

Chapter 8

Energy-Aware Virtual Machine Consolidation in IaaS Cloud Computing

Md Hasanul Ferdaus and Manzur Murshed

Abstract With immense success and rapid growth within the past few years, cloud computing has been established as the dominant paradigm of IT industry. To meet the increasing demand of computing and storage resources, infrastructure cloud providers are deploying planet-scale data centers across the world, consisting of hundreds of thousands, even millions of servers. These data centers incur very high investment and operating costs for the compute and network devices as well as for the energy consumption. Moreover, because of the huge energy usage, such data centers leave large carbon footprints and thus have adverse effects on the environment. As a result, efficient computing resource utilization and energy consumption reduction are becoming crucial issues to make cloud computing successful. Intelligent workload placement and relocation is one of the primary means to address these issues. This chapter presents an overview of the infrastructure resource management systems and technologies and detailed description of the proposed solution approaches for efficient cloud resource utilization and minimization of power consumption and resource wastages. Different types of server consolidation mechanisms are presented along with the solution approaches proposed by the researchers of both academia and industry. Various aspects of workload reconfiguration mechanisms and existing works on workload relocation techniques are described.

Keywords Cloud computing · Energy-awareness · Virtualization · Server consolidation · Reconfiguration · Virtual machine migration · Combinatorial optimization

M. H. Ferdaus (✉)

Faculty of Information Technology, Monash University, Churchill, VIC 3842, Australia
e-mail: md.ferdaus@monash.edu

M. Murshed

School of Information Technology, Faculty of Science, Federation University Australia, Churchill, VIC 3842, Australia
e-mail: manzur.murshed@federation.edu.au

8.1 Introduction

With the rapid development of computing and storage technologies and the extreme success of the Internet, computing resources have become more powerful, cheaper, and ubiquitously available than ever before. This technological shift has enabled the realization of a new computing paradigm called *Cloud Computing*. Technically speaking, clouds are large pool of easily accessible and readily usable virtualized resources, such as hardware (e.g., CPU, memory, storage), development platforms (e.g., Java, .NET, Go), and services (e.g., Email, CRM, HR) that can be dynamically reconfigured to adjust to a variable load in terms of scalability, elasticity, and load balancing, and thus allow opportunities for optimal resource utilization. This pool of resources is typically provisioned as a pay-per-use business model in which very high availability and guarantee (e.g., 99.99% for Amazon S3) are offered by the cloud infrastructure provider by means of service level agreements (SLAs) [49]. Consumers of cloud can access resources and services based on their requirements without any regard of the location of the consumed resource and service. A similar concept of delivering computing resources has been termed *Utility Computing* in the arena of information technology for a few decades. Recent advancement in technologies like high-speed internet, virtualization, and web 2.0, and high availability of commodity computing equipment have paved the way of cloud computing to a quick success.

According to the National Institute of Standards and Technology (NIST) definition [32], the five essential elements of cloud computing are:

- On-demand computing service
- Broad network access
- Resource pooling
- Rapid elasticity, and
- Measured service

In addition to these five essential characteristics, the cloud community has extensively used the following service models to categorize the cloud services [49]:

- Infrastructure as a service (IaaS): Cloud provides provision for computing resources (e.g., processing, network, storage) to cloud customers in the form of virtual machines (VM), for example Amazon EC2 and Google compute engine.
- Platform as a service (PaaS): PaaS providers offer a development platform (programming environment, tools, etc.) that allows cloud consumers to develop cloud services and applications as well as a deployment platform that hosts those services and applications, thus supports full software lifecycle. Examples include Google App Engine and Windows Azure.
- Software as a service (SaaS): Cloud consumers release their applications on a hosting environment fully managed and controlled by SaaS cloud providers and the applications can be accessed through internet from various clients (e.g., web browser and smartphones). Examples are Google Apps and Salesforce.com.

To respond to the rapid growth of customer demands for processing power and storage, cloud providers like Amazon, Microsoft, and Google are deploying large number of planet-scale power-hungry data centers across the world. Cloud giants like Microsoft and Google individually have more than 1 million servers in their data center infrastructures, as recent report shows [35]. As a consequence, a huge amount of energy is required to run the servers and keep the cooling systems operating for these gigantic data centers. As per the Data Center Knowledge report [42], power is one of the critical total cost of ownership (TCO) variables in managing data centers, and servers and data equipment are responsible for 55% of energy used by the data center followed by 30% for the cooling equipment.

Large data centers are not only expensive to maintain, but also have enormous effects on environment. According to McKinsey report [25], world data centers consume 0.5% of world's electricity and drive in more carbon emission than both Argentina and the Netherlands. The reason behind this extremely high energy consumption is not just the amount of computing resources used and the power inefficiency of the hardware, but also lies in inefficient use of these resources. Data collected from more than 5000 production servers over 6-month period showed that on average servers operate only at 10–15% of their full capacity most of the time, leading to expenses on overprovisioning of resources [4]. Narrow dynamic power range of server further aggrandizes the problem: even completely idle servers consume about 70% of their peak power usage [17]. As cloud promises unlimited resources through elastic provisioning, absolute reliability and availability, as well as customer demands show high dynamics, overprovisioning of resources in cloud data centers is a common phenomenon.

Among all the service models, the key for the success of cloud computing is the IaaS substrate that enables cloud service providers to provision the computing infrastructure needed to deliver the services simply by renting resources as long as needed without even buying a single component. Cloud infrastructures depend on one or more data centers, either centralized or distributed and on the use of various cutting-edge resource virtualization technologies, which enable the same physical resource (computing, network, or storage) to be shared among multiple application environments. Virtualization technologies allow data centers to address resource and energy inefficiency by creating multiple VMs in a single physical machine, each of which representing a runtime environment completely isolated from one another and by live migrating VMs [11] from one server to another, and thus improving resource utilization. Reduction of energy consumption can be achieved by switching idle physical servers to lower power states (suspended or turned off) while still preserving customers performance requirements. Thus, monitoring server utilization, making appropriate workload relocation decision, and by this process, improving data center resource utilization and energy consumption, technically termed *VM Consolidation (or Server Consolidation or Workload Consolidation)* is an essential part of resource management of virtualized data centers [54], including cloud data centers.

Higher resource utilization and energy efficiency in cloud data centers through server consolidation come with the associated overhead or cost of reconfiguration

of the workloads. Relocation of VM from one machine to another using VM live migration consumes nonnegligible amount of computing and network resources [11]. Also, VM live migration may lead to significant performance issues for the hosted applications depending on the current resource utilization conditions in the physical servers, network traffic, types of applications, and other colocated workloads [1, 24, 55]. The most obvious effect of VM live migration that hosted applications perceive is the VM downtime when the applications will be unavailable to the clients. The domain of applications that leverages the cloud platforms is broad, including high performance computing (HPC), video processing, scientific simulation, and web applications. With the wide adaptation of Web 2.0 technologies, modern web applications such as social networking and e-commerce websites exhibit highly dynamic and interactive characteristics and thus, resulting in particular client/server communication patterns, write patterns, and server load compared with traditional static web applications. Proper estimation of the total cost or overhead of reconfiguration through VM live migration techniques in a cloud setting is essential to guide server consolidation, VM multiplexing and scheduling schemes so that trade-off between VM packing efficiency that gives measure of server resource utilization and reconfiguration overhead that impacts customer SLA can be performed. As a response, research community has contributed to the appropriate design, modeling, and validation techniques to estimate realistic reconfiguration costs considering both system parameters and application characteristics.

The rest of the chapter is organized as follows: Sect. 8.2 presents a brief overview of the architectural components and underlying technologies of IaaS cloud infrastructure. Resource management issues and challenges of IaaS clouds including server resource utilization and energy management along with the solution approaches in existing works are described in Sect. 8.3. Finally, Sect. 8.4 summarizes the content of the chapter.

8.2 IaaS Cloud Management Systems

While the number and scale of cloud computing services and systems are continuing to grow rapidly, significant amount of research is being conducted both in academia and industry to determine the directions to the goal of making the future cloud computing platforms and services successful. As most of the major cloud computing offerings and platforms are proprietary or depend on software that is not accessible or amenable to experimentation or instrumentation, researchers interested in pursuing cloud computing infrastructure questions as well as future cloud service providers have very few tools to work with [41]. Moreover, data security and privacy issues have created concerns for enterprises and individuals to adopt public cloud services [2]. As a result, several attempts and ventures of building open-source cloud computing solutions came out of both academia and industry collaborations including

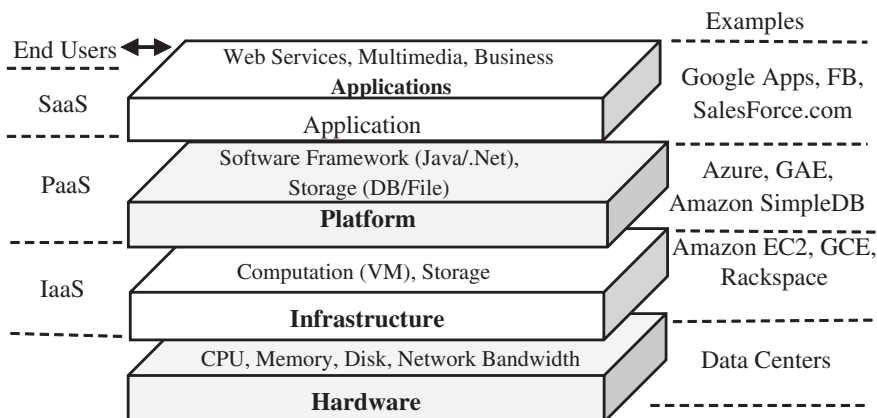


Fig. 8.1 Cloud computing architecture

Eucalyptus [41], OpenStack, OpenNebula [44], and Nimbus¹. These cloud solutions provide various aspects of cloud infrastructure management such as:

- Management services for VM life cycle, compute resources, networking, and scalability.
- Distributed and consistent data storage with built-in redundancy, failsafe mechanisms, and scalability.
- Discovery, registration, and delivery services for virtual disk images with support of different image formats (VDI, VHD, qcow2, VMDK).
- User authentication and authorization services for all components of cloud management.
- Web and console-based user interface for managing instances, images, cryptographic keys, volume attachment/detachment to instances, and similar functions.

From the architectural perspective, the cloud computing environment is divided into four layers as presented in Fig. 8.1, as follows:

- *Hardware layer*: This layer is responsible for managing the physical resources of the cloud, including physical servers, routers, switches, power, and cooling systems.
- *Infrastructure layer*: This layer (also known as *Virtualization layer*) creates a pool of computing and storage resources by partitioning the physical resources using virtualization technologies such as Xen [3] and VMware.
- *Platform layer*: Built on top of the infrastructure layer, this consists of operating systems and application frameworks and minimizes the burden of deploying applications directly on the VM containers.

¹ Nimbus Project. <http://www.nimbusproject.org/>.

- *Application layer*: This layer consists of the actual cloud applications, which are different from traditional applications and can leverage the automatic-scaling feature of cloud to achieve better performance, availability, and lower operating cost.

8.2.1 *Virtualization Technologies*

One of the main enabling technologies that paved the way of cloud computing toward its extreme success is *virtualization*. Cloud leverages various virtualization technologies (machine, network, storage) to provide users an abstraction layer that provides a uniform and seamless computing platform by hiding its hardware heterogeneity, geographic boundaries, and internal management complexities [59]. It is a promising technique by which resources of physical servers can be abstracted and shared through partial or full machine simulation by time-sharing and hardware and software partitioning into multiple execution environments each of which runs as complete and isolated system. It allows dynamic sharing and reconfiguration of physical resources in cloud computing infrastructure that makes it possible to run multiple applications in separate VMs having different performance metrics. It is virtualization that makes it possible for the cloud providers to improve utilization of physical servers through VM multiplexing [33] and multitenancy (i.e., simultaneous sharing of physical resources of same server by multiple cloud customers). It also enables on-demand resource pooling through which computing resources, like CPU and memory, and storage resources are provisioned to customers only when needed [27]. This feature helps avoid static resource allocation based on peak resource demand characteristics. In short, virtualization enables higher resource utilization, dynamic resource sharing, and better energy management, as well as improves scalability, availability, and reliability of cloud resources and services [9].

Virtualization in modern computing has been implemented using different approaches. Two significant techniques that have been heavily deployed in cloud computing infrastructures are full virtualization and paravirtualization:

- *Full virtualization* [3] provides a complete VM enabling unmodified guest operating systems (guest OS) to run in isolation. It provides flexibility to run different versions of different operating systems and the guest OS does not know that it is being virtualized. However, full virtualization requires Hardware Virtualization support (e.g., Intel-VT, AMD-V) from underlying host server.
- *Paravirtualization* [14] provides a complete but specialized VM to each guest OS allowing modified guests to run in isolation. It provides a lightweight and near native speed, and allows the guest OS to cooperate with hypervisor to improve performance. However, this technology is only limited to open source guest OS.

Hypervisor, also termed *Virtual Machine Monitor* (VMM), is the piece of software that multiplexes hardware among the VMs that it provides, the way traditional operating systems multiplexes hardware among the various processes [43]. Among

the various virtualization systems, VMware, Xen, and KVM (Kernel-based Virtual Machine) [26], as listed below, have proved to be the most successful by combining features that make them uniquely well suited for many important applications:

- VMware Inc. is the first company to offer commercial virtualization technology. It offers a hypervisor called ESXi² server that supports full virtualization. Paravirtualization can also be supported by using VMI [31].
- Xen [15] is one of a few Linux hypervisors that support both full virtualization and paravirtualization. Each guest OS (termed domain in Xen terminology) uses a preconfigured share of the physical server. A privileged domain called Domain0 is a bare-bone OS that actually controls physical hardware and create, configure, migrate, or terminate other VMs.
- KVM [26] also supports full virtualization. It is a modification to the Linux kernel that actually makes Linux into a hypervisor on inserting a KVM kernel module. One of the most interesting KVM features is that each guest OS running on it is actually executed in user space of the host system. This approach makes each guest OS look like a normal process to the underlying host kernel.

8.2.2 VM Migration Techniques

One of the most prominent features of the virtualization systems is the VM *Live Migration* [11], which allows for the transfer of a running VM from one physical machine to another, with little downtime of the services hosted by the VM. It transfers the current working state and memory of a VM across the network while they are running. This has been already a built-in feature for both Xen and KVM. VMware also added live migration feature called VMotion [39]. Other architectures including Microsoft Hyper-V, Oracle VirtualBox, and OpenVZ also support this feature.

Another approach for VM migration is *Cold* or *Static Migration* [47] in which the VM to be migrated is shut down and a configuration file is sent from the source machine to the destination machine. The same VM can be started on the target machine by using the configuration file. This is a much faster and convenient way to migrate a VM with negligible increase in network traffic, but static VM migration incurs high downtime.

8.3 Energy-Aware VM Consolidation and Reconfiguration in IaaS Cloud Data Centers

Resource allocation in cloud has been challenging because of the unique service features that cloud claims to provide; on-demand resource provisioning and pay-as-you-go pricing policy not only create flexible and attractive business models, but also intricate the resource management functions and operations. To support

² vSphere ESX and ESXi, VMware Inc. <http://www.vmware.com/au/products/esxi-and-esx/>.

such service models, cloud providers need to deploy dynamic resource management systems that would maximize resource utilization while minimizing energy consumption and operating costs. Cloud provides elasticity and high scalability of resources that require autonomous and self-configured management systems [59]. To ensure constant high resource utilization, clouds allow multitenancy and shared resource pooling where workloads and VMs from different users and possibly of different application environments can colocate on the same physical servers [8]. Clouds leverage virtualization technologies [14] that allow integration of flexible and efficient resource management strategies into cloud infrastructure. Resource management policies and algorithms in the arena of public clouds are not disclosed due to business reason. Moreover, the current open-source cloud management systems like OpenStack and Eucalyptus take simplistic views on resource management and provide very basic algorithms such as random, round-robin, or uniform with primary focus on load balancing.

8.3.1 *Energy-Efficient VM Consolidation*

While cloud computing provides many advanced features, it still has some shortcomings such as the relatively high operating costs for both public and private clouds. The area of *Green Computing* is also becoming increasingly important in a world with limited energy resources and an ever-rising demand for more computational power. As pointed out before, energy costs are among the primary factors that contribute to the TCO and its influence will grow rapidly due to the ever increasing demands of resources and continuously increasing electricity costs [21]. As a consequence, optimization of energy consumption through efficient resource utilization and management is equivalent to operating cost reduction in data center management. To optimize the energy consumption of the physical devices, different techniques have been proposed and used, including server consolidation, energy-aware resource management frameworks and design strategies, and energy-efficient hardware devices.

Resource management and optimization is getting more challenging day-by-day for large-scale data centers like cloud data centers due to their rapid growth, high dynamics of hosted services, resource elasticity, and guaranteed availability and reliability. Static resource allocation techniques used in traditional data centers are simply inadequate to address these newly immersed challenges [23]. With the advent of virtualization technologies, server resources are now better managed and utilized through server consolidation by placing multiple VMs hosting several applications and services in a single physical server, and thus ensuring efficient resource utilization. Energy-efficiency is achieved by consolidating the running VMs in minimum number of servers and transitioning idle servers into lower power states (i.e., sleep or shut down mode).

VM consolidation techniques provide VM placement decisions that indicates the mapping of each running VM to appropriate server. Depending on the initial

condition of data centers that VM consolidation techniques start with, it is categorized into two variants: *Static* and *Dynamic VM Consolidation*.

8.3.1.1 Static VM Consolidation

The static VM consolidation techniques start with a set of fully empty physical servers, either homogenous or heterogeneous with specific resource capacity and a set of workloads in the form of VMs with specific resource requirements. Thus, such consolidation mechanisms require prior knowledge about all the workloads and their associated resource demands. Such techniques are useful in situations like initial VM placement phase or migration of a set of workload from one data center to another. Static consolidation does not consider the current VM-to-server assignments and thus unaware of the associated VM migration overheads on both the underlying network traffic and hosted application performance [19]. Considering the predominant energy-costs of running large data centers and low utilization of servers resulted by traditional resource management technologies, and through the blessings of virtualization techniques, VM placement strategies like server consolidation have become a hot area of research [18, 20, 22, 40, 48, 50].

8.3.1.2 Dynamic VM Consolidation

Consolidation mechanisms that consider the current VM-to-server assignments for the consolidation decision fall in the category of dynamic consolidation. Contrary to static consolidations where the current allocations are disregarded and whole new solution of VM placement is constructed without considering the cost of reallocation of resources, dynamic consolidation techniques include the cost or overhead of relocation of existing workloads into the modeling of consolidation and try to minimize relocation overhead and maximize consolidation. Such server consolidation mechanisms employ VM live or cold migration techniques [11, 39] to move around workloads from servers with low utilization and consolidate them into minimum number of servers, thus improving overall resource utilization of the data center and minimizing power consumption.

As clouds offer an on-demand pay-as-you-go business model, customers can demand any number of VMs and can terminate their VMs when needed. As a result, VMs are created and terminated in the cloud data centers dynamically. This causes resource fragmentation in the servers, and thus leads to degradation in server resource utilization. However, efficient resource management in clouds is not a trivial task, as modern service applications exhibit highly variable workloads causing dynamic resource usage patterns. As a result, aggressive consolidation of VMs can lead to degradation of performance when hosted applications experience an increasing customer demand resulting in a rise in resource usage. As cloud providers ensure reliable quality of service (QoS) defined by SLAs, resource management systems in cloud data centers need to deal with the energy-performance trade-off.

To estimate the cost of relocation of workloads by the dynamic VM consolidation techniques, several system and network level metrics and parameters are used as modeling elements, such as the number of VM live migrations required to achieve the new VM-to-server placement [19], VM active memory size, speed of network links used for the migration [1, 23, 51], page dirty rate [52], and application-specific performance model [24].

8.3.1.3 VM Consolidation Modeling Techniques

Cloud data centers consist of hundreds or thousands, or even millions of high-end servers, for example rack-mount servers and blade servers with virtualization enabled to allow on-demand creation and termination of VMs on them. Popular cloud providers (e.g., Google, Amazon, and Rackspace) offer their customers different categories of VM instances to run with specification for each type of resource like the number of CPU cores, amount of memory, network bandwidth, and storage capacity. According to modern data center architectures³, data storage is implemented as storage area network (SAN) or network attached storage (NAS) and is architecturally separate from compute servers. This type of architectural separation provides IaaS cloud providers the flexibility to offer on-demand storage blocks (e.g., Amazon EBS) to their customers. As a consequence, most of the recent works on VM placement considers compute (CPU and memory) and network resource (network I/O) that are relevant to the physical servers and the VMs running on them.

Moreover, VM instances offered by public cloud providers differ in their individual resource capacities: some instances are larger than others (e.g., AWS EC2 instances: small, large, extra-large, etc.) whereas some instances have relatively higher capacity for one type of resource compared with their other resources (e.g., Google instances: High CPU, High Memory, etc.). Such diverse range of VM instances are offered to match the workload characteristics of the hosted cloud applications that range from web and enterprise business applications to HPC, scientific, and complex workload applications.

As cloud VM instances host various types of applications, the active VMs in cloud data centers exhibit dynamic resource demands during run-time. This dynamic nature of VMs can be captured and intelligently used to perform workload prediction and estimation mechanisms [57]. Because of the various types of VM instances offered by the providers with emphasis on size and types of resources and dynamic change in workload demands, it is very common that they will have random and nonuniform resource demands in difference resource dimensions of CPU, memory, and network I/O. To appropriately capture the various types of resource capacities of physical servers and the different types of resource requirements of hosted VMs, the VM consolidation problem is usually modeled as a variant of multi-dimensional vector packing problem (mDVPP) [20, 36] and multi-dimensional bin

³ Cloud-ready Data Center Reference Architecture. Juniper Networks, Inc. <http://www.juniper.net/us/en/local/pdf/reference-architectures/8030001-en.pdf>.

packing problem (mDBPP) [18, 19, 23], and sometimes as multiple knapsack problem (MKP) [40, 48]. In [36], the authors argued that VM consolidation is in fact an instance of mDVPP rather than mDBPP and some analysis is presented in their work. All of the aforementioned problems fall in the broad category of Discrete Combinatorial Optimization and from computational complexity perspective, these problems are NP-hard in nature and the best known algorithms that guarantee to identify an optimal solution have exponential time worst case complexity [13].

Most of the research works on VM consolidation consider the cloud data center environment consisting of homogeneous physical servers (or PMs) having same types of resources (e.g., CPU, memory, and network I/O) with different capacity represented as 2-tuple (CPU, MEM) or 3-tuple (CPU, MEM, IO). Resource demands of active VMs are also represented in a similar fashion. It is assumed that individual VM resource demand does not exceed individual PM resource capacity; otherwise the VM request is rejected. Given the set of servers with their respective resource capacities and the VM with their respective resource demands, the VM consolidation algorithms try to find VM-to-server placement mappings with some defined objective function that they try to minimize or maximize while maintaining the physical servers' resource capacity constraints. In the case of static VM consolidation, the objective function is very often modeled as a minimization function that tries to minimize the number of active servers that are used for VMs assignments [18, 23, 40, 48]. On the other hand, in the case of dynamic VM consolidation, the objective function is often formulated as a combination of maximization of the number of released servers (i.e. servers that are made empty and turned to power saving states) and resource utilization of active servers, as well as minimization of the number of VM migrations required for the new VM placement [19].

Depending on the modeling technique, static VM consolidation is often regarded as a *single-objective problem* where dynamic VM consolidation is considered as a *multiobjective problem* [19]. However in [20], the authors modeled the static VM consolidation problem as a multi-objective combinatorial optimization problem with the goal of simultaneously optimizing the total resource wastage and power consumption.

Server Resource Utilization and Wastage Modeling Depending on the VM placement decisions, the remaining resources available to use in physical servers may vary greatly. As different VMs have different resource demands along multiple resource dimensions, server resource utilization and wastage models need to capture the level of imbalance in utilization for particular VM-to-server assignments. A simple approach of capturing the utilization of multidimensional resources of a server as presented in [18] that uses L_1 norm based mean estimator, is:

$$U = U^{\text{CPU}} + U^{\text{MEM}} + U^{\text{IO}},$$

where U^{CPU} , U^{MEM} , and U^{IO} represent the normalized CPU, memory, and network I/O utilization (i.e. the ratio of used resource to total resource) after the VM assignments.

As the goal of static VM consolidation is to minimize the number of active servers by placing as many VMs as possible in those servers, minimization of resource wastage along every possible resource dimension is essential to improve the VM packing efficiency of the consolidation algorithm. Focusing on this goal, authors in [20] presented server resource wastage model by the following formulation (considering CPU and memory resources only):

$$W = \frac{|L^{\text{CPU}} - L^{\text{MEM}}| + \varepsilon}{U^{\text{CPU}} + U^{\text{MEM}}},$$

where U^{CPU} and U^{MEM} represent the normalized CPU and memory resource usage, and L^{CPU} and L^{MEM} denote the normalized remaining CPU and memory resource, and ε is a very small positive real number that is set to be 0.0001. The key point of the above resource wastage modeling is to make effective use of the server resources along each dimension and balance the left out resources across different dimensions.

Power Consumption Modeling It has been shown experimentally that power consumption of physical servers is dominated by their CPU utilization and increases linearly [17]. As a result, the electricity energy drawn by a server is usually represented as a linear function of its current normalized CPU utilization U^{CPU} :

$$E = \begin{cases} (E_{\text{max}} - E_{\text{idle}}) \times U^{\text{CPU}} + E_{\text{idle}}, & \text{if } U^{\text{CPU}} > 0 \\ 0, & \text{otherwise} \end{cases},$$

where E_{max} and E_{idle} are the average electrical power drawn when the server is fully utilized and idle, respectively.

Finally, the estimate of the total energy consumed by a VM placement decision is computed as the sum of the individual energy consumption of the active servers [18, 20]. Due to the nonproportional power usage (i.e. high idle power) of commodity servers, the idle servers (i.e., servers that do not host any running VM) are turned off or put in suspended or sleep mode after the new VM placement and are not considered in the total energy consumption model. If a data center consists of n servers, the overall energy consumption of a VM placement decision x is formulated as follows:

$$E(x) = \sum_{p=1}^n E(p).$$

8.3.1.4 Taxonomy and Survey of VM Consolidation Mechanisms

With the increasing adoption of virtualization technologies and rapid success of hosting services, and very recently of cloud computing, VM consolidation techniques have been very attractive to reduce energy costs and increase data center resource utilization. As resource management mechanisms of public clouds (such

as Amazon AWS) are not known in the public domain due to business policies, several open-source cloud projects (such as Eucalyptus [41], OpenStack, and OpenNebula [44]) have emerged as a means of alternative solutions to the proprietary cloud infrastructures. However, one of the major limitations of these current cloud frameworks is the absence of efficient energy-aware workload consolidation mechanisms. As a result, a good amount of research works have been conducted and published within the past few years with focus on different aspects of consolidation ranging from energy saving and resource usage optimization to minimization of VM migration overhead and SLA violations.

To analyze, assess, and compare among the various research works, taxonomy and characterization have been established as proven methodologies in any research area. The proposed research works on VM consolidation have incorporated state-of-the-art technologies in data center management, including virtualization, autonomic data center management platforms, cloud management systems, and various types of simulated and real-world workloads and benchmarking tools. A brief description of the identified aspects of the research works used in the course of taxonomy is given below:

1. *System assumption*: Server resources in data center or IT infrastructure are primarily modeled as either homogeneous or heterogeneous. Homogeneous cluster of servers normally represent servers with same capacity for certain fixed types of resources (e.g., CPU, memory, and storage), whereas heterogeneous cluster of servers can represent either mean servers having different capacities of resources or different types of resources (e.g., virtualized servers powered by Xen or VMWare hypervisor, and servers with graphics processing units (GPUs)).
2. *Server resource*: Generally, optimization across different ranges of resources (i.e. CPU, memory, network I/O, storage, etc.) is harder than single resource optimization. Often various mean estimators (such as L1 norm, vector algebra, etc.) are used to compute equivalent scalar estimation while trying to optimize across multiple types of server resources. This aspect has direct influence on the modeling techniques applied in the research works and also on the consolidation performance.
3. *Modeling technique*: As for any research problem, the solution approach varies depending on the modeling (mathematical, analytical, or algorithmic) applied for the addressed problem. The characteristics of VM consolidation problem make it most resemble to the general mDBPP/mDVPP. Furthermore, depending on the objectives/goals set in the research projects, modeling can vary across other theoretical problems such as multiple multidimensional knapsack problem, constraint satisfaction problem (CSP), and multiobjective optimization problems.
4. *Objective*: Most of the works set objective as to minimize the overall power consumption of the data center and maximization of server resource utilization by increasing the VM/workload packing efficiency using minimum number of active/running servers. With the consolidation process comes the tradeoff between application performance (and hence, SLA) and power consumption. With given importance on SLA violations, some of the works consider the cost

of reconfiguration primarily due to VM live migrations, and thus incorporate this cost in the objective function modeling. Moreover, some the works further focus on automated and co-ordinated management frameworks with the VM consolidation as an integral component of the proposed frameworks.

5. *Solution approach/Algorithm*: Considering the fact that the VM consolidation is a strictly NP-hard problem, algorithmic approaches in the research works vary from simple greedy approaches to metaheuristic strategies and local search methods. Greedy approaches such as First Fit Decreasing (FFD) and Best Fit Decreasing (BFD) are very fast in producing results but are not guaranteed to produce optimal solutions. Metaheuristics such as Ant Colony Optimization (ACO), Genetic Algorithms (GA), and Simulated Annealing (SA) work on initial or existing solutions and refine them to improve on objective function value. Exhaustive search methods (e.g., Constraint Programming (CP)) normally fix the domain of possible values for the model variables to compute the optimal solution within a reasonable amount of time; however, in this process these methods effectively limit the size of the data center (in terms of the number of servers) or the volume of the workload (in terms of the number of VMs).
6. *Evaluation/Experimental platform*: Evaluation methodologies have direct impact on the performance and practicality of the research works, most importantly in the competency analysis. Proposals that primarily have theoretical contributions mostly apply simulation based evaluation to focus highly on the algorithmic and complexity aspects, whereas works involving various workload patterns and application characteristics conduct their performance evaluation on real test beds or experimental data centers, or even on emulated platforms.
7. *Workload*: Depending on the experimental environment, the workload data used as input for the evaluation of various consolidation techniques varies from synthetic data to real-time application/VM workloads. Simulation-based evaluation primarily relies on synthetic workload data generated using various statistical models such as random, Gaussian, or Poisson distribution, or on workload dataset collected from real data centers. Evaluations based on experimental test beds mostly use real time workload data generated from the applications that are deployed and run in the test bed servers. Such test beds though capture realistic behaviors of applications and systems suffer from scalability issues in the domain of VM consolidation.

Analysis of VM Consolidation Solution Approaches Table 8.1 illustrates the most significant aspects of the notable recent research works in the area of energy-aware VM consolidation based on the contents and description found in the published materials. Depending on the analytical modeling techniques used in the existing works, various algorithmic and problem solving techniques are applied to solve server consolidation and related energy management problems [20], e.g.:

- *Greedy algorithms*: mDVPP and mDBPP as well as various knapsack problems have been well studied over the past few decades, and as a result a good amount of greedy heuristics have been proposed for both bin packing and knapsack problems in the fields of computer science and operations research. First-fit (FF),

Table 8.1 Aspects of notable recent research works on workload and server consolidation

Research project	System assumption	Server resources	Modeling technique	Objectives	Solution approach/algorithm	Evaluation/experimental platform	Workload
Adaptive threshold-based approach for energy-efficient consolidation of virtual machines in cloud data centers	Heterogeneous	CPU, memory, IO, and storage	Bin Packing problem	Threshold-based dynamic VM consolidation, minimization of energy consumption, SLA, and number of VM migration	Greedy Approach, Modified Best Fit Decreasing algorithm	CloudSim-based simulation	CPU utilization data from CoMon monitoring project for PlanetLab
EnaCloud: an energy-saving application live placement approach for cloud computing environments	Homogeneous	CPU and memory	Bin Packing problem	Energy-aware application scheduling and placement	Greedy heuristic based on First Fit and Best Fit algorithms	EnaCloud framework implemented in IVIC virtual computing environment and running on cloud server pool powered by Xen	Randomly generated workloads Web and database servers, compute-intensive applications, and common applications
pMapper: power and migration cost aware application placement in virtualized systems	Heterogeneous	CPU	Bin Packing problem	Power consumption minimization under performance constraints	Greedy Approach, Modified First Fit Decreasing	Simulation based on pMapper framework running on VMWare ESX-based testbed	Server utilization trace data from server farm
A mathematical programming approach for server consolidation problems in virtualized data centers	Homogeneous	CPU, memory, and bandwidth	Multi-dimensional Bin Packing problem, Linear Programming Relaxation	Server consolidation through workload data pre-processing and resource usage optimization	Greedy heuristic based on Branch and Bound, First Fit, and First Fit Decreasing, LP-relaxation-based heuristic	Open source solver Ip_solve 5.5.0.9 and COIN-OR CBC branch-and-cut IP solver, as well as ad hoc simulator for First Fit and First Fit Decreasing heuristics	Workload traces from Web, application, and database servers, as well as ERP applications from industry partner

Table 8.1 (continued)

Research project	System assumption	Server resources	Modeling technique	Objectives	Solution approach/algorithm	Evaluation/experimental platform	Workload
Entropy: a consolidation manager for clusters	Heterogeneous	CPU and memory	Instances of 2-dimensional Bin Packing and Knapsack Problem. Modeled as Constraint Satisfaction Problem	Dynamic VM consolidation, Minimization of the number of active servers and VM migrations	Exhaustive search (depth first) based on Constraint Programming	Simulation and Xen-powered Grid'5000 experimental testbed	Randomly generated synthetic data and NASGrid benchmark data
Autonomic virtual resource management for service hosting platforms	Heterogeneous	CPU and memory	Instance of Knapsack Problem. Modeled as Constraint Satisfaction Problem	Optimization of global utility function through maximization of SLA fulfillment and minimization of operating costs (number of servers)	Time-bound exhaustive search using Constraint Programming	Simulation based on Choco constraint solver	Synthetic Web workload distributed in a round-robin fashion
Performance and power management for cloud infrastructures	Homogeneous	CPU and memory	Instance of Multiple Knapsack Problem. Modeled as Constraint Satisfaction Problem	Utility maximization and energy cost minimization	Choco constraint programming sol-ver based brute-force search	Xen-powered testbed running cluster of Apache servers and farm of rendering applications controlled by Condor Grid scheduler	Web workload generated by CLJF load injector and batch workload using constant stream of jobs

Table 8.1 (continued)

Research project	System assumption	Server resources	Modeling technique	Objectives	Solution approach/algorithm	Evaluation/experimental platform	Workload
SLA-aware virtual resource management for cloud infrastructures	Homogeneous	CPU and memory	Instance of Multiple Knapsack Problem. Modeled as Constraint Satisfaction Problem	Autonomic dynamic VM provisioning and placement with utility maximization, active server minimization, and reduction of reconfiguration cost due to VM live migrations	Self-optimization through the combination of utility functions and constraint programming	Simulation environment based on 4 servers running cluster of web servers and multiplayer online game. The simulation relies on the Choco constraint solver	Synthetic workload distributed in a round-robin algorithm
Energy-aware ant colony based workload placement in clouds	Homogeneous	CPU, memory, IO, and storage	Multidimensional Bin Packing Problem	Energy consumption minimization through optimization of server resource utilization and minimization of the number of active servers	Ant Colony Optimization-based algorithm	Ad hoc simulation toolkit	Randomly generated VM resource demands

Table 8.1 (continued)

Research project	System assumption	Server resources	Modeling technique	Objectives	Solution approach/algorithm	Evaluation/experimental platform	Workload
A case for fully decentralized dynamic VM consolidation in clouds	Heterogeneous	CPU, memory, and IO	Decentralized system based on unstructured network of servers	Maximization of VM Packing efficiency and scalability, as well as minimization of the number of associated VM live migrations	Decentralized dynamic VM consolidation schema based on Cyclon membership protocol and Ant Colony Optimization-based algorithm	Python-based Cyclon P2P system emulator running on Grid'5000 experimental testbed	Randomly generated VM resource demands
A multiobjective ant colony system algorithm for virtual machine placement in cloud computing	Heterogeneous	CPU and memory	Multi-objective optimization and Multi-dimensional Vector Packing Problem	Minimization of power consumption and server resource wastage	Ant Colony Optimization-based algorithm	Ad hoc simulation toolkit	Randomly generated VM resource demands

best-fit (BF), next-fit (NF), FFD, BFD, choose pack (CP), and permutation pack (PP) are among the widely used greedy approaches [18]. A survey on the existing greedy solutions on single-dimensional bin packing problem can be found in [12]. In [5], the authors have presented a modified version of the BFD algorithm for the workload placement problem and have reported substantial energy saving based on simulation-driven results. Similarly in [29], a framework called EnaCloud is presented where a modified version of the BF algorithm is used. In [51], Verma et al. proposed pMapper, a VM placement scheme that models the workload placement as an instance of single-dimensional bin-packing problem and applies a modified version of the FFD heuristic to perform server consolidation. Further works on greedy algorithm based energy-aware VM placement approaches can be found in [30] and [46].

- *Linear programming*: This is a popular and traditional analytical approach to solve combinatorial optimization problems. Such linear programming formulations for server consolidation problems are presented in [6] and [45]. The authors also described constraints for limiting the number of VMs to be assigned to a single server and the total number of VM migrations, ensuring that some VMs are placed in different servers and placement of VMs to specific set of servers that has some unique properties. To minimize the cost of solving the linear programming problem, the authors further developed an LP-relaxation-based heuristic. Based on linear and quadratic programming model, Chaisiri et al. [10] presented an algorithm for finding optimal solutions to VM placement with the objective of minimizing the number of active servers.
- *CP*: VM placement and packing problem is also modeled as CSP, which is defined as a set of variables, a set of domains that represent the set of possible values for each variable and a set of constraints that denote the required relations between the values of the variables [48]. A solution of the CSP is a variable assignment that tries to maximize or minimize the value of a particular variable while maintain all the defined constraints. Based on CP, Hermenier et al. [23] proposed Entropy, a dynamic server consolidation manager for clusters that finds solutions for VM placement with the goal of active server minimization and tries to find any reconfiguration plan of the proposed VM placement solution with objective to minimize the necessary VM migration costs. Both the problems are solved using CP solver CHOCO [37]. The authors have provided detailed analysis and experimental results of the impacts of VM activity and VM memory size on the necessary VM migration duration and VM performance. Furthermore, several optimizations for the constraint solver are also suggested. Authors in [40] and [50] proposed an autonomic virtual resource management framework that separates the VM provisioning and VM packing phases. The VM provisioning phase takes resource level utility function [56] for each application environment as input and determines the necessary VMs from a list of pre-defined VM classes while maximizing a global utility function. The VM packing phase determines the best possible placement for all the VMs in the servers with the goal of minimizing the number of active servers. Both the phases resort to CHOCO CP solver [37]. Later in [48], the authors proposed extensions to their

framework with multiple components for modeling performance of applications, costs of provisioned VMs, and scheduling the VM provisioning and placement (with packing) phases. However, the proposed analysis does not allow scaling-up of VMs in terms of resources and does not consider multiplexing of VMs in a time-sharing manner, which is very often used as an efficient way to improve resource utilization in virtualized environments, especially in clouds.

- *Evolutionary algorithms*: Evolutionary algorithms like GA have already been proven as efficient techniques for solving optimization problem including combinatorial problems. Jing et al. [58] formulated the VM placement problem as a multiobjective optimization problem with objective of minimizing power consumption, total resource wastage, and thermal dissipation costs. As a solution, the authors proposed a modified GA with fuzzy multiobjective evaluation to search the large solution space efficiently and combining possibly conflicting objectives. In [34], the authors proposed GABA, a GA based adaptive and self-reconfiguration mechanism for VMs in cloud data centers that consist of heterogeneous servers. Based on time-varying requirements and dynamic environmental conditions, GABA can efficiently decide the optimal VM placements.
- *Swarm intelligence*: Swarm Intelligence is a relatively new approach to problem solving that takes inspiration from the social behaviors of insects and animals. Within the past two decades, ants have inspired a number of methods and techniques among which the most studied and the most successful is the general purpose optimization technique known as ACO [16]. In ACO, multiple artificial agents work independently within its local search space in a random, decentralized fashion with indirect form of interaction, and after multiple interactions the produced solutions converge to near optimality. ACO metaheuristics have been proven to be efficient in different problem domains and so far it has been tested on more than 100 different NP-hard problems, including discrete optimization problems. First work on solving single-dimensional bin-packing problem based on ACO metaheuristics was proposed in [28]. The authors argued that the complementary nature of ACO metaheuristics and local search can benefit from each other and presented experimental results and showed that their proposed algorithm can compete with the contemporary best known solutions. In [7], the authors have proposed AntPacking, an improvement over the previous algorithm shown to perform as good as the best known GA. In [18], Feller et al. first proposed a single-objective static VM consolidation algorithm based on a variant of ACO, namely Max-Min Ant System and presented improved performance over FFD greedy algorithm. Later in [19], the authors presented a multiobjective dynamic VM consolidation schema using appropriate adaptation of ACO metaheuristics. They proposed decentralized approach to solve the problem based on an unstructured peer-to-peer network of servers to address the issues of scalability and improved packing efficiency. Another ACO based multi-objective static VM consolidation algorithm is presented in [20] where the authors have developed models for server resource wastage and power consumption with focus on balanced resource utilization across multiple resource dimensions. The algorithm simultaneously tries to minimize the power consumption and total resource wastage of the servers that host running VMs.

8.3.1.5 Advantages and Disadvantages of VM Consolidation

Virtualization technologies have revolutionized the IT management works and opened up a new horizon of opportunities and possibilities. It has enabled application environments to be compartmentalized and encapsulated within VMs. By the use of VM and VM live migration techniques, virtualized data centers have emerged as highly dynamic environments where VMs hosting various applications are created, migrated, resized, and terminated instantaneously as required. Utilizing virtualization, IT infrastructure management has widely adapted VM consolidation techniques to reduce operating costs and increase data center resource utilization. The most notable advantages of adopting VM consolidation techniques are mentioned below:

1. *Reduction in physical resources:* By the help of efficient dynamic VM consolidation, multiple VMs can be hosted in single physical server without compromising hosted application performance. As a result, compared with static resource allocations where computing resources such as CPU cycles and memory frequently lay idle, through dynamic VM consolidation fewer numbers of physical machines can provide the same QoS and maintain SLAs, and thus effectively cut the TCO. Reduction in the number of servers also implies reduction in the cooling equipment necessary for the cooling operations in data centers.
2. *Energy consumption minimization:* Unlike other approaches of energy efficiency (e.g., implementing efficient hardware and operating systems), VM consolidation is a mechanism under the disposal of data center management team. If same level of service can be provided by fewer servers through VM consolidation, it implies minimization of energy costs both for the running servers and the operating cool systems. As energy costs continue to escalate, this implies a significant saving that will continue during the course of the data center operation.
3. *Environmental benefits:* World data centers contribute a significant portion of CO₂ emission and thus have enormous effects of environment. With recent trend toward *Green Data Centers*, VM consolidation is a major business drive in IT industry to contribute to the Green Computing.
4. *Minimization of physical space:* Reduction in the number of hardware implies reduction in the space needed to accommodate the servers, storage, network, and cooling equipment. Again, this contributes to the reduction of the TCO, as well as the operating costs.
5. *Decreased labor cost:* A major portion of the TCO of data centers is derived from administrative, support, and outsourced services, and thus VM consolidation can help trim down these costs significantly by reducing the maintenance effort.
6. *Automate maintenance:* By incorporating autonomic and self-organizing VM consolidation and VM migration techniques, much of the administrative and support tasks can be reduced and automated; and therefore, it can further reduce the maintenance overhead and costs.

With all the above mentioned benefits, if not managed and applied appropriately, VM consolidation can be detrimental to the services provided by the data center in at least the following ways:

1. *System failure and disaster recovery*: VM Consolidation puts multiple VMs hosting multiple service applications in a single physical server, and therefore can create single-point-of-failure (SPOF) for all the hosted applications. Moreover, upgrade and maintenance of a single server can cause multiple applications to be unavailable to users. Proper replication and disaster recovery plans can effectively remedy such situations. Since VMs can be saved in storage devices as disk files, virtualization technologies provide tools for taking snapshots of running VMs and resuming from saved checkpoints. Thus, with the help of shared storages such as NAS or SAN, virtualization can be used as convenient disaster recovery tool.
2. *Effects on application performance*: Consolidation can have adverse effects on hosted application performances due to resource contention, as they would share the same physical resources. Delay sensitive applications such as voice-over-IP (VoIP) and online audio-visual conferencing services as well as database management systems that require heavy disk activity need to be given special consideration during resource allocation phase of VM consolidation. Such applications can be given dedicated resources whereas delay-tolerant and less resource hungry applications can be scheduled with proper workload prediction and VM multiplexing schemes.
3. *VM migration and reconfiguration overhead*: Performing VM consolidation dynamically requires VM live migrations that have overheads on network links of the data center as well as on the CPU cycles of servers executing the migration operations. As a consequence, VM migrations and postmigration reconfigurations can have non-negligible impact on application performance. Experimental results [53] show that applications that are being migrated as well as colocated applications can suffer from performance degradation due to VM live migrations. As a consequence, VM consolidation mechanisms need to minimize the number of VM live migrations and its effects on applications.

Despite all the drawbacks of VM consolidation, due to its benefits in continuous reduction in energy and operating costs and increasing resource utilizations data center owners are increasing adopting VM consolidation mechanisms, especially for large data centers. As VM consolidation can have adverse effects on application performance, various characteristics and features of data center resources and hosted applications need to be taken into account during the design and implementation of VM consolidation schemes, such as heterogeneity of servers and storage devices, system software and tools, middleware and deployment platforms, physical and virtual network parameters, as well as application types, workload patterns, and load forecasting.

8.3.2 VM Migration and Reconfiguration

Dynamic reconfiguration of workloads in virtualized data centers is achieved through VM resizing and VM live migration techniques [11, 39]. While VM resizing

overhead in modern hypervisors is negligible, anecdotal evidence and experimental findings [24, 55] identified the VM live migration as reconfiguration mechanism with significant performance impact both on application and system resources. Thus, achievement of high packing efficiency with large number of VM migrations can effectively null and void the benefit of workload consolidation with the risk of possible high number of SLA violations of hosted applications and high resource wastage due to handling the migrations. However, the number of VM migrations alone does not represent the true overhead of the reconfiguration, as the total migration time and total VM downtime primarily depend on the Active Memory size of VM and speed of the network links used for the migration [1].

Moreover, both the source server and destination server experience extra CPU overhead during live migration, mostly due to the successive precopying phases [11, 39], which is an essential part of the state-of-the-art live migration subsystems in modern hypervisors like Xen [3], KVM [26], and VMWare ESXi. As multitenancy in cloud infrastructures is a common characteristic in today's clouds where VMs (and also applications) from different cloud customers can colocate in a single physical server, VM live migration overhead can have adverse effects on other customers' applications. Current cloud-hosted application domain is dominated by web applications, especially multitier web applications, and it is shown experimentally in [24] that the different J2EE-based tiers of RUBiS⁴, a widely used multitier benchmark, experience 40% to more than 200% change in their end-to-end mean response time due to live VM migrations. Furthermore, an extra amount of network bandwidth is consumed due to live migration, potentially affecting the responsiveness of hosted internet applications. Last but not the least, a slowdown of VM performance is also expected due to the cache warm-up at the destination server after the migration [38].

8.3.2.1 Reconfiguration Cost Modelling Principles

To design an efficient and pragmatic workload consolidation mechanism, it is important to properly estimate the associated overall cost of the reconfiguration plans, which is mostly dominated by cost of VM migrations. Several existing approaches for dynamic consolidation consider migration cost to be a function of single system parameter, like VM active memory size [23, 51], page dirty rate [52], or use an application-specific model [24], and thus being oblivious to server resource utilization levels, other colocated workloads, and resource usage characteristics as well as the demands of the hosted applications. The importance of considering such aspects in migration overhead estimation is evident from the report [51], which shows that the duration of a live migration for an application running identical workloads can vary by 50% or more depending on server utilization and other colocated VMs. Therefore, a usable model for live migration not only needs to be aware of application

⁴ RUBiS Benchmark, OW2 Consortium. <http://rubis.ow2.org/>.

and system parameters like active memory and write rate, but also take into account other colocated VMs, physical server utilization, as well as network parameters. A practical and accurate model of live migration is needed to complement dynamic consolidation schemes and provide an estimate of the cost of reconfiguration in cloud data centers.

Technically, live migration at the level of an entire VM refers to the process of transferring the active memory and execution state from the source server to the destination server. As in a typical cloud data center, the secondary memory or storage is implemented by SAN/NAS connected to compute servers through Internet small computer system interface (iSCSI), network file system (NFS), or server message block (SMB) protocols, VM disks are not transferred during migration. The most important aspect in terms of the performance impact of a live migration activity is the copying of in-memory state, as pre- and postmigration overheads (e.g., reattaching device drivers, advertising moved IP addresses) are pretty static [1, 11]. Among the several techniques for live migration in modern hypervisors, *Pre-copy Migration* is proven to be the most effective in terms of VM. Precopy migration involves two phases:

1. *Push phase* when Active Memory pages of running VM are copied from source to the target server in multiple rounds until some stop condition is fulfilled (e.g., the number of dirty pages during the last pre-copy iteration is less than some constant, like 50 for Xen) and
2. *Stop-and-copy phase* when the stop condition is met and the VM is stopped (and also its application) and all the remaining dirty pages are copied to the target server.

Two obvious temporal parameters are defined to measure the performance of a live migration, viz:

1. *Total migration time*: The total time required to move the VM between physical servers and
2. *Total downtime*: The portion of total migration time when the VM is not running.

Generally, the stop-and-copy phase is comparatively small for typical applications, usually between 1 to 3 s [55] and the push phase is much longer and increases with the size of memory being copied, page write patterns of applications, server resource utilization levels, and network link speed. As VM live migration requires significant amount of spare CPU, current resource utilization and the resource demands from colocated workloads, it can have significant effects on the total migration time and hosted application performance.

8.3.2.2 Related Works

Though the designers of the VM live migration technology do provide empirical evidence that suggests that the performance impact of live migration is manageable

[11, 39], recent experiments on live migrating VMs hosting different applications indicates that live migration can have significant impact on application performance and system resources [24, 55].

In [1], Akoush et al. addressed reconfiguration overhead solely in terms of the migration times and provided analytical derivation to define the upper and lower bounds of migration times, with particular emphasis on the Xen virtualization platform [3] and its live migration subsystem [11]. They have identified that link speed and page dirty rates are the major factors impacting migration behavior (in terms of migration times) and have a nonlinear effect on migration performance largely because of the hard stop conditions of Xen live migration algorithm that forces the migration to its final stop-and-copy phase. They also provided two migration simulation models based on average memory page dirty rate and historical data on page modification to predict migration times. The authors have also presented the effects of the following system and network parameters:

- *Network link bandwidth*: It is perhaps the most influential parameter on migration performance. Total migration time and VM downtime are inversely proportional to the migration link capacity.
- *Page dirty rate*: It is the rate at which memory pages of each VM are modified that directly affects the number of pages transferred in each push phase of the precopy migration. Higher page dirty rate causes more data to be sent per iteration leading to longer total migration time. Moreover, higher page dirty rates results in longer VM downtime, as more pages need to be sent in the final transfer round.
- *VM memory size*: In the precopy migration, the first iteration tries to copy across the entire VM allocated memory to the destination. As a result, on average the total migration time increases linearly with VM memory size.
- *Pre- and postmigration overhead*: It refers to operations that are not part of the actual transfer process. These are operations related to initializing a container on the destination host, mirroring block devices, maintaining free resources, etc.

In [38], an autonomic and transparent mechanism for proactive fault tolerance for arbitrary message passing interface (MPI) application has been studied and implemented using Xen live migration technology. In their research, the authors have given a general overview on the total migration time and possible parameters that affects it, but emphasis was given primarily on the amount of memory allocated to guest VMs.

In [24], Jung et al. have shown that runtime reconfiguration actions such as VM replication and migration can impose significant performance costs in multitier applications running in virtualized data center environments and proposed a middleware for generating cost-sensitive adaptation actions using a combination of predictive models and graph search techniques.

Voorsluys et al. in [55] showed experimental results of VM live migration on Internet applications using Web 2.0 benchmarking tool. They have shown that the average response times of typical multitier web application increases rapidly dur-

ing the live migration period, especially due to the postmigration overhead. Their results also demonstrate that in an instance of a nearly oversubscribed system, live migration causes a significant downtime (up to 3 s), a larger value than expected. The work presents valuable and realistic insights on the effects of VM live migration on SLA violations of today's web applications. However, the work lacks proper characterization and modelling of the factors and parameters that contribute to the migration cost.

In [52], Verma et al. presents a study on the cost of reconfiguration of cloud-based IT infrastructure with response to workload variations. Their study suggests that VM live migration requires a significant amount of spare CPU capacity on the source server. The study also suggests that if spare CPU cycles are not available, it impacts both the duration of migration and the performance of the hosted application. Later, in [53], the authors designed CosMig model that predicts (1) the total VM migration time, (2) performance impact of migration on the migrating VM, and (3) performance impact of migration on other colocated VMs. This model is based on CPU utilization and active memory size as these two parameters are normally monitored in large data centers. The authors also showed that by the use of selected microbenchmarks and representative applications, CosMig model has been able to accurately estimate the impact of live migration in a cloud environment. The following parameters were used in CosMig to determine the performance impact of migrating VM V_i :

- *Duration*: Time duration for the full migration completion.
- *VM self-impact*: Ratio between the drop in throughput of the hosted application of V_i during the migration period and the throughput without migration.
- *VM coimpact*: Ratio between the drop in throughput of any other application in colocated VM V_j during the migration period of V_i and the throughput of the same without migration of V_i .

8.4 Conclusions and Future Research Directions

Cloud computing is quite a new computing paradigm and from the very beginning it has been growing rapidly in terms of scale, reliability, and availability. Because of its flexible pay-as-you-go business model, virtually infinite pool of on-demand resources, guaranteed QoS, and almost perfect reliability, consumer base of cloud computing is increasing day-by-day. As a result, cloud providers are deploying large data centers across the globe. Such gigantic data centers not only incur huge energy costs, but also have environmental effects. Power consumption of such data centers can be improved by employing efficient resource allocation and management strategies through better server resource utilization. This chapter has discussed various virtual resource management technologies used in virtualized data centers including cloud data centers, as well as algorithms and mechanisms for achieving higher resource utilization and optimization of energy consumption through VM consolidation and data center reconfiguration. An in depth analysis

on the different approaches proposed by the recent research works has also been presented.

Virtual resource allocation and VM placement strategies play significant roles in resource management and optimization decisions in data centers. Modern cloud applications are composed of multiple compute and storage components, and such components exhibit communication correlations among themselves. Incorporation of the communication correlations during VM placement decisions is a very important area of research that is not yet explored enough. A typical objective for network-aware VM placement and relocation would be keeping the heavily communicating VMs in the same server so that inter-VM communication would take place through memory or in near proximity under the same edge switch, and thus keeping the overall network overhead minimum on the physical network infrastructure. Development of realistic power consumption models for network devices and VM placement and reallocation policies with power management capabilities are areas of potential optimization in data center management.

VM consolidation and resource reallocation through VM migrations with focus on both energy-awareness and network overhead is yet another area of research that requires much attention. VM placement decisions focusing primarily on server resource utilization and energy consumption reduction can produce data center configurations that are not traffic-aware or network optimized, and thus can lead to higher SLA violations. As a consequence, VM placement strategies utilizing both VM resource requirements information and interVM traffic load can come up with placement decisions that are more realistic and efficient.

Cloud environments allow their consumers to deploy any kind of applications in an on-demand fashion, ranging from compute intensive applications such as HPC and scientific applications, to network and disk I/O intensive applications like video streaming and file sharing applications. Colocating similar kinds of applications in the same physical server can lead to resource contentions for some types of resources while leaving other types under-utilized. Moreover, such resource contention will have adverse effects on application performance, thus leading to SLA violations and profit minimization. Therefore, it is important to understand the behavior and resource usage patterns of the hosted applications to efficiently place VMs and allocate resources to the applications. Utilization of historical workload data and application of appropriate load prediction mechanisms need to be integrated with VM consolidation techniques to minimize resource contentions among applications and increase resource utilization and energy efficiency of data centers.

Centralized VM consolidation and placement mechanisms can suffer from the problems of scalability and SPOF, especially for cloud data centers. One possible solution approach would be replication of VM consolidation managers; however, such decentralized approach is nontrivial, as VMs in the data centers are created and terminated dynamically through on-demand requests of cloud consumers, and as a consequence consolidation managers need to have updated information about the data center. As initial solution, servers can be clustered and assigned to the respective consolidation managers and appropriate communication and synchronization among the managers need to be ensured to avoid possible race conditions.

References

1. Akoush S, Sohan R, Rice A, Moore A, Hopper A (2010), Predicting the performance of virtual machine migration, in Modeling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS), 2010 IEEE International Symposium on, pp 37–46
2. Armbrust M, Fox A, Griffith R, Joseph A, Katz R, Konwinski A, Lee G, Patterson D, Rabkin A, Stoica I et al (2010) A view of cloud computing. *Communications of the ACM* 53(4):50–58
3. Barham P, Dragovic B, Fraser K, Hand S, Harris T, Ho A, Neugebauer R, Pratt I, Warfield A (2003) Xen and the art of virtualization, in *ACM SIGOPS operating systems review*, pp 164–177
4. Barroso L, Holzle U (2007) The case for energy-proportional computing. *Computer* 40(12):33–37
5. Beloglazov A, Buyya R (2010) Adaptive threshold-based approach for energy-efficient consolidation of virtual machines in cloud data centers, in proceedings of the 8th international workshop on middleware for grids, clouds and e-science, p 4
6. Bichler M, Setzer T, Speitkamp B (2006) Capacity planning for virtualized servers, in workshop on information technologies and systems (WITS), Milwaukee, Wisconsin, USA
7. Brugger B, Doerner K, Hartl R, Reimann M (2004) AntPacking—an ant colony optimization approach for the one-dimensional Bin Packing problem, evolutionary computation in combinatorial optimization, pp 41–50
8. Buyya R, Yeo CS, Venugopal S, Broberg J, Brandic I (2009) Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility. *Future Generat Comput Syst* 25(6):599–616
9. Buyya R, Broberg J, Goscinski, AM (2011) *Cloud computing: principles and paradigms*. Wiley Online Library
10. Chaisiri S, Lee B-S, Niyato D (2009) Optimal virtual machine placement across multiple cloud providers, in services computing conference, 2009. APSCC 2009. IEEE Asia-Pacific, pp 103–110
11. Clark C, Fraser K, Hand S, Hansen J, Jul E, Limpach C, Pratt I, Warfield A (2005) Live migration of virtual machines, in proceedings of the 2nd conference on symposium on networked systems design & implementation, vol 2, pp 273–286
12. Coffman EG Jr, Garey MR, Johnson DS (1996) Approximation algorithms for Bin Packing: a survey, in *approximation algorithms for NP-hard problems*, pp 46–93
13. Cormen T, Leiserson C, Rivest R, Stein C (2001) *Introduction to algorithms*. MIT press, Cambridge
14. Crosby S, Brown D (2006) The virtualization reality. *Queue* 4(10):34–41
15. David C (2008) *The definitive guide to the Xen Hypervisor*. Prentice Hall, Upper Saddle River
16. Dorigo M, Stutzle T (2004) *Ant colony optimization*. Mit Press, Cambridge
17. Fan X, Weber W, Barroso L (2007) Power provisioning for a warehouse-sized computer. *ACM SIGARCH Comput Archit News* 35(2):13–23
18. Feller E, Rilling L, Morin C (2011), Energy-aware ant colony based workload placement in clouds, in proceedings of the 2011 IEEE/ACM 12th international conference on grid computing, pp 26–33
19. Feller E, Morin C, Esnault A (2012) A case for fully decentralized dynamic VM consolidation in clouds, in cloud computing technology and science (CloudCom), 2012 IEEE 4th international conference on, pp 26–33.
20. Gao Y, Guan H, Qi Z, Hou Y, Liu L (2013) A multi-objective ant colony system algorithm for virtual machine placement in cloud computing. *J Comput Syst Sci* 79(8):1230–1242
21. Guazzone M, Anglano C, Canonico M (2011) Energy-efficient resource management for cloud computing infrastructures, in cloud computing technology and science (CloudCom), 2011 IEEE third international conference on, pp 424–431
22. He S, Guo L, Guo Y (2011) Real time elastic cloud management for limited resources, in cloud computing (CLOUD), 2011 IEEE international conference on, pp 622–629

23. Hermenier F, Lorca X, Menaud J-M, Muller G, Lawall J (2009) Entropy: a consolidation manager for clusters, in proceedings of the 2009 ACM SIGPLAN/SIGOPS international conference on virtual execution environments, ACM, New York, NY, USA, pp 41–50
24. Industry Perspectives (2013) Using a total cost of ownership (TCO) model for your data center. <http://www.datacenterknowledge.com/archives/2013/10/01/using-a-total-cost-of-ownership-tco-model-for-your-data-center/>. Accessed 2 Jan 2014
25. Jung G, Joshi K, Hiltunen M, Schlichting R, Pu C (2009) A cost-sensitive adaptation engine for server consolidation of multitier applications, *Middleware 2009*, pp 163–183
26. Jussien N, Rochart G, Lorca X (2008) The CHOCO constraint programming solver, in CPAIOR'08 workshop on open-source software for integer and constraint programming (OS-SICP'08)
27. Kaplan J, M Forrest W, Kindler N (2008) Revolutionizing data center efficiency. McKinsey & Company
28. Kivity A, Kamay Y, Laor D, Lublin U, Liguori A (2007) Kvm: the Linux virtual machine monitor, in proceedings of the Linux Symposium, pp 225–230
29. Kusic D, Kephart JO, Hanson JE, Kandasamy N, Jiang G (2009) Power and performance management of virtualized computing environments via lookahead control. *Clust Comput* 12(1):1–15
30. Levine J, Ducatelle F (2004) Ant colony optimization and local search for Bin Packing and cutting stock problems. *J Oper Res Soc* 55(7):705–716
31. Li B, Li J, Huai J, Wo T, Li Q, Zhong L (2009) EnaCloud: an energy-saving application live placement approach for cloud computing environments, in cloud computing, 2009. CLOUD'09. IEEE international conference on, pp 17–24
32. Lim MY, Rawson F, Bletsch T, Freeh VW, (2009), Padd: power aware domain distribution, in distributed computing systems, 2009. ICDCS'09. 29th IEEE international conference on, pp 239–247
33. Lo J (2005) VMware and CPU virtualization technology. World Wide Web electronic publication
34. Mell P, Grance T (2011) The NIST definition of cloud computing (draft). NIST special publication vol 800, pp 145
35. Meng X, Isci C, Kephart J, Zhang L, Bouillet E, Pendarakis D (2010) Efficient resource provisioning in compute clouds via VM multiplexing, in proceedings of the 7th international conference on autonomic computing, pp 11–20
36. Mi H, Wang H, Yin G, Zhou Y, Shi D, Yuan L (2010) Online self-reconfiguration with performance guarantee for energy-efficient large-scale cloud computing data centers, in services computing (SCC), 2010 IEEE international conference on, pp 514–521
37. Miller R (2013) Ballmer: Microsoft has 1 million servers. <http://www.datacenterknowledge.com/archives/2013/07/15/ballmer-microsoft-has-1-million-servers/>. Accessed 2 Jan 2014
38. Mishra M, Sahoo A (2011) On theory of VM placement: anomalies in existing methodologies and their mitigation using a novel vector based approach, in cloud computing (CLOUD), 2011 IEEE international conference on, pp 275–282
39. Nagarajan A, Mueller F, Engelmann C, Scott S (2007) Proactive fault tolerance for HPC with Xen virtualization, in proceedings of the 21st annual international conference on supercomputing, pp 23–32
40. Nelson M, Lim B, Hutchins G et al (2005) Fast transparent migration for virtual machines, in proceedings of the annual conference on USENIX annual technical conference, pp 25–25
41. Nguyen Van H, Dang Tran F, Menaud J-M (2009) Autonomic virtual resource management for service hosting platforms, in proceedings of the 2009 ICSE workshop on software engineering challenges of cloud computing, IEEE Computer Society, Washington, DC, USA, pp 1–8
42. Nurmi D, Wolski R, Grzegorzczak C, Obertelli G, Soman S, Youseff L, Zagorodnov D (2009) The eucalyptus open-source cloud-computing system, in cluster computing and the grid, 2009. CCGRID'09. 9th IEEE/ACM International Symposium on, pp 124–131
43. Smith J, Nair R (2005) Virtual machines: versatile platforms for systems and processes. Morgan Kaufmann, Burlington

44. Sotomayor B, Montero R, Llorente I, Foster I (2009) Virtual infrastructure management in private and hybrid clouds. *Internet Comput IEEE* 13(5):14–22
45. Speitkamp B, Bichler M (2010) A mathematical programming approach for server consolidation problems in virtualized data centers. *IEEE Trans Serv Comput* 3(4):266–278
46. Srikantaiah S, Kansal A, Zhao F (2008) Energy aware consolidation for cloud computing, in proceedings of the 2008 conference on power aware computing and systems
47. Takemura C, Crawford L (2009) *The book of Xen: a practical guide for the system administrator*. No Starch, San Francisco
48. Van Hein N, Tran F, Menaud J-M (2009) SLA-aware virtual resource management for cloud infrastructures, in computer and information technology, 2009. CIT '09. Ninth IEEE international conference on, pp 357–362.
49. Van Hein N, Tran F, Menaud J-M, (2010), Performance and power management for cloud infrastructures, in cloud computing (CLOUD), 2010 IEEE 3rd international conference on, pp 329–336.
50. Vaquero L, Rodero-Merino L, Caceres J, Lindner M (2008) A break in the clouds: towards a cloud definition. *ACM SIGCOMM Comput Commun Rev* 39(1):50–55
51. Verma A, Ahuja P, Neogi A (2008) pMapper: power and migration cost aware application placement in virtualized systems, in proceedings of the 9th ACM/IFIP/USENIX international conference on middleware, Springer-Verlag New York, Inc., New York, NY, USA, pp 243–264
52. Verma A, Kumar G, Koller R (2010) The cost of reconfiguration in a cloud, in proceedings of the 11th international middleware conference industrial track, pp 11–16
53. Verma A, Kumar G, Koller R, Sen A, (2011), Cosmig: modeling the impact of reconfiguration in a cloud, in modeling, analysis & simulation of computer and telecommunication systems (MASCOTS), 2011 IEEE 19th international symposium on, pp 3–11
54. Vogels W (2008) Beyond server consolidation. *Queue* 6(1):20–26
55. Voorsluys W, Broberg J, Venugopal S, Buyya R (2009) Cost of virtual machine live migration in clouds: a performance evaluation, *Cloud Computing*, pp 254–265
56. Walsh W, Tesauo G, Kephart J, Das R, (2004), Utility functions in autonomic systems, in autonomic computing, 2004. Proceedings international conference on, pp 70–77
57. Wood T, Shenoy P, Venkataramani A, Yousif M (2009) Sandpiper: Black-box and gray-box resource management for virtual machines. *Comput Netw* 53(17):2923–2938
58. Xu J, Fortes JA (2010) Multi-objective virtual machine placement in virtualized data center environments, in Green Computing and Communications (GreenCom), 2010 IEEE/ACM Int'l conference on & int'l Conference on Cyber, Physical and Social Computing (CPSCom), pp 179–188
59. Zhang Q, Cheng L, Boutaba R (2010) Cloud computing: state-of-the-art and research challenges. *J Internet Serv Appl* 1(1):7–18