the taxonomy of objective function considered for classification of different approaches present in literature.

In cloud computing, for the increasing cost and shortage of power, an objective function is the measure of increase power usage for a resource allocation. Moreover, the objective function changes with the implementation of the algorithm for specific system architecture under specified user requirements [53].

3.3 Allocation method

The energy conservation in hosting centers and server farms is of increasing importance. In the paper, we have broadly classified the allocation method in power-aware and thermal-aware allocation methods. Figure 4 presents the allocation method taxonomy considered for the study.

3.3.1 Power-aware

In the context of cloud computing, power-aware design techniques aim to maximize service-level performance measures (e.g. SLA violation rate, SLA accomplishment rate, waiting time, etc.) under power dissipation and power consumption constraints [54,55]. Moreover, a power-aware technique can also help to reduce the energy cost. Power-aware strategies can be activated either in hardware or software level [56]. For instance, dynamic component deactivation (DCD) strategy at hardware level is applied

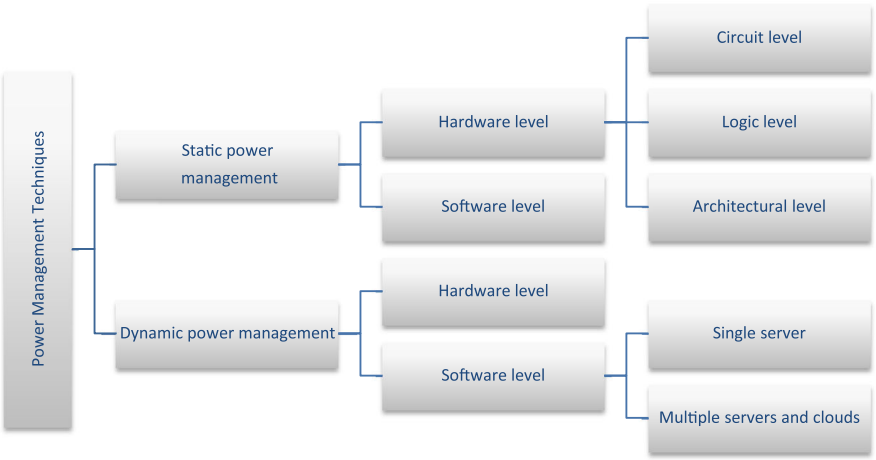

Fig. 5 Taxonomy of power management techniques [48]

along with Advanced Configuration and Power Interface (ACPI[1]) strategy at software level, since even with optimized hardware, poor software level design or optimization can lead to extensive power losses [30].

Since temperature is closely related to the power density, the power/energy factor is involved in the process of calculating dynamic criticality in power-aware allocation and scheduling. Power-aware technologies either use low power energy-efficient hardware equipment (e.g., CPUs and power supplies) to reduce energy usage and peak power consumption, or reduce energy usage based on the knowledge of current resource utilization and application workloads. Power-aware scheduling process works at circuit, device, architectural, compiler, operating system, and networking layers [57]. The most efficient and direct method is to use more power efficient component in the hardware design phase. Other approaches include developing algorithms for scaling down power or even turning down a system for unused resources [56]. In this regard, Beloglazov et al. [30] proposed a high-level taxonomy of power and energy management, Fig. 5.

3.3.2 Thermal-aware

Thermal-aware management predicts the thermal effects of a task placement and the resource allocation is based on the predicted thermal effect [30,58]. The thermal-aware approaches take the temperature as one of the major considerations for resource allocation. The temperature depends on the power consumption of each processing element, dimension, and relative location on the embedded system platform [60]. The goal of thermal-aware allocation is to minimize peak air inlet temperature resulting in minimization of the cost of cooling. Thermal-aware scheduling approaches keep the

[1] http://www.acpi.info/.

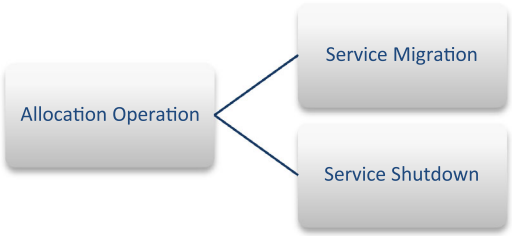

Fig. 6 Allocation operation taxonomy

temperature of all the data processing equipment below a certain threshold and at the same time maximize the energy efficiency of the system [61].

3.4 Allocation operation

In this paper, we have broadly classified the allocation operation for optimizing energy efficiency of cloud resource into following categories: (a) service migration and (b) service shutdown. The details of service migration and service shutdown allocation operations are provided below. Figure 6 presents the allocation operation taxonomy considered for the study.

3.4.1 Service migration

The transferring of process states and local data of an application component instance (e.g. web server and database server) to a new CPU resource (virtual machine container or virtual server) is called service migration. The service migration process enables: (a) dynamic load distribution by migrating processes from overloaded CPU or storage resources to less loaded ones, (b) fault resilience by migrating processes from cloud resources that may have experienced a partial failure, (c) eased system administration by migrating processes from the cloud resources that are about to be shut down or otherwise made unavailable, and (d) data access locality by migrating processes closer to the source of some data [60]. The major decision concerns of a service migration process are the time when a migration will occur, the selection process of the service which will migrate, and at which destination resource a service will move.

Although, there are different power-aware algorithms for the host overload/underload detection, CPU selection, and CPU placement, the service migration has still a lot of complexities. For instance, consider an application component migration from small instance to medium instance or large instance. These instances vary in their hardware configuration such as RAM, cores, local storage size, addressing bits and I/O performance. This issue will be more complicated in case of service migration to heterogeneous resources such as CPU in which, for example, Amazon EC2 offers more than 10 types of CPU resources[2] and also migrating across diverse cloud providers. In this regard, a sample scenario would be migration of a service from a server with

[2] http://aws.amazon.com/ec2/instance-types/.

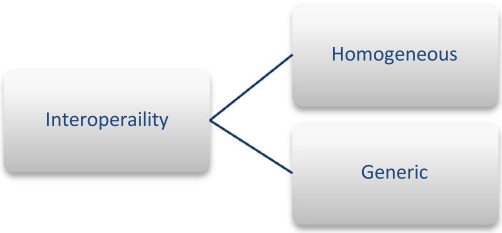

Fig. 7 Interoperability taxonomy

32 virtual cores in GoGrid provider to a server with extra-large CPU instance (which has 40 virtual cores) in Amazon.

3.4.2 Service shutdown

Service shutdown refers to automatic switching/powering off the system, hardware component, or network on the failure or abnormal termination of the previously active resource allocation. Service shutdown can be automatic or might require human intervention [62]. One of the major reasons for service shutdown is to conserve energy. However, before shutting down a server, all the running services have to be consolidated and migrated to other nodes power and migration cost aware application placement algorithms in virtualized systems [15,63,64].

3.5 Interoperability

The Interoperability refers to the applicability domain of energy optimization technique. In fact, they can be applied to single, multiple, or both cloud resource types, in which they are called homogeneous and generic respectively. A generic resource allocation has ability to operate across multiple cloud resource types including hardware, software, and application. With implementation of generic technique, a datacenter vendor can actuate the energy efficiency operations across multiple ICT resources during the resource allocation phase. On the other hand, a homogeneous resource allocation technique is only capable of effecting the energy efficiency operation on a single datacenter resource type (Fig. 7).

Table 1 summarizes the energy-efficient resource allocation approaches evaluated with the dimensions of adaption policy, type of resource, objective function, allocation method, allocation operation, and Interoperability constraints.

4 State of the art in resource allocation mechanisms

Lee and Zomaya [65]. The authors proposed two energy-aware task consolidation mechanisms (ECTC and MaxUtil). The goal of the aforementioned heuristics was to maximize resource utilization, taking into consideration active and idle energy consumption. The energy consumed by a task was calculated based on an objective function. The objective function was derived through number of experiments. The

Table 1 Comparison of energy-efficient resource allocation approaches

Approach	Adaptation policy	Type of resource	Objective function	Allocation method	Allocation operation	Interoperability
Lee and Zomaya [65]	Reactive	CPU	Composite	Power-aware	Service migration	Generic, homogenous CPU + memory
Srikantaiah et al. [15]	Reactive	CPU + disk	Composite	Power-aware	Service migration	Generic resources
Torres et al. [66]	Predictive	CPU + memory	Single	Power-aware	Service shutdown	Generic application workload
Nathuji and Schwan [64]	Predictive	CPU + memory	Composite	Power-aware	Service migration	Generic, CPU bound applications, homogeneous processors
Subrata et al. [67]	Hybrid	CPU	Composite	Power-aware	Service migration	Generic
Mazzucco et al. [68]	Predictive + reactive	CPU	Composite	Power-aware	Service shutdown	Specific, homogeneous servers
Raghavendra et al. [69]	Reactive	CPU	Composite	Power-aware	Service shutdown	Generic application workloads
Kusic et al. [70]	Reactive	CPU	Composite	Power-aware	Service shutdown	Generic
Cardosa et al. [71]	Reactive	CPU	Composite	Power-aware	Service migration	Generic
Verma et al. [63]	Hybrid	CPU	Composite	Power-aware	Service shutdown	Generic (applicable to virtualized cluster)
Gandhi et al. [72]	Predictive	CPU + memory + disk + cache	Composite	Power-aware	Service shutdown	Generic, homogeneous servers with heterogeneous workloads
Gong and Xu [73]	Reactive	CPU	Composite	Power-aware	Service migration	Specific, homogeneous servers with heterogeneous workloads
Chen et al. [50]	Reactive	CPU + memory	Composite	Power-aware	Service shutdown	Generic, homogeneous servers with heterogeneous workloads

performance evaluation of ECTC and MaxUtil heuristics were carefully evaluated with number of experiments with a diverse set of tasks. Both algorithms are similar in nature but the main difference is the computation of the cost function. The authors assumed that relocation of running tasks can reduce energy consumption. Therefore, the variants of the aforementioned algorithms were further implemented to incorporate task migration.

Lee and Zomaya also suggested that the total energy utilization can considerably be reduced by combining more than one task instead of assigning it individually to a server. Moreover, the authors also focused on the characteristics and issues of resource management. The underline assumption of energy model was the incorporation of the resources in the target system with an effective power-saving mechanism. Moreover, the energy model had a direct relationship with processor utilization. The ECTC and MaxUtil algorithms assign each task to the minimum energy consumption resource without any degradation in performance. The results of experiments reveal promising energy-savings.

Srikantaiah et al. [15]. The authors evaluated the relationship between resource and energy consumption. Moreover, the performances of consolidated workloads were evaluated. The authors used two main features including CPU cycles, and disk usage in a bin-packing problem for task consolidation. The authors merged tasks and balance energy consumption by computing optimal points based on Pareto frontier algorithm. There are two major steps incorporated in the proposed technique. The first step was to compute optimal points from profiling data. The profiling step was used by an energy aware resource allocation mechanism. For each server, the mechanism computes the Euclidean distance measure between the current and the ideal point.

Based on the experimental results, the authors concluded that the outcome of energy expenditure per transaction is a "U" shaped curve, and they have figured out the optimal utilization point from the curve. The experiments computed energy usage, performance changes, and resource utilization as multiple workloads with varying resource usages are combined on common servers. The study revealed the energy performance trade-offs for consolidation and indicated that optimal operating point. The paper focused only on a practicable but vital domain covered by CPU and disk resource combinations. The paper focused on some of the intricacies involved in performing consolidation for power optimization and proposed practical directions to address the challenges involved.

Tang et al. [57]. The authors proposed a utility analytic model to represent the interaction between computing servers arrival requests based on the queuing theory. The implemented model considered real time tasks as a request. The authors combined a linear and low complexity heat recirculation power model for the modeling inlet temperatures. The model also considered the impact factor measure to reflect the task performance degradation. Moreover, the distribution of incoming task to maximize the supply temperature while respecting the redline temperatures was also modeled in the paper.

The authors used task assignment approaches to address the issue of minimizing the cooling cost as the problem of reducing the peak inlet temperature. The experiments were conducted on a real world e-book database and e-commerce web ser-

vice. The main performance objective was to have the: (a) same QoS guarantee like present in dedicated servers, and (b) maximize resource utilization along with reduced energy consumption. Some of the features of proposed algorithms are cooling-oriented thermal-aware assignment, no performance negotiation, uniformity of equipment, and low-complexity heat recirculation model.

Torres et al. [66]. The authors proposed a technique to minimize the total number of active servers to process heterogeneous workload without performance penalties in terms of degradation of the QoS. The proposed technique also considered a real workload of a case study (a national travel website) for experiments. The algorithm combined two interesting techniques: (a) request discrimination and (b) memory compression. The request discrimination blocked unnecessary requests to eliminate unfavorable resource usage. The memory compression allows more tasks consolidation by converting power consumed by CPU into memory capacity. The results show the effectiveness of the proposed approach because the task was accomplished with 20 % less computations.

The algorithms decrease the number of nodes required for a certain service level criteria. The main contribution of the research was to reflect and validate that the consolidation through virtualization of heterogeneous workloads is not only the major factor to save energy. Moreover, the authors presented alternate procedures of rescuing resources through reducing the resource wasted. The authors also identified new prospects to improve the energy efficiency and reducing the resources required without negatively influencing the performance and user satisfaction. The obtained results show that the combined use of memory compression and request discrimination can dramatically boost the energy savings.

Nathuji and Schwan [64]. The authors proposed a virtual power approach that combined 'hard' and 'soft' scaling methods and integrated task consolidation into power management. The technique had a comprehensive virtualization methodology to provision effective policies of power management. Both of the algorithms proposed in the paper were based on power management services of DVFS and resource usage control made with CPUs and physical processors. The experiments demonstrated the benefits of active power management with the use of multiple processors. The techniques were deployed on power-efficient Intel core micro-architecture and system utilization is improved by up to 35 % using Qstates.

In particular, the consolidation of multiple customer applications onto multicore servers introduces performance interference between collocated workloads, significantly impacting application QoS. To address this challenge, the authors advocate that the cloud should transparently provision additional resources as necessary to achieve the performance that customers would have realized if they were running in isolation. Accordingly, Q-Clouds was developed [8] that is a QoS-aware control framework that tunes resource allocations to mitigate performance interference effects. Q-Clouds uses online feedback to build a multi-input multi-output (MIMO) model that captures performance interference interactions, and uses it to perform closed loop resource management. In addition, the authors utilize this functionality to allow applications to specify multiple levels of QoS as application Q-states. For such applications, Q-

Clouds dynamically provisions underutilized resources to enable elevated QoS levels, thereby improving system efficiency.

Subrata et al. [67]. In the paper a cooperative, for web-scale systems a power-aware scheduling framework was proposed. The proposed scheduling scheme includes Nash Bargaining Solution and maintains a specified QoS measure among all the service providers to reduce energy usage. The level of energy usage was under controlled at a certain threshold limit to maintain the desired QoS measure dictated by SLA. The proposed algorithm fairly allocates the resources among all end-users and shows robust performance improvements against all the performance inaccuracies in prediction of information. The experiments revealed the significant energy savings for a targeted QoS level.

In this paper, the authors proposed a cooperative power-aware scheduling algorithm for web-scale distributed systems. The proposed algorithm directly takes into account the multitude of ownerships of providers and incentives for cooperation, and simultaneously maximizes the utility of the providers. The schemes can be considered semi-static, as they respond to changes in system states during runtime. However, they do not use as much information as traditional dynamic schemes; as such, they have relatively low overhead, are relatively insensitive to inaccuracies in performance prediction information, and are stable.

Csorba et al. [74]. This paper proposed a virtual power approach that supports the isolated and independent operation running on virtual machines. The virtual power technique self-organized the placement of CPU images onto physical servers in a cloud infrastructure. The goal of the algorithm was to improve scalability. The algorithm also handled the dynamism of resources by using Cross-Entropy Ant system [75]. The Ant system also calculated the best placement mappings of CPUs to the servers present within the cloud infrastructure. The experiments were conducted on a real world scenario for a large number of virtual machine image replicas that are deployed concurrently. The results revealed that without performance penalties, application requirements can be met and resulting in power consumption reduced up to 34 %.

This paper conjectures that using self-organizing techniques for system (re)configuration can improve both the scalability properties of such systems as well as their ability to tolerate churn. Specifically, the paper focuses on deployment of virtual machine images onto physical machines that reside in different parts of the network. The objective is to construct balanced and dependable deployment configurations that are resilient. To accomplish this, a method based on a variant of Ant Colony Optimization is used to find efficient deployment mappings for a large number of virtual machine image replicas that are deployed concurrently. The method is completely decentralized; ants communicate indirectly through pheromone tables located in the nodes.

This paper examines the effects of a hybrid environment in which services are deployed in the private cloud, public clouds, or both depending on the present usage pattern. Such a scenario is especially interesting with respect to handling load overshoots that may be caused by dependability and/or performance requirements. For example, as the service usage pattern change, CPU instances may be added or removed from the public cloud, while retaining the same number of CPU instances within the

private cloud. During execution in such a hybrid cloud environment, a plethora of highly dynamic parameters influence the optimal deployment configurations, e.g. due to the influence of concurrent services and varying client load. Ideally, the deployment mappings should minimize and balance resource consumption, yet provide sufficient resources to satisfy the dependability requirements of services.

Fujiwara et al. [24]. A market-based framework for allocation of different resource requests in a cloud computing infrastructure has been presented by Fujiwara et al. [24]. The resources were virtualized and delivered to end-users as different set of services. The approach allowed end-users to request an arbitrary combination of services to different providers. The mechanism utilized the forward and spot market independently to make predictable and flexible allocations at the same time. The proposed technique considered the massive amount of providers' ownership incentives for cooperation. The algorithm maximized computing as a utility from the service providers. The experimental results showed that the proposed mechanism scale up to a feasible allocation for resources in cloud infrastructures.

The authors used mixed integer programming method to strictly maximize the economic efficiency. Moreover, the proposed mechanism employed dual market mechanisms, the forward market for advance reservations of resource and the spot market for immediate reservations. The proposed mechanism accepts combinational bids, with which the user can express complemental requirements for resource allocation. Moreover, both resource providers and users compete to offer/receive resources and the algorithm employed the double-sided auction model. The proposed mechanism is characterized by three properties: (a) the bidding language, (b) the allocation scheme, and (c) the pricing scheme.

Mazzucco et al. [68]. The authors proposed an energy aware resource allocation mechanism to maximize the profits incurred by service providers. The resource allocation methods that were introduced and evaluated are based on the dynamic states such as on and off of powering servers during experiments. The goal was to minimize the amount of electricity consumed and to maximize the user experience. This is achieved by improving the utilization of the server farm, i.e., by powering excess servers off. The policies are based on (a) dynamic estimates of user demand, and (b) models of system behavior.

The proposed technique was not appropriate for scheduling applications having critical performance requirements. The development model scheduled the servers for completing customer jobs with deadlines. The emphasis of the latter is on generality rather than analytical tractability. Some approximations are done to handle the resulting models. However, those approximations lead to algorithms that perform well under different traffic conditions and can be used in real systems. The authors used a data center composed of 25,000 machines and assume a server farm with a Power Usage Effectiveness (PUE) of 1.7. Moreover, the power consumption of each machine used ranges between 140 and 220 W and the cost for electricity, r, is 0.1 $ per kWh. The results of the experiments and simulations revealed the effectiveness of proposed approach under different traffic conditions.

Raghavendra et al. [69]. The authors explored a combination of five dissimilar power management policies in a data center environment. In the proposed framework, to coordinate controllers' actions, the authors applied a feedback control loop mechanism. The approach adopted is independent of different workload types and was based on the similar assumption made by Nathuji and Schwan [64]. The authors only considered CPU allocation as the resource for the experiments. Moreover, the authors claimed that with different types of workload the actual power savings change. However, the benefits of the approach were similar for all different types of workloads. The disadvantage of the approach was failing to support strict and variable SLAs for number of applications in cloud infrastructures.

The authors proposed and evaluated a coordinated architecture for peak and average power management across hardware and software for complex enterprise environments. The proposed approach leverages a feedback mechanism to federate multiple power management solutions and builds on a control-theoretic approach to unify solutions for tracking, capping, and optimization problems, with minimal interfaces across controllers. Simulation results, based on server traces from real-world enterprises, demonstrate the correctness, stability, and efficiency advantages. Second, the authors perform a detailed quantitative evaluation of the sensitivity of such a coordinated solution to different architectures, implementations, workloads, and system design choices. The five solutions that we considered in our proposed architecture are representative of the key attributes and challenges in previously proposed power management solutions, e.g., average versus peak, local versus global, per-server versus cluster, power versus performance, and fine-grained versus coarse-grained.

Kusic et al. [70]. The power management was sequentially optimized and addressed using Limited Look ahead Control (LLC) technique for virtualized heterogeneous environments. This approach allows for multi-objective optimization under explicit operating constraints and is applicable to computing systems with non-linear dynamics where control inputs must be chosen from a finite set. The major goal was to: (a) maximize the profit of the resource provider, (b) decrease SLA violations, and (c) minimize power consumption. The approach predicted the forthcoming state of the system and performed necessary reallocations by the use of Kalman filter [76]. The proposed approach also required simulation-based learning mechanisms instead of relying on heuristics for the application specific adjustments.

The revenue generated by each service is specified via a pricing scheme or service-level agreement (SLA) that relates the achieved response time to a dollar value that the client is willing to pay. The control objective is to maximize the profit generated by this system by minimizing both the power consumption and SLA violations. The LLC formulation models the cost of control, i.e., the switching costs associated with turning machines on or off. The aforementioned model was quite complex and had several steps that resulted in severe degradation of performance with varying workload demand. The running time of the controller used during experiments was 30 min for running 15 nodes that was not appropriate for running applications on real-time systems.

Cardosa et al. [71]. For virtualized heterogeneous computing environments, the problem of power efficient resource allocation was also addressed by Cardosa et al. [71].

The authors proposed a group of procedures in data centers for dynamic placement and power amalgamation of CPU. The aforementioned algorithms controlled minimum and maximum percentage of the CPU utilization for the virtual machines having similar workload. The resource allocation uses application heterogeneity and ensures the high application utility. The technique suits only enterprise environments and requires the knowledge of application priorities [69]. The allocation of Virtual Machines is static and only CPU is used as a resource during reallocation.

Verma et al. [63]. The problem of power-aware dynamic application placement as continuous optimization problem was investigated in virtualized heterogeneous environments by Verma et al. [63]. In the proposed model titled "pMapper", at any time instant, the problem of placement of CPUs on physical servers is addressed and optimized to maximize performance issues and minimize total power consumption. The authors also used a heuristic similar to [15]. The authors also used a similar live migration algorithm of virtual machines [64]. The experimental evidence was established to show the efficiency of pMapper. The proposed algorithms were unable to handle strict requirements of SLAs because of the different violations in the workload.

5 Discussion and future directions

We identified some key problems that can be addressed for energy aware resource allocation in clouds. The concept of virtualization offers the capability to transfer CPUs from physical servers to the cloud infrastructure. The mechanism of static or dynamic clustering of different CPUs on a physical server depending upon varying resource requests can be used. During the process, some idle servers can be either switched off, put to sleep, or in hibernating mode so that the total energy consumption is reduced. We already discussed several resource allocation techniques. However, forceful clustering of CPUs will result in performance penalties due to SLA violation. For this reason, earlier resource allocation approaches effectively addressed the balancing of factors such as, performance measures and energy usage.

The cloud computing paradigm offers services that provide the ability to assign virtual machines and provision a number of applications to the cloud users. By employing the aforementioned technique, the different types of applications can be consolidated on a physical computer server. The applications may not be related to one another because of static or variable workload. One of the research problems is to figure out what type of applications can be consolidated to a single working server. A similar research direction is also reported in [29–31,33]. The aforementioned technique, results in efficient resource allocations and consolidation for maximizing throughput. Moreover, the focus of such consolidation is on uniform types of workload on servers and the servers do not take in to consideration divergent types of the applications running on the virtual machines.

The virtual machines communicate with one another in a network of different topologies. If the allocation of resources is not done in an optimized way, then many migrations of processes will occur. A similar research direction is also presented in [29–31]. The result of such a scheme is an expensive data transfer because CPUs are logically hosted on distant physical servers. With this scheme, the communication

is a bottle neck and involves switches, access-points, and routers that also consume power. The aforementioned scheme will also result in delaying packets because packets need to travel across the network. To eliminate the data transfer delays and reduce power consumption, observing the communication pattern among CPUs is important. Therefore, more work is needed to place CPUs on the same or closely located servers.

In addition to energy efficient allocation of resources, application interface may offer different levels of performance to end-users. Therefore, in Cloud computing QoS aware resource allocation policies also plays an important role. A comprehensive study of services offered by cloud and workload distribution is needed to identify common patterns of behaviors. A similar research direction is also presented in [31,32]. More efforts are needed to study the relationship between varying workloads, while an attempt should be made to build frameworks that can minimize the trade-offs between SLA and provide energy efficiency in algorithms.

6 Conclusion

In recent years, energy efficient resource allocation of datacenter resources has evolved as one the critical research issue. We have identified power and energy inefficiencies in hardware and software, and categorized existing techniques in the literature along with the summary of their benefits and limitations. We have discussed the resource allocation problem in general along with associated challenges and issues. Moreover, we discussed the advantages and disadvantages of various resource allocation strategies in literature, and highlighted open issues and future directions.

References

1. Mell P, Grance T (2009) Definition of cloud computing. Technical report SP 800–145, National Institute of Standard and Technology (NIST), Gaithersburg, MD
2. Wang L, Kunze M, Tao J, Laszewski G (2011) Towards building a cloud for scientific applications. Adv Eng Softw 42(9):714–722
3. Wang L, Laszewski G, Younge AJ, He X, Kune M, Tao J, Fu C (2010) Cloud computing: a perspective study. New Gener Comput 28(2):137–146
4. Wang L, Fu C (2010) Research advances in modern cyberinfrastructure. New Gener Comput 28(2):111–112
5. Wang L, Chen D, Zhao J, Tao J (2012) Resource management of distributed virtual machines. IJAHUC 10(2):96–111
6. Wang L, Chen D, Huang F (2011) Virtual workflow system for distributed collaborative scientific applications on Grids. Comput Electr Eng 37(3):300–310
7. Wang L, Laszewski L, Chen D, Tao J, Kunze M (2010) Provide virtual machine information for grid computing. IEEE Trans Syst Man Cybern Part A 40(6):1362–1374
8. Nathuji R, Kansal A, Ghaffarkhah A (2010) Q-Clouds: managing performance interference effects for QoS-aware clouds. In: 5th European conference on computer system (EuroSys'10), pp 237–250
9. Google Whitepaper (2011) Google's green data centers: network POP case study. Google. http://static.googleusercontent.com/external_content/untrusted_dlcp/www.google.com/en/us/corporate/datacenter/dc-best-practices-google.pdf
10. Armbrust M, Fox A, Griffith R, Joseph AD, Katz R, Konwinski A, Lee G, Patterson D, Rabkin A, Stoica I, Zaharia M (2010) A view of cloud computing. Commun ACM 53(4):50–58
11. Sadashiv N, Kumar S (2011) Cluster, grid and cloud computing: a detailed comparison. In: 6th international conference on computer science and education (ICCSE 2011), pp 477–482

12. Jansen W (2011) Cloud hooks: security and privacy issues in cloud computing. In: 44th Hawaii international conference on systems science (HICSS), pp 1–10
13. Barroso LA, Hölzle U (2009) The datacenter as a computer: an introduction to the design of warehouse-scale machines, 1st edn. In: Hill MD (ed) Morgan and Claypool Publishers, University of Wisconsin, Madison
14. Berl A, Gelenbe E, Girolamo MD, Giuliani G, Meer HD, Dang MQ, Pentikousis K (2010) Energy-efficient cloud computing. Comput J 53(7):1045–1051
15. Srikantaiah S, Kansal A, Zhao F (2008) Energy aware consolidation for cloud computing. In: Conference on power aware computer and systems (HotPower '08)
16. Lee YC, Zomaya AY (2012) Energy efficient utilization of resources in cloud computing systems. J Supercomput 60(2):268–280. doi:10.1007/s11227-010-0421-3
17. Ourghanlian B (2010) Improving energy efficiency: an end-user perspective, the green grid EMEA technical forum the green grid. http://www.thegreengrid.org/~/media/EMEATechForums2010/Improving%20Energy%20Efficiency%20-%20An%20End%20User%20Perspective_Paris.pdf?lang=en. Accessed 3 Oct 2011
18. Paradiso JA, Starner T (2005) Energy scavenging for mobile and wireless electronics. Pervasive Comput 4(1):18–27
19. Elnozahy M, Kistler M, Rajamony R (2002) Energy-efficient server clusters. Power aware computer systems, vol 2325. Springer, Berlin, pp 179–197
20. Sharma V, Thomas A, Abdelzaher T, Skadron K (2003) Power-aware QoS management in web servers. In: Real-time systems symposium (RTSS 2003), pp 63–72
21. Horvath T, Abdelzaher T, Skadron K, Liu X (2007) Dynamic voltage scaling in multitier web servers with end-to-end delay control. IEEE Trans Comput 56(4):444–458
22. Liu X, Shenoy P, Gong W (2004) A time series-based approach for power management in mobile processors and disks. In: 14th international workshop on network and operating systems support for digital audio and video (NOSSDAV '04), pp 74–79
23. Steere DC, Goel A, Gruenberg J, Mcnamee D, Pu C, Walpole J (1999) A feedback-driven proportion allocator for real-rate scheduling. In: Third symposium on operating system designand implementation (OSDI), pp 145–158
24. Fujiwara I, Aida K, Ono I (2009) Market based resource allocation for distributed computing. IPSJ SIG Technical Report 1, 34
25. Wei G, Vasilakos A, Zheng Y, Xiong N (2009) A game-theoretic method of fair resource allocation for cloud computing services. J Supercomput 54(2):252–269
26. Shu W (2007) Optimal resource allocation on grid computing using a quantum chromosomes genetic algorithm. In: IET conference on wireless, mobile and sensor networks (CCWMSN07), pp 1059–1062
27. Ismail L, Mills B, Hennebelle A (2008) A formal model of dynamic resource allocation in grid computing environment. In: 9th ACIS international conference on software engineering, artificial intelligence, networking, and parallel/distributed computing (SNPD '08), pp 685–693
28. Huang Y, Chao B (2001) A priority-based resource allocation strategy in distributed computing networks. J Syst Softw 58(3):221–233
29. Beloglazov A, Abawajy J, Buyya R (2012) Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. Future Gener Comput Syst 28:755–768
30. Beloglazov A, Buyya R, Lee YC, Zomaya AY (2011) A taxonomy and survey of energy-efficient data centers and cloud computing systems. Adv Compt 82:47–111
31. Buyya R, Broberg J, Goscinski A (2011) Cloud computing principles and paradigms. Wiley, Hoboken
32. Malet B, Pietzuch P (2010) Resource allocation across multiple cloud data centres. In: 8th international workshop on middleware for grids, clouds and e-science (MGC '10), pp 1–6
33. Demchenko Y, Ham J, Strijkers R, Ghijsen M, Ngo C, Cristea M (2011) Generic architecture for cloud infrastructure as a service (IaaS) provisioning model, Technical report SNE-UVA-2011-03, System and Network Engineering Group, University van Amsterdam
34. GESI (2008) Smart 2020: enabling the low carbon economy in the information age.http://www.smart2020.org/_assets/files/02_Smart2020Report.pdf.Accessed 3 Oct 2011
35. Gupta M, Singh S (2003) Greening of the internet. In: Applications technology of architecture, protocols and computer communication, pp 19–26
36. Koomey J (2007) Estimating total power consumption by servers in the US and the world. Lawrence Berkeley National Laboratory, Analytics Press, CA, p 31. http://sites.amd.com/de/Documents/svrpwrusecompletefinal.pdf. Accessed 3 Oct 2011

37. Singh T, Vara P (2009) Smart metering the clouds. In: 18th IEEE international workshops on enabling technologies: infrastructures for collaborative enterprises, pp 66–71
38. Baliga J, Ayre R, Hinton K, Sorin W, Tucker R (2009) Energy consumption in optical IP networks. J Lightweight Technol 27(13):2391–2403
39. Tamm O, Hermsmeyer C, Rush A (2010) Eco-sustainable system and network architectures for future transport networks. Bell Labs Tech J 14(4):311–327
40. Vukovic A (2005) Datacenters: network power density challenges. J ASHRAE 47:55–59
41. Liu J, Zhao F, Liu X, He W (2009) Challenges towards elastic power management in internet datacenters. In: IEEE international conference on distributed systems, pp 65–72
42. Chase J, Anderson D, Thakur P, Vahdat A (2001) Managing energy and server resources in hosting centers. In: 18th symposium on operating systems principles (SOSP '01), pp 103–116
43. Hermenier F, Loriant N, Menaud J (2006) Power management in grid computing with Xen. Lecture notes in computer science. Springer, Berlin
44. Cook G, Horn J (2011) How dirty is your data. GreenPeace International, Amsterdam
45. Ranjan R, Benatallah B (2012) Programming cloud resource orchestration framework: operations and research challenges. CoRR abs/1204.2204
46. Duy TVT, Duy S, Inoguchi Y (2010) Performance evaluation of a green scheduling algorithm for energy savings in cloud computing. In: 2010 IEEE International Symposium on Parallel and distributed processing, workshops and PhD forum (IPDPSW), pp 1–8, 19–23
47. Mezmaz M-S, Kessaci Y, Lee YC, Melab N, Talbi E-G, Zomaya AY, Tuyttens D (2010) A parallel island-based hybrid genetic algorithm for precedence-constrained applications to minimize energy consumption and makespan. In: GRID, pp 274–281
48. Hussin M, Lee YC, Zomaya AY (2011) Efficient energy management using adaptive reinforcement learning-based scheduling in large-scale distributed systems. In: ICPP, pp 385–393
49. Kalyvianaki E (2009) Resource provisioning for virtualized server applications. Technical Report UCAM-CL-TR-762, Computer Laboratory, University of Cambridge
50. Chen Y, Gmach D, Arlitt M, Marwah M, Gandhi A (2011) Minimizing data center SLA violations and power consumption via hybrid resource provisioning. In: Second international green computing conference (IGCC 2011), pp 1–8
51. Tan CH, Luo M, Zhao YZ (2010) Multi-agent approach for dynamic resource allocation. SIMTech technical reports (STR_V11_N3_03_MEC), vol 11, No. 3
52. Xie T, Wilamowski B (2011) Recent advances in power aware design. In: 37th annual conference on IEEE industrial electronics society IECON 2011, pp 4632–4635
53. Fu S (2005) Service migration in distributed virtual machines for adaptive computing. In: International conference on parallel processing (ICPP 2005), pp 358–365
54. Singh R, Sharma U, Cecchet E, Shenoy P (2010) Autonomic mix-aware provisioning for non-stationary data center workloads. In: Proceedings of the 7th IEEE international conference on autonomic computing and communication (ICAC '10)
55. Moreno IS, Xu J (2011) Energy-efficiency in cloud computing environments: towards energy savings without performance degradation. Int J Cloud Appl Comput 1(1):17–33
56. Poladian V, Garlan D, Shaw M, Satyanarayanan M, Schmerl B, Sousa J (2007) Leveraging resource prediction for anticipatory dynamic configuration. In: Proceedings of the first international conference on self-adaptive and self-organizing systems (SASO '07)
57. Tang Q, Gupta S, Varsamopoulos G (2008) Energy-efficient, thermal-aware task scheduling for homogeneous, high performance computing data centers: a cyber-physical approach. IEEE Trans Parallel Distrib Syst 19(11):1458–1472
58. Goldman C, Reid M, Levy R, Silverstein A (2010) Coordination of energy efficiency and demand response. Environmental Energy Technologies Division, Berkeley National Laboratory
59. Khargharia B, Hariri S, Yousif MS (2008) Autonomic power and performance management for computing systems. Clust Comput 11(2):167–181
60. Hung W-L, Xie Y, Vijaykrishnan N, Kandemir M, Irwin MJ (2005) Thermal-aware task allocation and scheduling for embedded systems. In: Proceedings of the conference on design, automation and test in Europe (DATE '05), vol 2, pp 898–899
61. Vasic N, Scherer T, Schott W (2010) Thermal-aware workload scheduling for energy efficient data centers. In: 7th international conference on autonomic computing (ICAC '10), pp 169–174
62. Cai C, Wang L, Khan SU, Jie T (2011) Energy-aware high performance computing—a taxonomy study. In: 17th international conference on parallel and distributed systems (ICPADS), pp 953–958

63. Verma A, Ahuja P, Neogi A (2008) pMapper: power and migration cost aware application placement in virtualized systems. In: 9th ACM/IFIP/USENIX international conference on middleware (Middleware '08), pp 243–264
64. Nathuji R, Schwan K (2007) VirtualPower: coordinated power management in virtualized enterprise systems. In: 21st ACM SIGOPS symposium on operating systems principles (SOSP'07), pp 265–278
65. Lee Y, Zomaya A (2010) Energy efficient utilization of resources in cloud computing systems. J Supercomput 1(13):1–13
66. Torres J, Carrera D, Hogan K, Gavaldà R, Beltran V, Poggi N (2008) Reducing wasted resources to help achieve green data centers. In: IEEE international symposium on parallel and distributed proceedings (IPDPS 2008), pp 1–8
67. Subrata R, Zomaya AY, Landfeldt B (2010) Cooperative power-aware scheduling in grid computing environments. J Parallel Distrib Comput 70(2):84–91
68. Mazzucco M, Dyachuk D, Deters R (2010) Maximizing cloud providers' revenues via energy aware allocation policies. In: 3rd international conference on cloud computing (CLOUD), pp 131–138
69. Raghavendra R, Ranganathan P, Talwar V, Wang Z, Zhu X (2008) No "power" struggles: coordinated multi-level power management for the data center. SIGARCH Comput Archit News 36(1):48–59
70. Kusic D, Kephart JO, Hanson JE, Kandasamy N, Jiang G (2009) Power and performance management of virtualized computing environments via look ahead control. Cluster Comput 12(1):1–15
71. Cardosa M, Korupolu M, Singh A (2009) Shares and utilities based power consolidation in virtualized server environments. In: 11th IFIP/IEEE integrated network management (IM 2009), pp 327–334
72. Gandhi A, Harchol-Balter M, Das R, Lefurgy C (2009) Optimal power allocation in server farms. In: 11th international joint conference on measurement and modeling of computer systems (SIGMETRICS '09), pp 157–168
73. Gong J, Xu C-Z (2010) A gray-box feedback control approach for system-level peak power management. In: 39th international conference on parallel proceedings (ICPP'10), San Diego, CA
74. Csorba MJ, Meling H, Heegaard PE (2010) Ant system for service deployment in private and public clouds. In: 2nd workshop on bio-inspired algorithms for distributed systems (BADS '10), pp 19–28
75. Heegaard P, Helvik B, Wittner O (2008) The cross entropy ant system for network path management. Telektronikk 104(1):19–40
76. Grewal M, Andrews A (2010) Applications of Kalman filtering in aerospace 1960 to the present. Control Syst 30(3):69–78