J Supercomput (2013) 65:445–468
DOI 10.1007/s11227-011-0722-1

State-of-the-art research study for green cloud computing

Si-Yuan Jing · Shahzad Ali · Kun She · Yi Zhong

Published online: 8 December 2011

Abstract Although cloud computing has rapidly emerged as a widely accepted computing paradigm, the research on cloud computing is still at an early stage. Cloud computing suffers from different challenging issues related to security, software frameworks, quality of service, standardization, and power consumption. Efficient energy management is one of the most challenging research issues. The core services in cloud computing system are the SaaS (Software as a Service), PaaS (Platform as a Service), and IaaS (Infrastructure as a Service). In this paper, we study state-of-the-art techniques and research related to power saving in the IaaS of a cloud computing system, which consumes a huge part of total energy in a cloud computing system. At the end, some feasible solutions for building green cloud computing are proposed. Our aim is to provide a better understanding of the design challenges of energy management in the IaaS of a cloud computing system.

Keywords Cloud computing · IaaS · Data center · Energy efficiency · Virtualization

1 Introduction

Over the past few years, cloud computing has attracted considerable attention and it has rapidly emerged as a widely accepted computing paradigm. The research and development community has quickly reached consensus on core concepts such as on-demand computing resources, elastic scaling, elimination of up-front capital and operational expense, and establishing pay-as-you-go business model for computing and information technology services.

Although cloud computing has been widely adopted by the industry, the research on cloud computing is still at an early stage. Many existing issues have not been fully

S.-Y. Jing (✉) · S. Ali · K. She · Y. Zhong
University of Electronic Science and Technology of China, Chengdu, China
e-mail: jingsiyuan_628@126.com

addressed, while new challenges keep emerging from industry applications. Energy management is one of the challenging research issues [1, 2].

Cloud Infrastructure is the most important component in a cloud. It may comprise tens of thousands of servers, network devices and disks, and typically serve millions of users globally. Such a large-scale data center will consume considerable amount of energy. For example, according to Amazon's estimates, at its data centers, expenses related to the cost and operation of the servers account for 53% of the total budget (based on a 3-year amortization-schedule), while energy-related costs amount to 42% of the total, and include both direct power consumption (approx. 19%) and the cooling infrastructure (23%) amortized over a 15-year period [3]. Therefore, improving the energy efficiency in cloud data center, without sacrificing Service Level Agreements (SLA), can save significant energy budget for data center operators and would also make a significant contribution to greater environmental sustainability.

In this paper, we review the core concepts and characteristics in a cloud computing system and moreover study the state-of-art techniques and research related to the IaaS service layer of a cloud computing system, which can lead to a reduction of energy consumption in a cloud infrastructure. Furthermore, some feasible solutions for building green cloud computing are proposed. Our aim is to provide a better understanding of the design challenges of energy management in IaaS of a cloud computing system.

The remainder of this paper is organized as follows. In Sect. 2, we review the core concepts and architecture of a cloud computing system. In Sect. 3, we study techniques and research which can lead to energy efficiency in the IaaS layer. In Sect. 4, we propose some feasible ideas or a research roadmap to achieve energy efficiency. Finally, we conclude in Sect. 5.

2 Cloud computing review and motivation

2.1 Cloud computing definition, architecture and its IaaS layer

Up to now, there is no standard definition for cloud computing and the main reason is that cloud computing, unlike other technical terms, is not a new technology, but rather a new operations model that brings together a set of existing technologies to run business in a different way. In this paper, we adopt the definition of cloud computing proposed by NIST [4],

> Cloud computing is 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.

A cloud system delivers infrastructure, platform, and software (applications) as services which are made available to consumers as subscription-based services under the pay-as-you-go model. In industry, these services are referred to as the Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS), respectively.

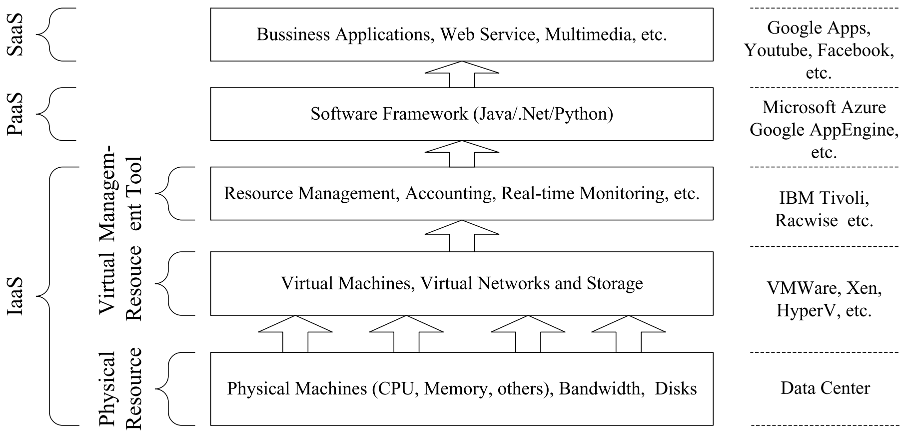

Fig. 1 Architecture of cloud computing

In general, a cloud computing system can be coarsely divided into three layers corresponding to the three concepts, IaaS, PaaS and SaaS (see Fig. 1). The IaaS layer is responsible for managing the physical machines, creating a pool of virtual computation or storage resource by using virtualization technologies, such as Xen [37] and VMWare [38], and providing an elastic resource service to the upper layer. The PaaS layer is built on top of the infrastructure layer, and the platform layer consists of operating systems and application frameworks. The SaaS layer is at the highest level of the hierarchy, the application layer consists of the actual cloud applications. In this paper, we only focus on the IaaS layer.

According to the cloud architecture, the IaaS layer is further composed of three layers, namely, physical resource, virtual resource, and management tool. The physical resource layer consists of traditional data center and it comprises servers, network devices and disks, etc., or non-IT equipment (such as cooling equipment, lightening, air conditioner, etc.). The virtual resource layer holds a pool of unified virtualized computation or storage resources which are transformed from physical resource by using virtualization technique, such as VMWare [38], Xen [37], and Hyper-V [39], etc. Virtualization is one of the biggest differences between a traditional data center and a cloud data center and it makes the resources much easier to access and much more flexible to provide elastic service to the upper layer. The management tool layer is responsible for virtual resource management, accounting, and monitoring, such as OpenNebula [5] and Eucalyptus [6].

2.2 Motivation

The problem is that cloud infrastructure is not only expensive to maintain, but also unfriendly to the environment. High energy costs and huge carbon footprints are incurred due to massive amounts of electricity needed to power and cool numerous servers hosted in data centers. Cloud infrastructure providers need to adopt measures to ensure that their profit margin is not dramatically reduced due to high energy costs. For instance, Google, Microsoft, and Yahoo are building large data centers in barren desert land surrounding Columbia River, USA to exploit cheap and reliable hydroelectric power [7]. There is also increasing pressure from governments worldwide to reduce carbon footprints which have a significant impact on climate change. For example, the Japanese government has established the Japan Data Center Council to address the soaring energy consumption of data centers [8]. Leading computing service providers have also recently formed a global consortium known as The Green Grid [9] to promote energy efficiency for data centers and minimize their environmental impact.

Although Berl et al. [10] have reviewed and introduced some techniques of energy efficiency in cloud computing, our work is much more detailed because we only focus on the IaaS layer. We will study the state-of-the-art techniques and research which can lead to a reduction of energy consumption in the IaaS layer in the next section.

3 Study of the state-of-the-art techniques and research

Recent works have indicated that over 90% of a data center's electricity is consumed by the information technology equipment (namely server, storage, and network) and cooling facility, with the rest dedicated to lights and everything else [11, 12]. In this paper, we study the state-of-art energy conserving techniques related to these components. Specially, we also discuss a processor which plays a critical role in a data center. We believe that this can provide a guideline to later discussions.

3.1 Processor

Current processors have been designed to offer not only high performance but also world-class efficiency. This is due to high energy consumption which not only translates to high power bills for maintenance, but also results in high heat dissipation which decreases the reliability of hardware. Exploiting hardware-level techniques to improve energy efficiency is one of the efficient approaches. For example, modern processors commonly provide ability of slowing down CPU clock speeds (by dynamic voltage scaling technique abbreviated as DVS) or powering off parts of the chips (by dynamic power management technique abbreviated as DPM) to save energy (e.g., Intel SpeedStep and AMD PowerNow!). Some recent processors, like Intel® Xeon® processor and AMD Opteron™ processors, even provide more advanced features like excellent performance per watt, more efficient workload management, virtualization support, and the ability of running in higher temperature environments [13, 14].

On the software-level, ACPI is a well-known standard for unified operating system-centric device configuration and power management [15]. Furthermore, a tremendous amount of research is focused on energy efficient task scheduling. The aim of scheduling is to reduce the power consumption of a processor used for task execution. The power consumption typically includes two parts, namely dynamic power and leakage power [17]. In previous approaches, most of the research exploits the DVS technique to obtain energy saving because dynamic power is the primary contributor to total power dissipation. DVS works because the dynamic power is a strictly convex and increasing function of its operating voltage. Thus, reducing the voltage, which consequently lead to the frequency reduction, provides substantial savings in power at the cost of slower program execution [18].

To the best of our knowledge, theoretical exploration of DVS systems for real-time jobs was first given by Yao et al. [19] who considered a set of aperiodic jobs on an ideal processor. (An ideal processor can operate at any speed in a pre-defined section, and a non-ideal processor has only discrete speeds with negligible or non-negligible speed transition overheads.) After that, a large amount of research on energy efficient task scheduling with various constraints and assumptions was proposed, such as ideal or non-ideal processor [19], hard real-time or soft real-time constraint [21], periodic or sporadic or aperiodic tasks, etc. [20].

Yet other works consider non-DVS peripheral devices in typical systems [22, 23]. For example, Cheng and Goddard consider the fact that waiting for an I/O resource typically prolongs the execution time, thus leading to more energy consumption on devices although some energy saving is obtained because of DVS scheduling over the processor [23]. Hence, the solution should consider the energy consumption of both the processor and devices.

When the CMOS processes go below 100 nm, the leakage power is typically considered non-negligible and the effectiveness of traditional DVS becomes limited [24]. Jejurikar et al. [17] exploit a combined slowdown and shutdown approach to minimize energy consumption. Strategy handoff depends on a critical speed, in which executing any task at any speed less than the critical speed would consume more energy than that at the critical speed. Adaptive body biasing technique (abbreviated as ABB) is also an effective way to reduce leakage power dissipation [24]. For example, Qiu et al. [25] recently proposed an optimal loop scheduling and voltage assignment algorithm to minimize both dynamic and leakage energy via DVS and ABB with the soft real-time constraint.

A multi-processor system and a multi-core system are considered much more energy efficient than the single processor system due to the convex power consumption functions. Unfortunately, multi-processor (or multi-core) energy-efficient scheduling is often NP-hard under various application constraints. Therefore, authors typically design heuristic algorithms or approximation algorithms to solve the problem. For example, Chen et al. propose a series approximation algorithm to solve this problem considering different constraints [29, 30]. In [29], they propose a 1.13-approximation algorithm for multi-processor energy-efficient scheduling by considering a set of frame-based real-time tasks on homogeneous multi-processor systems. In [30], they proposed a 1.283-approximation leakage-ware algorithm for the minimization of energy consumption and the satisfaction of task timing constraints. Most of the existing research can be divided into two categories: partition-based [26] and global scheduling based [27], and the rest follow a hybrid way. For example, Zhu et al. [28] propose a slack-reclamation based scheduling algorithm in a multi-processor embedded system in which a global queue of ready tasks is used for the selection of a candidate such that the slack time, due to the early completion of another task, is used to slow down the execution speed of the selected task.

Although energy efficient scheduling problem has been comprehensively studied for a decade, it is still an ongoing work, and many critical problems are still open. Fortunately, some successful ideas inspire other domains' research, e.g., slowing down the computing speed when the workload is low or powering off parts of the components when they are idle, etc.

3.2 Server

Typically, today's data center hosts tens of thousands of servers to serve millions of worldwide customers. Unfortunately, most of the servers are under-utilized, around 30%, but consume significant amount of energy annually [31]. Moreover, the increasing energy consumption makes the data center suffer from a high power-density problem, and this problem has been exacerbated due to the use of blade servers which occupy less space but consume five times more power than a traditional rack server [32]. The combination of these factors drives the researchers to explore powerful solutions or techniques for power management.

To the best of our knowledge, the most commonly used techniques for servers-level power management mainly include DVS and reconfiguration (i.e., dynamical turn on/off or VOVO).

Pinheiro et al. [34] and Chase et al. [35] concurrently proposed similar strategies for managing energy in the context of server clusters. Pinheiro et al. called the technique Load Concentration (LC). The basic idea behind LC is to dynamically distribute the load offered to a server cluster so that, under light load, some hardware resources can be idled and put in low-power modes, e.g., can be turned off. Like in PDC and MAID (see Sect. 3.3), the key here is that this type of concentration is usually beneficial in terms of overall power consumption. Under heavy load, the inverse operation should be performed, i.e., we should reactivate components and redistribute the load to alleviate or eliminate any performance degradation. Heath et al. [36] further consider the heterogeneity among servers in a cluster-level solution.

Elnozahy et al. [33] evaluated different combinations of cluster reconfiguration and DVS for clusters in which power supplies are more efficient (i.e., power supply losses are relatively low). Their work proposed independent voltage scaling (IVS) and coordinated voltage scaling (CVS). In IVS, each server node makes its own independent decision about what voltage and frequency to use, depending on the load it is receiving. In CVS, nodes coordinate their voltage and frequency settings in order to optimize the overall energy consumption. They showed that CVS conserves slightly more energy than IVS. However, the additional benefits achieved by CVS come at the expense of increased implementation complexity. They also showed that, depending on workload, either CVS or cluster reconfiguration is the best technique in the presence of low supply losses. For the workloads they examined, the combination of these techniques is always the best approach.

Recently, the virtualization technique attracted much attention from academia, industries, and businesses because it can significantly improve the resource utilization, meanwhile providing advantages, e.g., flexibility, reliability, performance isolation [40], etc. The VM technology (such as Xen [37], VMWare [38], and the new Microsoft Hyper-V technology [39], etc.) enables multiple OS environments to coexist on the same physical computer, in strong isolation from each other. Moreover, multiple VMs can be started and stopped on-demand to meet accepted service requests, hence providing maximum flexibility to configure various partitions of resources on the same physical machine to different specific requirements of service requests.

Furthermore, virtual machine migration, which is used to transfer a VM across physical machines, has served as a main approach to achieve better energy efficiency of virtualized data centers. This is because in doing so, server consolidation via VM migrations allows more computers to be turned off. Generally, there are two varieties: regular migration and live migration. The first one moves a VM from one host to another by stopping the originally used server, copying its memory contents, and then resuming it on the destination. The second one performs the same logical functionality but without the need to stop the server domain for the transition. Live migration enables us to perform such migrations within a server cluster without interrupting the services the VMs are providing. It reduces the extra work traditionally involved with moving VMs, including notifying users, shutting down the applications, moving the VMs to new servers, and then restarting the VMs and each of the applications. Live migration is an essential technology for an agile, dynamic data center environment based on server virtualization. For example, an application workload may grow over time due to changing business requirements. As a result, the memory or computing resources of the physical server hosting the workload may become constrained. When this happens, additional capacity can made available by migrating the virtual machine (VM) containing the application to a less-utilized server [37].

By taking advantage of VM, a server or a server cluster can be much more energy efficient. An intuitive method is that the "size" of a virtual machine can be dynamically reconfigured according to service workload (termed dynamical resource allocation). For example, Stoess et al. [41] design a novel two-level framework for energy management which targets hypervisor-based virtual machine systems. On the one hand, a host-level subsystem embedded in hypervisor is responsible for machine-wide energy control; on the other hand, a guest-level built-in guest OS is responsible for fine-grained application-specific (or service) energy management.

A combination of dynamical resource allocation and DVS can lead to much more efficient results. Most of the DVS-based works adopt a coarser power model which comprises two parts, namely fixed power and dynamic power, without leakage power consideration. The reason is that even if the processor scales down to the minimal power consumption, other components are still working.

Kim and Buyya et al. [42] explore the problem of power-aware allocation of VMs in DVS-enabled cloud data centers for application services based on user QoS requirements such as the deadline and budget constraints. In their research, the system is composed of a set of cloud data centers. When a user submits a request to the system, a global broker would ask each data center to calculate the price of provisioning a resource (i.e., VM). Then the broker chooses the data center which provides the lowest price to provide the service. Three policies are proposed for scheduling real-time VMs to reduce energy consumption, while meeting deadline constraints and maximizing the acceptance rate of provisioning requests, called Lowest-DVS, δ-Advanced-DVS, and Adaptive-DVS, respectively. The first one adjusts the CPU's P-state to the lowest level, while ensuring that all the VMs do not violate their deadlines. The δ-Advanced-DVS policy over-scales the CPU speed up to $\delta\%$ to increase the acceptance rate. The Adaptive-DVS policy uses a queuing model to calculate the optimal CPU speed if the arrival rate and service time of real-time VMs can be estimated in advance. The results show that the δ-Advanced-DVS policy obtains the best performance in terms of profit per unit of the consumed power, as the CPU performance is automatically adjusted according to the system load. The performance of the Adaptive-DVS is limited by the simplified queuing model.

In [44], Laszewski et al. proposed a power-aware algorithm for scheduling a virtual machine in a DVS-enabled computing cluster. They regard the virtual machine as a group of requests and model it in terms of required processor frequency and required executing time, just as most of the power-ware tasks scheduling works, and schedule VM from high frequency server to that which frequency is low to achieve load balance. However, these works only consider the dynamic power and ignore the fixed energy consumption.

Recent research considers further reducing the fixed energy consumption by dynamically consolidating VMs onto fewer servers and then turning off the idle ones. For example, Younge et al. [45] indicate that in a multi-core environment, the power consumption curve illustrates that as the number of processing cores increases, the amount of energy used does not increase proportionally. In fact, that changes as power consumption decreases. In another word, in such an environment, allocating virtual machines on as few as possible physical machines and turning off the others consumes less energy than when equally allocating these virtual machines to all the physical machines. Rodero et al. [46] present an energy-aware online provisioning approach for HPC applications on consolidated and virtualized computing platforms. Energy efficiency is achieved by using a workload-aware, just-right dynamic provisioning mechanism and the ability to power down subsystems of a host system that are not required by the VMs mapped to it.

Kusic et al. [43] implement and validate a dynamic resource provisioning framework for virtualized server environments wherein the provisioning problem is posed as one of sequential optimization under uncertainty and solved using a lookahead control scheme. Their objective is to maximize the service provider's profit by minimizing both the energy consumption and SLA violation. Switching servers on/off and resizing and dynamically consolidating VMs via VM migration are applied as power saving mechanisms. Two key challenges are considered in the model: (i) quickly changing the workload and (ii) the cost for switching hosts and VMs on/off. However, DVS is not considered in their model and the proposed model requires simulation-based learning for the application specific adjustments. Moreover, due to the complexity of the model, the optimization execution time reaches 30 minutes even for a small experimental setup (15 servers), which is not suitable for large-scale real-world systems.

Meisner et al. [47] propose an approach for power conservation in server systems based on fast transitions between active and low power states. Different from the above research, their approach only needs two power states (sleep and fully active states), and reduces the energy consumption by putting the idle servers into the sleep state. Experiments show a key result that if the transition time is less than 10 ms, power savings are approximately linear to the utilization for all workloads. However, the problem is that the current level of technology cannot satisfy such a requirement.

A recent work which is proposed by Lee and Zomaya in [48] suggested two energy-conscious task consolidation heuristics which aim at maximizing resource utilization and explicitly take into account both active and idle energy consumption.

Their work is based on a linear power model proposed in [49] and the problem is regarded as various Bin-Packing problems. However, this work only focuses on the resource utilization and ignores many non-ignorable factors.

According to our studies, there are several major challenges in energy efficient server management:

(a) Optimal resource allocation policies for specific allocations. Generally, a system is asked to satisfy a certain QoS or keep SLA violation below a given threshold; this is not easy because it needs prior knowledge about allocation's activities which is learned from historical information to estimate the resource requirement in the future. Unfortunately, this is not precise, consequently leading to unexpected results.
(b) Operation overhead must be considered in decision making, such as VM migration and power state transition, since these operations will lead to extra resource and time overhead. Moreover, it may impact the performance of shared services. It is only beneficial when the extra processing unit time incurred for operation is significantly less than the amount of processing unit time that operation makes available.
(c) The solution must be scalable since the size of data center continuously increases.
(d) ...

Although the above techniques can greatly reduce energy consumption, they are still not globally optimal because they ignore other facilities in a data center, such as cooling systems and network devices, which also consume significant amount of energy. In Sects. 3.4 and 3.5, we continue to discuss some more comprehensive LC (or consolidation) solutions by considering cooling and the network, respectively.

3.3 Storage

With the development of Internet and information technologies in recent years, the storage requirement in a data center explosively grows, and the storage devices contribute to a significant fraction of the overall power budget. Moreover, greater energy consumption leads to more heat dissipation which not only impacts the hardware reliability but also in turn leads to greater cooling requirements. The combined effect also limits the density of server racks. The lower density of servers leads to more space requirements, thus higher operating costs. Therefore, seeking energy conserving techniques for storage systems becomes very important.

To the best of our knowledge, techniques for conserving storage energy have been comprehensively explored for more than ten years, and the research object mainly includes a single-disk-based mobile device (e.g., laptop) and a multiple-disks-based storage system in a data center [60]. In this work, we focus on the latter.

Any disk power management scheme essentially attempts to exploit one fact: disks can be run in high power mode or low-power mode, with a corresponding performance tradeoff. In the limit, a disk can be turned off so that it consumes no power. Given a large cluster of disks, only a fraction of them is accessed at any time, so that the rest could potentially be switched to a low-power mode. However, since mode transitions consume time and power, disk management schemes have to walk the tightrope of finding the right balance between power consumption and performance. According to our recent studies, the solution space explored thus far in the literature can be divided as follows: (i) Hardware-based solutions, (ii) Disk-Array Management solutions, and (iii) Caching solutions. Each of these solutions proposes a new system of some kind: hardware-based solutions propose novel storage hierarchies to strike the right balance between performance and power consumption; disk-array management solutions interject a new 'management layer' on top of the file system, which controls disk-array configuration and data layout to achieve power-optimal disk access patterns; caching solutions devise new power-aware caching algorithms that allow large fractions of the storage system to remain idle for longer periods of time, allowing them to be switched to lower power modes [67]. Next, we discuss part of the state-of-art researches.

1. *Hardware-based solutions*

The concept of a memory hierarchy arose as a result of the natural tradeoff between memory speed and memory cost. Carrera et al. point out in [53] that there exists a similar tradeoff between performance and power-consumption among high-performance disks and low-performance disks, such as laptop disks. They explore the possibility of setting up a disk hierarchy by using high- and low-performance disks in conjunction with each other.

In a related vein, Gurumurthi et al. [61] propose Dynamic Rotations Per-Minute (DRPM) technology, whereby disks can be run at multiple speeds depending on whether power or performance takes precedence. DRPM use changes in the average response time and the length of the disk request queue to drive dynamic disk-speed transitions. This idea is similar to the DVS technique. However, it poses a significant engineering challenge whose feasibility is far from obvious.

The MAID has been proposed as a replacement for old tape backup archives with hundreds or thousands of tapes [50]. Because only a small part of the archive would be active at a time, the idea is to copy the required data to a set of "cache disks" and spin down all the other disks. All accesses to the archive should then check the cache disk(s) first. Cache disk replacements are implemented using an LRU policy. Replaced data can simply be discarded if it is clean, since its original copy is still stored on one of the non-cache disks. Replaced data that is dirty has to be written back to the corresponding non-cache disk.

2. *Disk-array management solutions*

RAID techniques are a long-standing solution for improving disk array performance and reliability. Unfortunately, conventional server-class RAID cannot easily reduce power because loads are balanced such that they require the use of all the disks in the array for even light loads. Thus the intuitive idea of trading off availability in favor of energy by powering down "unnecessary" disk drives in disk arrays results in little benefit. More success has been achieved with schemes that concentrate disk array workloads onto a subset of their disks so that the other disks can stay in low power modes. We discuss five approaches here, including data reorganization, layout optimization, replication-based, power-aware scheduling, and a hybrid scheme.

Pinheiro and Bianchini [54] suggest that if data is laid out on disks according to the frequency of access, with the most popular files being located in one set of disks, and the least popular ones in another, then the latter set of disks could be spun down to conserve energy. Their scheme is called Popular Data Concentration (PDC) and they implement and evaluate a prototype file server called Nomad FS which runs on top of the file system and monitors data layout on disks. The difference between PDC and MAID is that the former relies on file popularity and migration whereas the latter relies on temporal locality and copying.

Son et al. [51] propose another data layout management scheme to optimize disk access patterns. Their approach uses finer-grained control over data layout on a disk, tuning it on a per-application basis. Applications are instrumented and then profiled to obtain array access sequences, which their system then uses to determine optimal disk layouts by computing optimal stripe factor, stripe size, start disk, etc. But the wisdom of marrying the disk layout to the application seems questionable.

Many modern file systems such as HDFS (Hadoop Distributed File System) use replication for fault tolerance and data recovery. This replication can be used to save energy by accessing data copies from spinning disks while transitioning disks that contain redundant data to standby [55]. Weddle et al. [57] proposed a power-aware RAID management, called PARAID, which suggests replication in a RAID group. Instead of running all disks in a RAID group, PARAID determines an appropriate gear level, i.e., how many disks in a RAID group are spinning vs. disks in the sleep mode. This decision is based on the current system load (derived from disk utilization). Based on a gear level, disks in the RAID system are spun down (when the gear goes down) or spun up (when the gear goes up). PARAID provides skewed data replication, so that it can continue service using only a subset of the disks in each RAID group. To meet the massive data requirement in a today's data center, this work has been extended to manage power in large data centers with multiple RAID groups [59]. Huang et al. [56] proposed FS2 to utilize free space on the disk to make replicas for hot data blocks at runtime to improve performance and save energy for the storage system. Their scheme trades disk space for performance and energy consumption. Therefore, its effectiveness is circumscribed by the available free space and the access locality in the running environment.

Chou et al. [58] recently developed a power-aware scheduling algorithm that takes a stream of requests as input and determines the disk location for serving each request to minimize energy consumption based on a given replicas about data placement. This method has no interference with any data placement scheme and is designed to work with any power management scheme in which a disk simply gets spun down after experiencing a fixed idleness threshold.

Additionally, some researchers adopt a hybrid way, such as Hibernator, proposed by Zhu et al. in [64], combining a number of ideas to conserve energy in a storage system. It assumes multi-speed disks and computes online the optimal speed that each disk should run at. To minimize speed transition overheads, disks maintain their speeds for a fixed (long) period of time—the authors call this the coarse-grained approach. Hibernator includes a file server that sits on top of the file system and manipulates data layout to put the most-accessed data on the highest speed disks. The authors address the issue of performance guarantees by stipulating that if performance drops below some threshold, then all disks are spun up to their highest speed.

3. *Caching solutions*

Not all accesses to a storage system go to disks. Many modern storage systems use a large storage cache to reduce the number of disk accesses and improve performance. For example, the EMC Symmetrix storage system with a capacity of 10–50 TB can be configured with up-to 128 GB of non-volatile memory as the storage cache [66]. Different from those small (usually 1–4 MB) buffers on an SCSI disk, which are mainly used for read-ahead purposes, these large caches are used to cache blocks for future accesses. Therefore, the cache replacement algorithm plays a very important role in a storage system.

The storage cache management policy influences the sequence of requests that access disks. Different cache management policies may generate different disk request sequences, which directly impacts disk energy consumption. In other words, by changing the cache management scheme, it is possible to change the average idle time between disk requests, thus providing more opportunities for the disk power management scheme to save energy. For example, if the cache replacement algorithm can selectively keep some blocks from a particular disk in the cache (without significantly increasing the number of misses to other disks), that disk can stay in a low power mode longer [65].

Zhu et al. observe that the storage cache management policy is pivotal in determining the sequence of requests that access disks. The authors present both offline and online power-aware cache replacement algorithms to optimize read accesses, called PA-LRU [63]. They also show through experiments the somewhat obvious fact that for write accesses, write-back policies offer more opportunities to save power than write-through policies. Furthermore, they develop power-aware cache partition scheme, called PB-LRU, which divides the storage cache into separate partitions, one for each disk, and dynamically re-configures the size of each partition by estimating the correlation between disk's energy consumption and its corresponding storage cache at run time [62].

A write off-loading technique has been proposed recently which is directed towards minimizing energy consumed due to write requests [52]. Newly written data is diverted to disks which are currently spinning (anywhere in the data center) thus enabling disks in standby mode to remain in their low energy mode for longer periods.

3.4 Cooling

Thermal management plays an important role in today's data center since (i) the power density of a data center increases rapidly, which might lead to a greater probability of thermal failure, impacting the availability of these systems. This problem is likely to be exacerbated by recent trends towards consolidation in data centers and adoption of higher-density computer systems, such as blade servers. And (ii) cooling power amounts to a significant fraction of data center's operating cost. Although the industry has recognized this problem, it is still not easy to optimally solve it because it touches upon several domains, including thermodynamics, aerodynamics, etc. Therefore, we will review a typical data center layout at first, and this can help us better understand the state-of-the-art research of thermal management.

In this section, we focus on the state-of-art research of energy efficient thermal management in data centers. Before that, we discuss a typical data center layout.

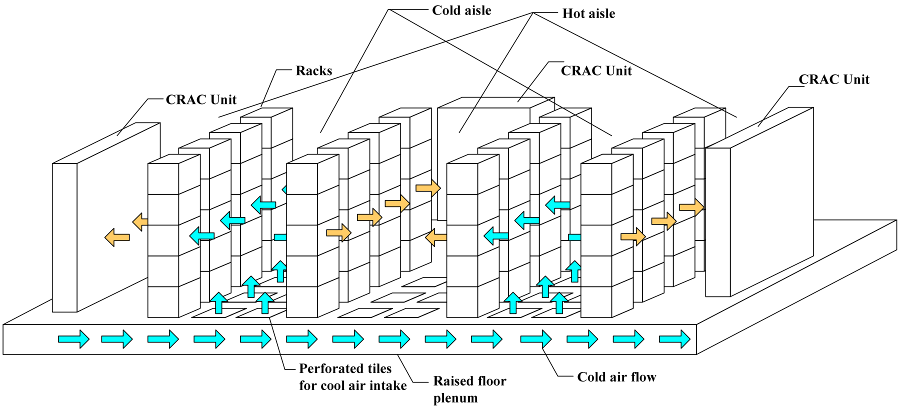

Fig. 2 Typical data center layout

3.4.1 Layout of a typical data center

A typical data center is shown in Fig. 2. Accurately, data centers come in all shapes and sizes and will always have a mixture of equipment and heat loads. So servers are installed in racks. A rack is a standardized metal frame or enclosure which is 19 inches wide and usually 42 units (U) tall. One unit refers to the height of 1.75 inches and most commodity servers are 1 or 2 U high. In a typical scenario, the server racks are placed on a raised floor area which allows the installation of cabling and flow of cool air under the hardware.

Maintenance of internal server temperatures and relative humidity within operating tolerances is a critical issue for data centers. Current practice is to mount server racks on a raised floor plenum area. Computer room air-conditioning (CRAC) units blow cold air into the plenum, pressurizing the space under the racks and forcing cool air up through perforated floor tiles. Server cooling fans then circulate this air through the rack cabinets, and exhaust it back to the room where air handling units draw off the waste heat.

A key concern which must be emphasized is the heat recirculation which plays a significant role in a data center's energy efficiency [73, 75]. As computing devices in a data center emit heat by running tasks, the cooling system must supply cold air to their air inlets at a temperature below their redline temperature. However, the recirculation of hot air from the air outlets of the IT equipment back into their air inlets increases the inlet temperatures (namely from hot aisle to cold aisle) and can cause the appearance of hot spots. Heat recirculation forces data center operators to operate their CRACs to supply cold air at a much lower temperature than the predefined redline.

3.4.2 Schemes for cooling energy saving

According to our studies, the schemes for reducing data center's cooling energy consumption can be divided into two groups: facility level and system level.

In the facility level, researchers seek to optimize the solution and redesign a data center to increase the cooling efficiency, thus reducing the energy consumed by cooling. Current works focus on optimizing the flow of hot and cold air in the data center and evaluate the layout of the computing equipment in the data center to minimize air flow inefficiencies (e.g., hot aisles and cold aisles) or design intelligent system controllers to improve cold air delivery. For example, Brunschwiler et al. [68] have tested hot water data center cooling by directly reusing the generated thermal energy in neighborhood heating systems. Their aim is to design a zero-emission data center. Hamann et al. [69] optimize the data center layout by using a mobile measurement technology (MMT) to achieve higher cooling efficiency.

Taking advantage of the environment can also significantly reduce the data center energy consumption. For example, in geographical locations where the outside temperature is less that 13°C for at least four months in a year, cooling efficiency can be considerably improved by taking advantage of the outside environment. It is no coincidence that Google decided to launch its newest data center in Finland.

In the system level, the most commonly used method is to optimally place workload across the servers to minimize the cooling requirement, e.g., temperature-aware workload placement [70, 72]. The goals of any temperature-aware workload placement are to (i) prevent server inlet temperatures from crossing a pre-defined redline, and (ii) maximize the temperature of the air the CRAC units pump into the data center, increasing their operating efficiency.

Moore and Chase et al. were amongst the earliest researchers to advocate conserving cooling energy based on such a method [74, 75]. The idea behind their work is to predict the thermal effects of a task placement, thus selecting a placement with best effects. In [75], two interesting workload placement policies are proposed, called Zone-Based Discretization (ZBD) and Minimizing Heat Recirculation (MinHR), respectively. The first way observes that because cooling is invariably uneven, even identical servers under equal loading reach different temperatures. For instance, air flow through the middle of the aisles between machine racks is better than that near the ends of the aisles, making the middle cooler than the ends. Therefore, uniformly distributing the loading over all the servers results in hot spots. Instead, the policy reduces cooling power by distributing the loading over all the servers as the inverse of the server temperature, thus achieving uniform temperature. The second way is designed to minimize the amount of heat that re-circulates within a data center based on a Heat Recirculation Factor (HRF) metric.

Tang et al. present a similar purpose work in [73]. The difference is that the authors apply a cyber-physical method to analyze the system and mathematically formalize the problem of minimizing the data center cooling cost as the problem of minimizing the maximal (peak) inlet temperature through task assignment.

Recently, a paradox has been proposed when data center operators prepare to manage the server power and cooling power at the same time [70–72]. On the one hand, data center operators want to increase the resource utilization and reduce energy consumption by consolidating workload to a subset of the servers and turning off the rest. Consequently, the active servers reach higher temperatures creating hot spots, even if the servers are chosen to be physically distributed across the data center. Because cooling effort increases with higher temperature even if the amount of heat removed is the same, this power management policy increases cooling power. On the other hand, the thermal management policies advocate placing workload across the servers to minimize the generating heat and avoid turnoff servers which lead to capacity decrease. These policies lead to a large amount of idle power. Therefore, seeking a balance between the idle and cooling power (i.e., power management and thermal management) becomes a popular issue now. For example, Ahmad and Vijaykumar [70] propose a joint optimization technique, called PowerTrade and SurgeGuard, to calculate the optimal tradeoff between server idle power and cooling power in a subset of servers to reduce the sum of the idle and cooling powers.

3.5 Network

Although energy saving techniques in wireless sensor networks have already been studied in detail [76, 77] because of the specific needs of battery-powered networks, the energy saving of wire-line networks have not received much attention until very recently (e.g., IEEE 802.3az). The reasons behind the change of mind are that (i) energy saving in a data center cannot be further improved by solely relying on servers and cooling, (ii) the energy consumed by network devices in a today's data center (e.g., routers, switchers, NICs, etc.) is non-trivial since the size of data centers and their network infrastructure are exploding; moreover, higher capacity network equipment becomes more power-hungry and requires greater amounts of cooling power.

In this section, we focus on the state-of-art research of energy efficient network management in large scale data centers. Before that, we discuss data center network architectures.

3.5.1 Data center network architecture

Today's data centers typically host over tens of thousands of servers and serve millions customers globally. The traditional 2-tier network architecture cannot satisfy such a requirement for it only supports not more than 5000 servers [78]. The inflexibility and insufficient bisection bandwidth prompt researchers to explore alternative much more advanced network architectures. However, all of these are only recent research proposals and have not been tested in a real-world data center. The most widely adopted network architecture is still the 3-tier topology (e.g., fat tree in Fig. 3), in which the network comprises three levels: the access level, aggregation level, and core level. Generally, the link capacity between the core level and aggregation level is much higher than that between the aggregation level and access level, e.g., 10 GE vs. 1 GE. This configuration can reduce the bottleneck in the network and improve the overall performance. Additionally, the network size cannot be infinitely extended and it depends on the used multi-path routing technology. For example, Equal Cost Multi-Path (ECMP) routing only allows eight paths, thus the maximal number of core switches is eight. More detailed analyses can be found in [78, 79, 85].

3.5.2 Schemes for networks energy saving

The energy saving of networks becomes possible due to two factors. First, networks are provisioned for the worst-case or busy-hour load, and this load typically exceeds

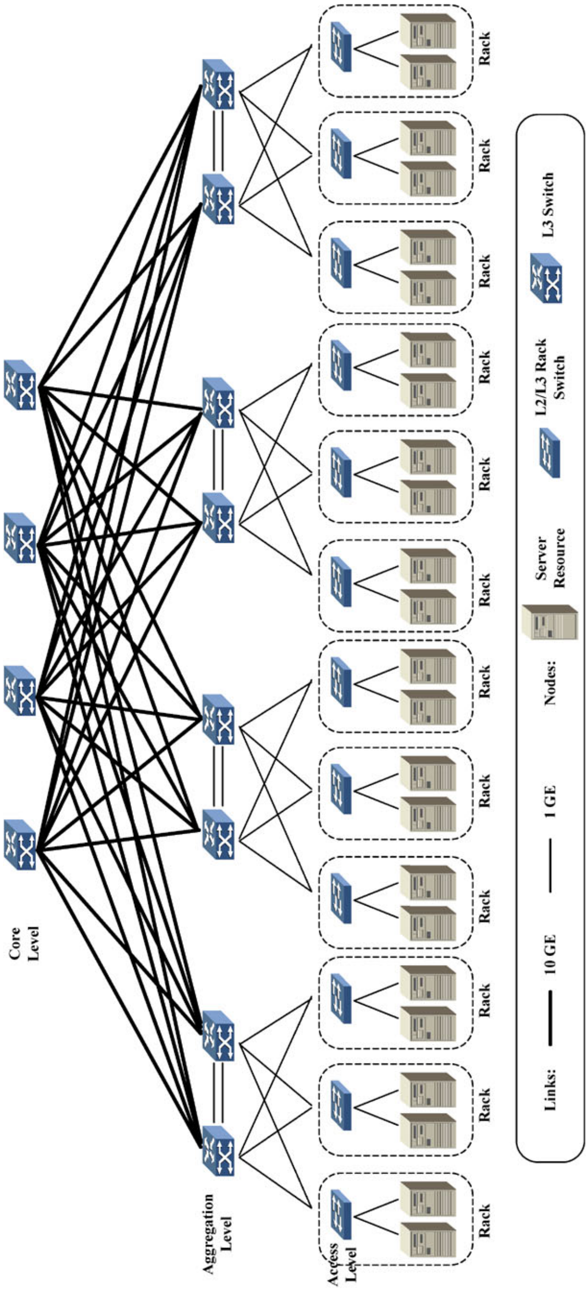

Fig. 3 Typical data center network architecture

their long-term utilization by a wide margin. Second, the energy consumption of network equipment remains substantial even when the network is idle.

According to our recent studies, there are three main approaches to achieve this aim. The first way is Network Traffic Consolidation (abbreviated as NTC). The NTC adopts a traffic engineering approach to route traffic such that it is consolidated on fewer links (and switches), while some of the non-utilized links (and switches) can be switched off. Gupta and Singh were amongst the earliest researchers who advocated this technique to conserve network energy [83]. The thesis of their paper is that components in network devices can be put to sleep (or into energy saving modes) with some changes to Internet protocols. Moreover, they explored the feasibility of this idea in a LAN setting in later papers [86, 87], which is focused on deciding when to sleep and how long to sleep.

The second way is called Server Load Consolidation (abbreviated as SLC). The SLC adopts a job scheduling technique to consolidate jobs onto fewer servers, thus requiring fewer links (and switches). For example, Kliazovich et al. [78] propose a work which underlines the role of communication fabric. The work firstly analyzes the influence of network fabric on performance under typical three-tier architecture (i.e., fat tree) and indicates that, under full load, available bandwidth for the individual computing servers is just a quarter of the theoretical value. Then they propose a Data center Energy-efficient Network-aware Scheduling algorithm (termed DENS) which considers energy consumption of both the network devices and servers. The aim of scheduling is to avoid hotspots within a data center while minimizing the number of computing servers required for job execution. The proposed methodology is implemented in their self-developed "GreenCloud" simulator [79].

The third approach to reduce energy consumption in existing networks is dynamically changing the link speed, called Link State Adaptation (abbreviated as LSA). This methodology is based on the fact that Ethernet power consumption is independent of link utilization but related to the data transmission rate. For example, current research shows that 1 Gbps Ethernet links consume about 4 W more than 100 Mbps links, and a 10 Gbps Ethernet link consumes even more—from 10 to 20 W [88]. The principle may be the same with the DVS technique. A proposal about this technique was recently made to the IEEE 802.3 committee outlining a scheme, called Ethernet Adaptive Link Rate (ALR), to automatically tune the link speed from 1 Gbps to 100 Mbps or from 100 Mbps to 10 Mbps in order to save energy during low periods of activity [16]. Gunaratne et al. [81] apply ALR to vary the link data rate in response to utilization to reduce the energy consumption of a typical Ethernet link.

Mahadevan et al. [80] test three schemes, namely LSA, NTC, and SLC, by simulating a real Web 2.0 workload in a real data center network topology. The results show that both the NTC and SLC can obtain significant network energy savings (about 75%) but incur a performance penalty. They also incorporate service level awareness (SL-aware) in the schemes to provide much more redundant path and then test the tradeoff between performance and energy savings.

Furthermore, some works explore combinations of these methods which seem to obtain better results, such as [82, 84]. The key issue in this research domain is the tradeoff among network energy savings, performance of a system, and robustness (or reliability).

4 Towards green cloud computing

Up to now, we have discussed a large amount of state-of-art research and techniques on how to improve energy efficiency in a data center from different perspectives. Based on this knowledge, we try to analyze deeper to obtain a much more comprehensive solution to implement overall energy savings in a cloud infrastructure. A major challenge is that we must consider the relations among system components and the tradeoffs that can result in the optimal balance between performance, QoS, and energy consumption and include self-aware runtime adaptation. In principle, we suggest designing a management system based on current hardware and software, without changing the existing systems because it makes the solution complex; moreover, it is too expensive to implement (e.g., due to time cost and human cost). Due to this principle, the processor will not appear in the later analysis because it needs redesigning the hardware and operating system.

The foundational technology for cloud computing is virtualization. This idea has been widely accepted by academia and industry. Virtual machines that encapsulate virtualized services can be moved, copied, created, and deleted depending on management decisions. This provides great opportunities to improve energy efficiency in a data center. We list three complementary steps which are very novel to achieve the purpose:

Step 1: Energy-aware dynamic resource allocation

VM provides strong performance isolation to the encapsulating services. When a VM is created for the first time, an intelligent system will allocate resource to it based on predefined mechanisms. In fact, the workload of service changes all the time. A static configuration will lead to energy inefficiency if the workload is far below the allocated resource or to the SLA violation if the workload exceeds redline. Therefore, our first solution is dynamically reconfiguring the VM based on online monitoring performance and workload prediction. It includes two levels. The first is the machine-level which is responsible for scaling the entire resource by DVS in terms of total workload. The second is the application-level which is responsible for individual application (or service) resource management (namely adjusting the proportion of resource allocation among several VMs) which is controlled by a VM hypervisor.

Step 2: Thermal-aware energy efficient load balancing

Obviously, the first step is focused on a single server; hence it is not optimal from a data center's perspective. An intuitive way to achieve global energy efficiency is load balancing and just-in-size resource provision since the server's dynamic power is a convex function of its frequency. Most of the research adopts absolute balancing in a homogeneous environment to achieve minimal energy consumption (in a heterogeneous environment, it depends on the respective resource capacity). Unfortunately, this type of strategy is inefficient because the researchers only consider the server and ignore the cooling power. In [75], the authors have indicated that identical servers under equal loading reach different temperatures. For instance, air flow through the middle of the aisles between machine racks is better than that near the ends of the aisles, making the middle cooler than the ends. Therefore, uniformly distributing the loading over all servers is not an optimal way. Our second solution is thermal-aware energy efficient load balancing. Load balancing occurs when deciding to start/stop a VM or perform a VM migration. The former is determined by application (or service) workload variation and needs to compute the best placement. The latter is triggered if the workload exceeds a predefined threshold considering both temperature and resource capacity. It needs to compute the optimal strategy about which service to migrate to which server.

Step 3: Efficient consolidation of VMs considering overall energy

Step 2 is efficient when the load is heavy. If the load is light, consolidation has been proven to be a more efficient way. In our opinion, only focusing on resource utilization maximization will lead to inefficiency. As we know, servers cooperate with networks and the cooling system. Considering the network topology, we can consolidate workload onto a subset of servers that require minimal network supporting. Thus we cannot only turn off the idle servers but also idle network links (and switches). Furthermore, we can consider the location of the active servers and network switches to further reduce the required cooling power. From another perspective, VMs often communicate between each other, establishing virtual network topologies. However, due to VM migrations or non-optimized consolidation, the communicating VMs may end up hosted on logically distant physical nodes providing costly data transfer between each other. If the communicating VMs are allocated to the hosts in different racks or enclosures, the network communication may involve network switches that consume a significant amount of power. On the contrary, this part of network traffic would be removed thus the switch's ports can be tuned to lower speed. Therefore, it is necessary to observe the communication between VMs and place them on the same or closely located nodes. Our third solution is implementing the VMs consolidation by an integrating way. It needs to consider two different relations in a decision model. The first one is the relation among the servers, network fabric, and cooling system. The second is the relation among VMs (namely applications or services).

Additionally, the ALR technique can be as a supplement to the above solutions because it can further reduce the energy consumed by network devices [81, 84].

Besides above analysis, we also consider the storage system in cloud computing. We focus on large scale and distributed storage systems which are architected with built-in redundancy to address requirements like availability, performance, and reliability. The ideal solution is seamlessly integrating hardware architecture design, disk-array based technique, and power-aware cache management. Unfortunately, it is a pity that most of successful solutions or techniques need to redesign the existing system, such as page replacement algorithm in cache, data placement function in disks, redundancy scheme, etc. Here we only advocate the redundancy-based solution. Redundancy is typically implemented by replicating the "original data" (as in mirrored disk arrays, cluster-based storage systems, or wide-area storage systems) or by storing additional information that can be used to reconstruct the original data in case of disk failures (as in RAID 0–5 or erasure-code-based wide-area storage systems). In our opinion, fully leveraging this redundancy can conserve significant storage energy without performance degradation. The basic approach is also turning off a subset of the disks or nodes in the system during low utilization, yet we require that each data item remains accessible at all times. For example, a possible way is proposed by Pinheiro et al. [55] in which a Diverted Accesses technique is applied to segregate original and redundant data on different disks. It allows the system to concentrate the requests on the original disks, leaving the redundant disks idle. The idle disks then are sent into low power mode. Furthermore, the write off-loading technique as a supplement is also effective [52]. We also find that none of the works consider the disk selection problem which impacts the required cooling power, and this provides opportunity to further improve the energy efficiency.

Finally, some supporting elements are also important. For example, an efficient data center's layout can greatly improve solution's efficiency; thermal sensors can add thermal management ability and make the decision model much more comprehensive, etc.

5 Conclusion

This work is focused on a detailed study of existing energy efficient techniques for building a green cloud computing system. In this paper, we review the cloud computing core concepts and study available energy efficient techniques at the IaaS layer of cloud computing, which can reduce energy consumption to a certain degree. At the end, we make a deeper analysis and propose a set of feasible solutions. It not only gives the infrastructure providers a guideline to develop a strong, competitive cloud computing industry, but also helps to reduce CO_2 emission from a cloud data center.

Acknowledgements This work was supported by the National High-tech Research and Development Program of China (Grant No. 2008AA04A107); the key fields Guangdong–Hongkong Technology Cooperation Funding (Grant No. 2009498B21).

References

1. Buyya R, Yeo CS, Venugopa S (2008) Market-oriented cloud computing: vision, hype, and reality for delivering it services as computing utilities. In: 10th IEEE international conference on high performance computing and communications (HPCC'08), Dalian, China, 2008, pp 5–13
2. Zhang Q, Cheng L, Boutaba R (2010) Cloud computing: state-of-the-art and research challenges. J Internet Serv Appl 1(1):7–18
3. Hamilton J (2009) Cooperative expendable micro-slice servers (CEMS): low cost, low power servers for internet-scale services. In: Proceedings of CIDR'09, California, USA
4. NIST Definition of Cloud Computing v15. http://csrc.nist.gov/groups/SNS/cloud-computing/cloud-def-v15.doc
5. Opennebula. http://www.opennebula.org/
6. Eucalyptus. www.eucalyptus.com
7. Markoff J, Hansell S (2006) Hiding in plain sight, google seeks more power. New York Times, 14 June 2006
8. Ministry of economy, trade and industry, government of Japan (2008) Establishment of the Japan Data Center Council, Press Release
9. The Green Grid Consortium. http://www.thegreengrid.org
10. Berl A, Gelenbe E, Di Girolamo M et al (2009) Energy-efficient cloud computing. Comput J 53(7):1045–1051

11. Sweeney J, Bradfield J (2008) Reducing data center's power and energy consumption: saving money and go 'Green'. White paper
12. Sawyer R (2004) Calculating total power requirements for data centers. White paper
13. Intel Corporation (2009) Why the Intel® Xeon® processor 5500 series is the ideal foundation for cloud computing. White paper
14. Farre T, Bulkeley (2009) Energy-efficient servers: a bright spot in the cloud. AMD online
15. ACPI Standard. http://www.acpi.info/
16. IEEE 802.3 Standard (2006) http://standards.ieee.org/getieee802/802.3.html
17. Jejurikar R, Pereira C, Gupta RK (2004) Leakage aware dynamic voltage scaling for real-time embedded systems. In: Proceedings of the 41st design automation conference (DAC'04), San Diego, USA, pp 275–280
18. Benini L, Bogliolo A, Micheli GD (2000) A survey of design techniques for system-level dynamic power management. IEEE Trans Very Large Scale Integr (VLSI) 8(3):299–316
19. Yao F, Demers A, Shenker S (1995) A scheduling model for reduced CPU energy. In: Proceedings of the 36th annual symposium on foundations of computer science (FOCS'95), Milwaukee, Wisconsin, pp 374–382
20. Scordino C, Lipari G (2006) A resource reservation algorithm for power-aware scheduling of periodic and aperiodic real-time tasks. IEEE Trans Comput 55(12):1509–1522
21. Qiu M, Sha EH-M (2009) Cost minimization while satisfying hard/soft timing constraints for heterogeneous embedded systems. ACM Trans Des Autom Electron Syst 14(2):1–30
22. Cheng H, Goddard S (2006) Online energy-aware i/o device scheduling for hard real-time systems. In: Proceedings of design, automation and test in Europe (DATE'06), Munich, Germany, pp 1055–1060
23. Aydin H, Devadas V, Zhu D (2006) System-level energy management for periodic real-time tasks. In: Proceedings of the 27th IEEE real-time systems symposium (RTSS'06), Rio de Janeiro, pp 313–322
24. Yan L, Luo J, Jha NK (2005) Joint dynamic voltage scaling and adaptive body biasing for heterogeneous distributed real-time embedded systems. IEEE Trans Comput-Aided Des Integr Circuits Syst 24(7):1030–1041
25. Qiu M, Yang LT, Shao Z et al (2010) Dynamic and leakage energy minimization with soft real-time loop scheduling and voltage assignment. IEEE Trans Very Large Scale Integr (VLSI) Syst 18(3):501–504
26. Aydin H, Qi Y (2003) Energy-aware partitioning for multiprocessor real-time systems. In: Proceedings of the 17th international parallel and distributed processing symposium, 113b
27. Baruah SK (2004) Optimal utilization bounds for the fixed-priority scheduling of periodic task systems on identical multiprocessors. IEEE Trans Comput 53(6):781–784
28. Zhu D, Melhem R, Childers BR (2003) Scheduling with dynamic voltage/speed adjustment using slack reclamation in multiprocessor real time systems. IEEE Trans Parallel Distrib Syst 14(7):686–699
29. Chen JJ, Hsu HR, Chuang KH et al (2004) Multiprocessor energy-efficient scheduling with task migration considerations. In: 16th EuroMicro conference on real-time systems (ECRTS'04), pp 101–108
30. Chen JJ, Hsu HR, Kuo TW (2006) Leakage-aware energy-efficient scheduling of real-time tasks in multiprocessor systems. In: 12th IEEE real-time and embedded technology and applications symposium (RTAS'06), pp 408–417
31. Barroso LA, Holzle U (2007) The case for energy-proportional computing. Computer 41(12):33–37
32. Ranganathan P, Leech P, Irwin D et al (2006) Ensemble-level power management for dense blade. In: Proceedings of the 33rd annual international symposium on computer architecture (ISCA'06), Boston, USA, pp 66–77
33. Elnozahy EN, Kistler M, Rajamony R (2002) Energy-efficient server clusters. In: Proceedings of the 2nd workshop on power-aware computing systems, pp 179–197
34. Pinheiro E, Bianchini R, Carrera EV et al (2001) Load balancing and unbalancing for power and performance in cluster-based systems. In: Proceedings of the international workshop on compilers and operating systems for low power
35. Chase JS, Anderson DC, Thakar PN et al (2001) Managing energy and server resources in hosting centers. In: Proceedings of the 18th ACM symposium on operating systems principles (SOSP'01), Banff, Canada, pp 103–116
36. Heath T, Diniz B, Carrera EV et al (2005) Energy conservation in heterogeneous server clusters. In: Proceedings of the 10th ACM SIGPLAN symposium on principles and practice of parallel programming (PPoPP'05), Chicago, USA, pp 186–195

37. Barham P, Dragovic B, Fraser K et al (2003) Xen and the art of virtualization. In: Proceedings of the 19th ACM symposium on operating systems principles (SOSP'03), Bolton Landing, USA, pp 164–177
38. VMware. http://www.vmware.com/
39. Microsoft Hyper-V. http://www.microsoft.com/hyper-v-server/en/us/default.aspx
40. Somani G, Chaudhary S (2009) Application performance isolation in virtualization. In: IEEE international conference on cloud computing (Cloud'09), Bangalore, pp 41–48
41. Stoess J, Lang C, Bellosa F (2007) Energy management for hypervisor-based virtual machines. In: Proceedings of the USENIX annual technical conference (ATC'07)
42. Kim KH, Beloglazov A, Buyya R (2009) Power-aware provisioning of cloud resources for real-time services. In: Proceedings of the 7th international workshop on middleware for grids, clouds and e-science (MGC'09), Champaign, USA
43. Kusic D, Kephart JO, Hanson JE et al (2009) Power and performance management of virtualized computing environments via lookahead control. Clust Comput 12(1):1–15
44. Laszewski G, Wang L, Younge AJ et al (2009) Power-aware scheduling of virtual machines in dvfs-enabled clusters. In: IEEE international conference on cluster computing and workshop, New Orleans, LA, pp 1–10
45. Younge AJ, Laszewski G, Wang L et al (2010) Efficient resource management for cloud computing environments. In: International conference on green computing, Chicago, USA, pp 357–364
46. Rodero I, Jaramillo J, Quiroz A et al (2010) Energy-efficient application-aware online provisioning for virtualized clouds and data centers. In: International conference on green computing, Chicago, USA, pp 31–45
47. Meisner D, Gold BT, Wenisch TF (2009) PowerNap: eliminating server idle power. In: Proceedings of the 14th international conference on architectural support for programming languages and operating systems (ASPLOS'09), Washington, USA, pp 205–216
48. Lee YC, Zomaya AY (2010) Energy efficient utilization of resources in cloud computing systems. J Supercomput. doi:10.1007/s11227-010-0421-3
49. Fan X, Weber WD, Barroso LA (2007) Power provisioning for a warehouse-sized computer. In: Proceedings of the 34th annual international symposium on computer architecture (ISCA'), NY, USA, pp 13–23
50. Colarelli D, Grunwald D (2002) Massive arrays of idle disks for storage archives. In: Proceedings of the ACM/IEEE conference on supercomputing (SC'02), Baltimore, USA, pp 1–11
51. Son SW, Chen G, Kandemir M (2005) Disk layout optimization for reducing energy consumption. In: Proceedings of the 19th annual international conference on supercomputing (ICS'05), Cambridge, USA, pp 274–283
52. Narayanan D, Donnelly A, Rowstron A (2008) Write off-loading: practical power management for enterprise storage. ACM Trans Storage 4(3):1–23
53. Carrera EV, Pinheiro E, Bianchini R (2003) Conserving disk energy in network servers. In: Proceedings of the 17th annual international conference on supercomputing (ICS'03), San Francisco, USA, pp 86–97
54. Pinheiro E, Bianchini R (2004) Energy conservation techniques for disk array-based servers. In: Proceedings of the 18th annual conference on supercomputing (ICS'04), Saint-Malo, France, pp 68–78
55. Pinheiro E, Bianchini R, Dubnicki C (2006) Exploiting redundancy to conserve energy in storage systems. In: Proceedings of ACM SIGMETRIC'06, Saint Malo, France, pp 15–26
56. Huang H, Hung W, Shin KG (2005) FS2: dynamic data replication in free disk space for improving disk performance and energy consumption. In: Proceedings of the 20th ACM symposium on operating systems principles (SOSP'05), Brighton, UK, pp 263–276
57. Weddle C, Oldham M, Qian J et al (2007) PARAID: The gear-shifting power-aware RAID. ACM Trans Storage 3(13):245–260
58. Chou J, Kim J, Rotem D (2011) Energy-aware scheduling in disk storage systems. In: 31st international conference on distributed computing systems (ICDCS'11), pp 423–433
59. Kim J, Rotem D (2010) Using replication for energy conservation in RAID systems. In: Parallel and distributed processing techniques and applications conference (PDPTA'10)
60. Liao XL, Bai S, Wang YP et al (2011) ISRA-based grouping: a disk reorganization approach for disk energy conservation and disk performance enhancement. IEEE Trans Comput 60(2):292–304
61. Gurumurthi S, Sivasubramaniam A, Kandemir M et al (2003) DRPM: dynamic speed control for power management in server class disk. In: Proceedings of the 30th annual international symposium on computer architecture (ISCA'03), San Diego, USA, pp 169–181

62. Zhu Q, Shankar A, Zhou Y (2004) PB-LRU: a self tuning power aware storage cache replacement algorithm for conserving disk energy. In: Proceedings of the 18th annual international conference on supercomputing (ICS'04), Saint Malo, France, pp 79–88
63. Zhu Q, David FM, Devaraj CF et al (2004) Reducing energy consumption of disk storage using power-aware cache management. In: 10th international symposium on high performance computer architecture (HPCA'04)
64. Zhu Q, Chen Z, Tan L et al (2005) Hibernator: helping disk arrays sleep through the winter. In: Proceedings of the 20th ACM symposium on operating systems principles (SOSP'05), Brighton, UK, pp 177–190
65. Zhu Q, Zhou Y (2005) Power-aware storage cache management. IEEE Trans Comput 54(5):587–602
66. EMC Symmetrix 3000 and 5000 Enterprise Storage Systems product description guide. http://www.emc.com/products/productpdfs/pdg/symm_3_5_pdg.pdf
67. Ganesh L, Weatherspoon H, Balakrishnan M et al (2007) Optimizing power consumption in large scale storage systems. In: Proceedings of the 11th USENIX workshop on hot topics in operating systems (HotOS'07), CA, USA
68. Brunschwiler T, Smith B, Ruetsche E et al (2009) Toward zero-emission data centers through direct reuse of thermal energy. IBM J Res Dev 53(3):11.1–11.13
69. Hamann HF, Kessel TG, Iyengar M et al (2009) Uncovering energy efficiency opportunities in data centers. IBM J Res Dev 53(3):10.1–10.12
70. Ahmad F, Vijaykumar TN (2010) Joint optimization of idle and cooling power in data centers while maintaining response time. In: Proceedings of the 15th international conference on architectural support for programming languages and operating systems (ASPLOS'10), Pittsburgh, USA, pp 243–256
71. Chen Y, Gmach D, Hyser C et al (2010) Integrated management of application performance, power and cooling in data centers. In: Network operations and management symposium (NOMS'10), Osaka, Japan, pp 615–622
72. Pakbaznia E, Ghasemazar M, Pedram M (2010) Temperature-aware dynamic resource provisioning in a power-optimized datacenter. In: Proceedings of the conference on design, automation and test in Europe (DATE'10), Dresden, Germany, pp 124–129
73. Tang Q, Kumar S, Gupta S (2008) Energy-efficient thermal-aware task scheduling for homogeneous high-performance computing data center: a cyber-physical approach. IEEE Trans Parallel Distrib Syst 19(11):1458–1472
74. Moore J, Chase J, Ranganathan P (2006) Weatherman: automated, online, and predictive thermal mapping and management for data center. In: International conference on autonomic computing (ICAC'06), pp 155–164
75. Moore J, Chase J, Ranganathan P (2005) Making scheduling cool: temperature-aware workload placement in data centers. In: Proceedings of the annual conference on USENIX annual technical conference (ATC'05)
76. Kim MG, Choi JY, Kang M et al (2008) Adaptive power saving mechanism considering the request period of each initiation of awakening in the IEEE 802.16e system. IEEE Commun Lett 12(2):106–108
77. He Y, Yuan R (2009) A novel scheduled power saving mechanism for 802.11 wireless LANs. IEEE Trans Mob Comput 18(10):1368–1383
78. Kliazovich D, Bouvry P, Khan SU (2010) DENS: data center energy-efficient network-aware scheduling. In: ACM/IEEE international conference on green computing and communications, Hangzhou, China, pp 69–75
79. Kliazovich D, Bouvry P, Khan SU (2011) GreenCloud: a packet-level simulator of energy-aware cloud computing data centers. J Supercomput. doi:10.1007/s11227-010-0504-1
80. Mahadevan P, Sharma P, Banerjee S et al (2009) Energy aware network operations. In: Proceedings of IEEE INFOCOM'09, Rio de Janeiro, pp 1–6
81. Gunaratne C, Christensen K, Nordman B et al (2008) Reducing the energy consumption of Ethernet with adaptive link rate (ALR). IEEE Trans Comput 57(4):448–461
82. Chabarek J, Sommers J, Barford P et al (2008) Power awareness in network design and routing. In: Proceedings of IEEE INFOCOM'08, Phoenix, AZ, pp 457–465
83. Gupta M, Singh S (2003) Greening of the internet. In: Proceedings of ACM SIGCOMM'03, Karlsruhe, Germany, pp 19–26
84. Nedevschi S, Popa L, Iannaccone G et al (2008) Reducing network energy consumption via rate-adaptation and sleeping. In: Proceedings of the 7th USENIX conference on networked systems design and implementation (NSDI'08), San Francisco, USA

85. Heller B, Seetharaman S, Mahadevan P et al (2010) ElasticTree: saving energy in data center networks. In: Proceedings of the 7th USENIX conference on networked systems design and implementation (NSDI'10), Boston, USA
86. Gupta M, Singh S (2007) Using low-power modes for energy conservation in Ethernet LANs. In: Proceedings of IEEE INFOCOM'07, Anchorage, AK, pp 2451–2455
87. Gupta M, Singh S (2007) Dynamic Ethernet link shutdown for energy conservation on Ethernet links. In: IEEE international conference on communications (ICC'07), Glasgow, pp 6156–6161
88. Gunaratne C, Christensen K, Nordman B (2005) Managing energy consumption costs in desktop PCs and LAN switches with proxying, split TCP connections, and scaling of link speed. Int J Netw Manag 15(5):297–310