Live Virtual Machine Migration in Dynamic Resource Management of Virtualized Cloud Systems

T.Mahiba
Dept.Information Science & Engineering
Srinivas Institute of Technology, Mangalore, India

Jayashree
Dept.Computer Science & Engineering
Srinivas Institute of Technology, Mangalore, India

Abstract— Live migration of virtual machines (VM) across distinct physical hosts is an important feature of virtualization technology. Virtual machine related features such as flexible resource provisioning, and isolation and migration of machine state have improved efficiency of resource usage and dynamic resource provisioning capabilities. Virtualization is a key concept in enabling the "computing-as-a-service" vision of cloud-based solutions. This paper presents the details of live virtual machine migration approaches and their usage in dynamic resource management of virtualized cloud environments. We outline components required for live virtual machine migration, details of migration heuristics and overview to reduce the total time required to migrate a running VM from one host to another.

Keywords – Virtual Machine, cloud computing, virtualization.

I. INTRODUCTION

Cloud computing [1] provides "cloud service models" in which compute resources are made available as a utility service — an illusion of availability of as much resources (e.g., CPU, memory, and I/O) as demanded by the user. Moreover, users of cloud services pay only for the amount of resources (a "pay-as-use" model) used by them. This model is quite different from earlier infrastructure models, where enterprises would invest huge amounts of money in building their own computing infrastructure. Typically, traditional data centers are provisioned to meet the peak demand, which results in wastage of resources during non-peak periods. To alleviate the above problem, modern-day data centers are shifting to the cloud. [2]

The important characteristics of cloud-based data centers are:
• *On demand service* Makes resources available on demand. The operation and maintenance of the data center lies with the cloud provider. Thus, the cloud model enables the users to have a computing environment without investing a huge amount of money to build a computing infrastructure.
•*Elastic resource provisioning*. This provides ability to dynamically scale or shrink the provisioned resources as per the dynamic requirements.
• *Metered service* This enables the "pay-as-use" model, that is, users pay only for the services used and hence do not need to be locked into long-term commitments.

However, implementing cloud-based data centers requires a great deal of flexibility and agility. For example, the dynamic scaling and shrinking requirement needs *compute resources* to be made available at very short notice. When computing hardware is overloaded, it may be required to dynamically transfer some of its load to another machine with minimal interruption to the users. *Virtualization technology* can provide these kinds of flexibilities.

Efficient *resource management* is a very critical component in cloud-based solutions. *Virtualization* is a popular solution that acts as a backbone for provisioning requirements of a cloud-based solution. Virtualization provides a "virtualized" view of resources used to instantiate virtual machines (VMs). A VM monitor (VMM) or hypervisor manages and multiplexes access to the physical resources, maintaining isolation between VMs at all times. As the physical resources are virtualized, several VMs, each of which is self-contained with its own operating system, can execute on a physical machine (PM). The hypervisor, which arbitrates access to physical resources, can manipulate the extent of access to a resource (memory allocated or CPU allocated to a VM, etc.).

The decoupling between physical and virtual resources provided by the hypervisor enables flexibility of resource provisioning for VMs. This decoupling also yields efficient state (more correctly memory state) capture of VMs, which enables migration and restoration of virtual machines across physical machines.

II. VIRTUAL MACHINE

The term "virtualization" refers to the process of turning a hardware-bound entity into a software-based component. The end result of such procedure encapsulates an entity's logic and is given the name of Virtual Machine (VM). The main advantage of this technique is that multiple VMs can run on top of a single physical host, which can make resource utilization much more efficient. A VM, like a PM, has associated resource levels of CPU, memory, and input/output (I/O) devices. These resources can be overcommitted, and multiplexing them across VMs is the hypervisor's responsibility. Initial provisioning- levels of a VM is determined by a "sizing" process which is based on resource-usage profiles of applications or estimations to meet load requirement

Virtual machine migration approaches

Virtual machine migration is the process of transferring the state (all memory pages) of a VM from one physical machine to another. Different approaches for live migration are

Pure stop-and-copy- The source VM is stopped, pages are copied across to the destination VM, then the new VM is started. In this the downtime and total migration time are proportional to the amount of physical memory allocated to the VM. This can lead to an unacceptable outage if the VM is running a live service.

Pre-copy -The contents of the VM's memory are first sent to the target host and then the VM is restarted. To keep the downtime, i.e., the time during which the VM is not running, to a minimum, data is sent in several iterations while the VM keeps running on the source host. In each following iteration, only the pages that have been modified

since the last round are sent. [4]

Post-copy -Here, only the VM's VCPU and device state is sent to the target host and restarted there immediately. Memory pages accessed by the VM are then fetched in parallel and on-demand while the VM is running on the target host.[5]

Precopy and postcopy approaches minimize the downtime of the VM at the expense of the total migration time, i.e., the time from when the migration is started until the VM runs independently on the target and can be destroyed on the source host. The vitual machine migration procedure is depicted in Figure.1

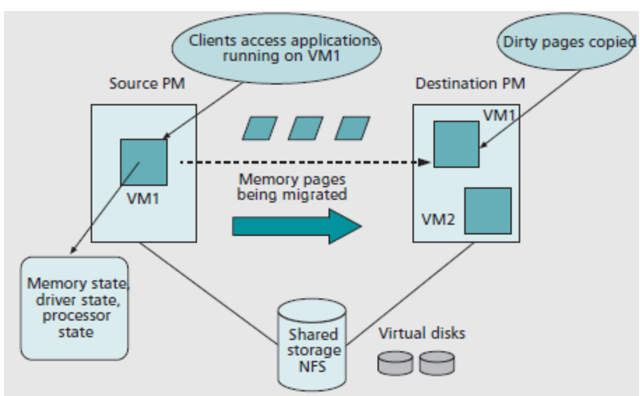

Figure 1.Live Virtual Machine Migration Scenario

III. DYNAMIC RESOURCE MANAGEMENT USING VIRTUAL MACHINE MIGRATION

Virtual machine migration is a key enabler for dynamic resource management in cloud-based systems. Figure.3 depicts the important components of resource management from a cloud provider's perspective.

Steps in resource management

- Virtual machine sizing, the first step toward deploying a VM, is to determine the expected resources required on deployment

-Once these levels are determined, a provisioning step determines where to "place" the VM and instantiate it.

- Once a VM is instantiated, a resource monitoring engine tracks the resource usage and performance indicators related to the (applications of the) VM.

Under dynamic workload conditions, VMs can experience “hot spots” (inadequate resources to meet performance demands) and “cold spots” (over provisioned resources with low utilization). Moving VMs in order to allocate more resources (to alleviate hot spots) or consolidate VMs on fewer PMs to tackle cold spots is enabled through migration. From a cloud provider’s perspective, alleviating hot spots is essential to meet SLAs with clients and tackling cold spots to use resources (including power consumption) efficiently.[6]

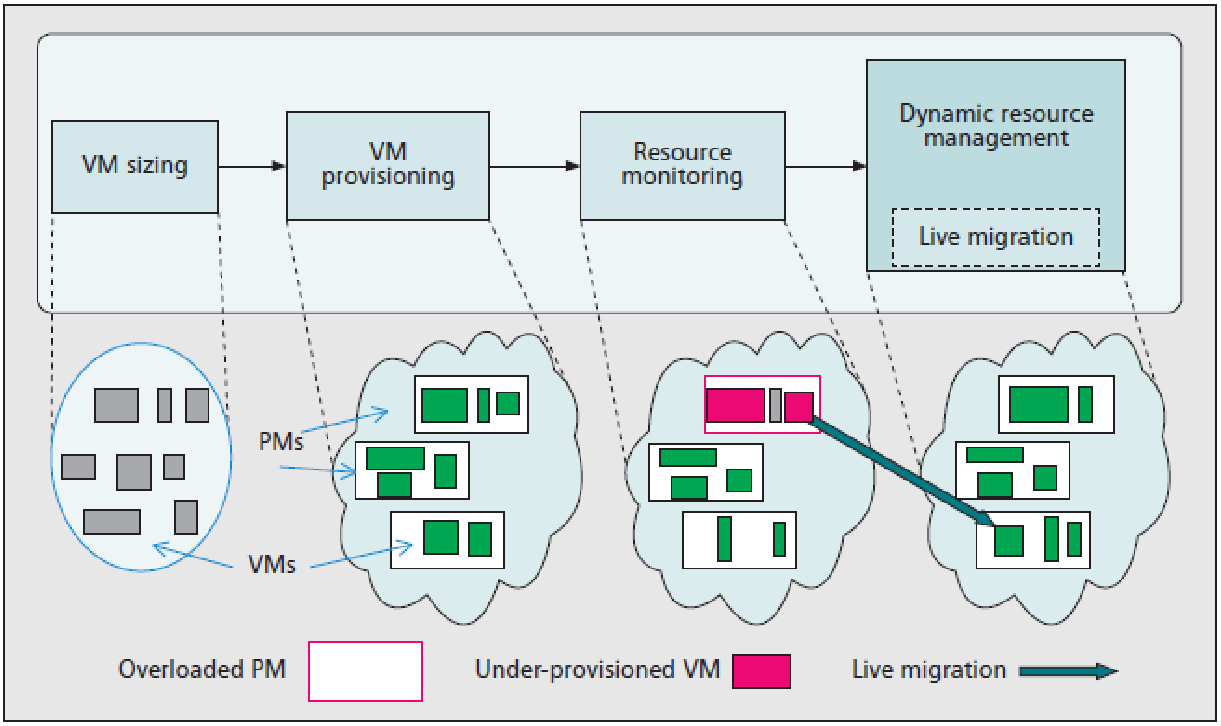

Figure.3 Steps in Resource Management

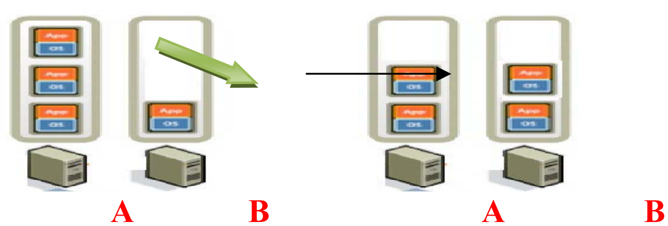

Figure.2(a) Load Balancing

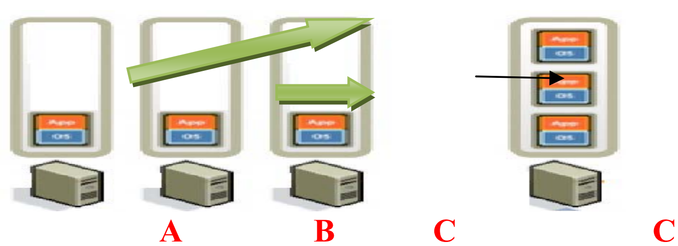

Figure.2(b) Server Consolidation

Figure 2 shows two conditions: load balancing and consolidation of VMs based on migration. In the first case, either the goal is to distribute “load” evenly across PMs, or a VM needs more resources and hence is migrated to another PM. With consolidation, machines are migrated to fewer PMs to reduce *server sprawl*.

A cloud provider's resource management actions toward simultaneously minimizing resource usage and maximizing SLA adherence can be classified as follows:

Server consolidation: The goal of consolidation is to avoid server sprawl — many PMs host low-resource-usage VMs. As shown in Fig.2 VMs on lightly loaded hosts can be "packed" onto fewer machines to meet resource requirements. The freed-up PMs can either be switched off (to save power) or represent higher-resource availability bins for new VMs.

Load balancing: The goal of load balancing is to avoid a situation where there is a large discrepancy in resource utilization levels of the PMs. A desired scenario could be to have equal residual resource capacity across PMs (to help increase local resource allocations during increase demands). Virtual machine migrations can be employed to achieve this balance.

Hotspot mitigation: Active resource and application-level monitoring of VMs if required to identify hot spot conditions in which a VM has inadequate resources to meet its SLA requirements. Under such conditions, additional resources can be allocated either locally (on the same PM) or within the set of PMs available for provisioning. When local resources are not sufficient to remove the hot spot, VMs can be migrated to another host to make the resources required available to mitigate the hot spot.

IV. HEURISTICS FOR RESOURCE MANAGEMENT USING MIGRATION

For each of the three goals — consolidation, hotspot mitigation and load balancing — VM migration-based heuristics need to address three important questions:

- When to migrate
- Which VMs to migrate
- The set of destination host machines for migration.

Migration heuristics address each of these questions based on several constraints: overhead of the migration process, impact on applications during migration, degree of improvement in intended goals of performance of resource utilization, and so on.

When to migrate

There are many situations when migration of VMs becomes necessary to maintain the overall efficiency of the data center. ***Periodic*** — The migrations in a data center can be triggered periodically. For example, data centers in one part of world may be heavily used in daytime (9 a.m. to 9 p.m.), whereas they may be underloaded during the night. Such "time of day" migration of VMs ensures that VMs are "near" clients, and the communication delays and overheads are minimized. Migrations can also be done periodically to consolidate the reduced loads.

Due to Hot Spot — A hot spot is the overloaded condition of a PM. It can also be defined as the state when performance of a system falls below the minimum acceptance level. Detection of a hot spot can be done both proactively and reactively. Proactive hot spot detection techniques predict the occurrence of a hot spot by analyzing the trends in resource utilizations of the VM. If the resource utilization shows an increase for some time window, it is likely that it may result in a hot spot in the future. Such time series analysis-based techniques help avoid hot spots even before they occur. One such technique to predict CPU utilization is discussed in [6]. More sophisticated proactive techniques analyze the request arrival rates. Increase in request arrivals suggests that the VM will require more resources to fulfill them, thus causing a potential hot spot. Reactive hot spot detection techniques use more direct techniques like observing the page thrashing rate, CPU and memory utilization levels, and so on. Hot spots can be locally mitigated if enough capacity is available at the host PM. Extra resources can be allocated to the VM showing signs of overload. When extra capacity is not available locally, migration is the only option available.

Excess Spare Capacity — Low utilization of PMs results in resource wastage. An optimum level of utilization is required to be maintained for the efficient working of a data center. Physical machines that have excess spare capacity (i.e., low resource utilization) cause overall inefficiency in the data center. At the level of a PM, the hypervisors have monitoring tools, similar to normal operating systems, which can provide the utilization information of different resources for that machine. Resource utilization levels of PMs across a data center are continuously monitored, and whenever the utilization levels fall below a certain threshold, migrations can be triggered. When a number of PMs are underutilized, VMs are migrated from such machines to make them completely free. Such "freed" PMs can then be shut down to save power, which results in consolidation.

Load Imbalance — Virtual machines change their resource requirements dynamically. This dynamism leads to imbalances in the resource utilization levels of different PMs. Some PMs can get heavily loaded while others may be lightly loaded. In a data center, resource utilization levels of PMs are monitored continuously. If there is large discrepancy in the utilization levels of different PMs, load balancing is triggered. Load balancing involves migration of VMs from highly loaded PMs to low loaded ones. An overloaded PM is undesirable as it causes delays in service of user requests. Similarly, the PMs that are lightly loaded cause inefficient resource utilization.

Addition/Removal of Virtual Machines and Physical Machines — Virtual machines and PMs can be added and removed in a virtualization- based data center. Addition/removal of VMs and PMs affects the availability of the resources and may require a change in the placement plan of VMs. A new PM can be used to offset the load of an overloaded PM by migrating VMs from the latter to the former. Similarly, hosting new VMs may result in future overloads of some PMs, which again require migrations to be triggered.

Which virtual machine to migrate

Selecting one or more VMs for migration is a crucial decision of the resource management heuristic. The migration process not only makes the VM unavailable for a certain amount of time but also consumes resources like network and CPU on source and destination PMs. Performance of other VMs that are hosted on source and destination PMs are also affected due to increased resource requirements during migration. Some VM selection approaches are straightforward and only consider the VM that is resource constrained (e.g., in a hot spot); other approaches employ a more holistic approach where all the VMs on a PM are considered before selecting the candidate VM. Generally, the aim of VM selection is to minimize the migration effort.

Resource Constrained Virtual Machine — This is the easiest way to select the candidate VM for migration. The VM whose resource requirements cannot be locally fulfilled is selected for migration. During hot spots, it is easy to find the most loaded VMs; hence, this simple selection can work. However, in operations like consolidation and load balancing, where the cause is not a single VM, the choice is not straightforward.

Affinity based — These heuristics also incorporate other objectives instead of considering the resource requirement only. For example, some *affinity-aware* migration techniques consider communication costs among VMs while performing migration. For instance, if two VMs are communicating with each other, it is better to host them on the same PM. This will reduce the overall communication cost among the VMs by reducing network usage. Similarly, memory sharing between VMs can also affect the VM selection for migration. Migrating a VM to a PM where it can share memory with other hosted VMs can result in effective memory usage [6]. Virtual machines, which share memory, can be migrated together with less effort as similar-content memory pages are required to be transferred only once. Such a scheme is known as *gang scheduling* of VMs. The approach is to proactively track the identical contents of collocated VMs and transfer those contents only once while migrating all those VMs simultaneously to another PM. This method optimizes both memory and network overhead of migration. Such mechanisms can be fruitful when an entire rack of servers have to be evacuated and all the collocated VMs running on them have to be shifted to a different location. [7]

Where to migrate

During migration, the destination PM should have enough resources so that it can support the incoming migrating VM. Here we discuss factors for selecting a PM as a destination for a migrating VM.

Depending on Available Resource Capacity— Only considering the availability of resources at the destination is not enough. Some other factors also need to be taken into consideration, such as whether the destination is a *best fit* (leaving minimum remaining resources) for the migrating VM, how will the performance of VMs that are already hosted on destination PM get affected. The destination selection to minimize waste of resources is a field of research in itself. The schemes proposed use heuristics of bin packing and vector packing problems [8, 9] for destination selection since the optimal placement solution is intractable. For example, in vector-based destination selection, vector arithmetic like dot product is performed on resource vectors to find the best fit [10]. Virtual machines and PMs are sorted in some order based on their resource requirements and then the First Fit or Best Fit scheme is applied to select most suitable PM.

Depending on Affinity of Virtual Machines
— Apart from selecting PMs solely on the basis of resource availability, some schemes try to leverage the relations (or affinity) between the VMs to identify a suitable host PM. For example, a scheme mentioned in [10] tries to achieve consolidation by collocating VMs that have high memory sharing potential. Periodically, based on memory fingerprints of VMs, best matches of hosts for VMs can be found and migrations can be triggered. This scheme is called *memoryaware* migration. The VM can be remigrated if some other VM on some other PM becomes a better memory sharing partner. The overhead of migration is taken into consideration. The rationale behind this method is that VMs that can share part of their memory will require less overall memory than VMs that do not share memory. Similarly, if two VMs, hosted on different PMs, communicate heavily, one of the VMs can be migrated to the PM where its communicating partner is hosted.

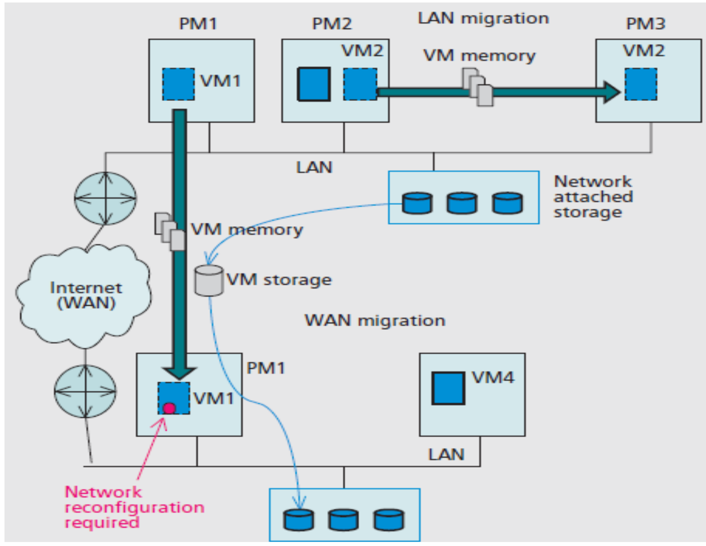

Figure.4 Virtual Machine Migrations Over LAN and WAN

V. CHALLENGES IN LIVE VIRTUAL MACHINE MIGRATION

There are two challenges in live virtual machine migration

- Virtual machine migration in wide area networks
- Efficiently migrating virtual machine with minimum downtime

Wide-area virtual machine migration

The discussion regarding live migration so far has considered resource management within a single (local) virtualization-based hosting setup. In such a setting, the VMs (and PMs) are assumed to be located within the same local area network. In a cloud-based provisioning scenario this assumption may not always hold; VM hosting data centers can be spread around the world, or enterprises with offices worldwide may have private data centers at several locations. In such cases, migrating VMs over wide area networks poses a set of different and interesting challenges. The main differences between VM migration over local and wide area networks are as follows.

Migrating storage: Machines within a local network connect to shared storage system using mechanisms such as NFS, NAS, and storage area networks (SANs). In such a setting, since access to disks is primarily via the network, migration of VM is accomplished by migrating the memory state. In a wide-area scenario, the shared storage assumption seldom holds. As a result, wide-area migration entails not only transferring memory state, but more important, the state of local disks. With the lack of high end-to-end network bandwidth over wide area networks and the potential transfer of large storage state, the downtime and migration time are expected to be high in such scenarios.

Network reconfiguration: Unlike migration within a local area network, crossing network boundaries results in network reconfiguration. Moving into a new subnet forces the machine to get a new IP address and, as a result, breaks existing network connections. Either the network addresses have to be preserved or applications need to be made aware of the network reconfiguration semantics. Figure.4 depicts the local-area and wide-area migration scenarios. While VMs assume availability of networked storage systems in local networks, this condition is hard to recreate over wire area networks. The limited end-to-end bandwidth makes such an equivalent exercise futile. Furthermore, IP reconfiguration will not support state-full network protocols.

Efficiently migrating live virtual machine

Migrating a running VM from one physical host to another requires that the entire state of the VM is transferred. The state of a VM comprises the VCPUs, the configuration of the drivers, the VM's, memory and the permanent storage. In data center setups, the permanent storage is typically a network storage device. Consequently, the contents of permanent storage do not need to be moved to the target host. With memory sizes of several gigabytes even for virtual machines, transferring the memory state to the target host thus becomes the bottleneck of live migration.

Modern operating systems cache data from permanent storage in unused volatile memory to hide the long access latency. The longer the system is running, the bigger an amount of otherwise unused memory is dedicated to this cache. In data centers, the physical hosts are typically attached over a very fast network to a network storage device as well as to the external world. During live migration, a lot of data needs to be sent from one physical host to another. In order not to affect the quality of service of the whole data center, the maximum bandwidth with which data is sent between distinct physical hosts is usually limited. This can easily lead to migration times of tens of minutes for VMs that have several gigabytes of main memory. The proposed goal is thus to detect duplicated data and fetch this data directly from the attached storage device. This considerably reduces the amount of data sent between the two hosts involved in live migration and has the potential to significantly shorten the total migration time.

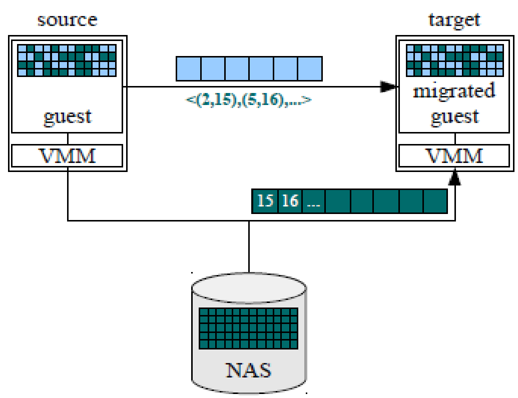

Figure.5 Duplicated memory pages are directly loaded from external storage

The key idea of the proposed method is to fetch memory pages that also exist on the attached storage device directly from that device instead of transferring them directly from the source to the target host. Figure 5 illustrates the idea: memory pages that have been recently read from or written to external storage and are thus duplicated are shown in green. Such pages include pages from code and read-only data sections of running applications and cached disk blocks. Shown in blue are memory pages which contain data that is not duplicated. To detect duplication between memory pages and storage blocks, we transparently track all I/O operation to the attached storage and maintain an up-to-date map of duplicated pages. When migrating a VM across distinct physical hosts, instead of sending the data of all dirty memory pages to the target host, only the data of ordinary (i.e., not duplicated) memory pages is sent. For pages that also exist on the attached storage device, the mapping memory pages to disk blocks is sent to the target host from where the data is fetched by a background process.

VI. CONCLUSION

In this paper, the important role of live virtual machine migration in dynamic resource management of virtualized cloud systems is discussed. Migration enables several resource management goals like consolidation, load balancing, and hot spot mitigation. Researchers have leveraged live virtual machine migration to come up with efficient resource management mechanisms. The components — when to migrate, which VM to migrate, and where to migrate — and approaches followed by different heuristics to apply migration techniques for goals of consolidation and hot spot mitigation were discussed. Challenges with wide-area migration, efficiently migrating virtual machine with minimum downtime and some techniques to enable it were also discussed.

REFERENCES

[1] M. Armbrust *et al.*, “A View of Cloud Computing,” *Commun. ACM*, vol. 53, no. 4, 2010, pp. 50–58.

[2] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz,Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia “Above the Clouds: A View of Cloud Computing” *UC Berkeley*

[3] Zhen Xiao, *Senior Member, IEEE,* Weijia Song, and Qi Chen ”Dynamic Resource Allocation using Virtual Machines for Cloud Computing Environment” IEEE Transaction On Parallel And Distributed Systems

[4] Diego Perez-Botero, Princeton University “A Brief Tutorial on Live Virtual Machine Migration From a Security Perspective”

[5] M. Mishra and A. Sahoo, “On Theory of VM Placement: Anomalies in Existing Methodologies and Their Mitigation Using a Novel Vector Based Approach,” *Proc. 4th Int’l. Conf. Cloud Computing*, 2011, pp. 275–82.

[6] T. Wood *et al.*, "Black-Box and Gray-Box Strategies forVirtual Machine Migration," *Proc. 4th Conf. Symp.Networked Sys. Design & Implementation*, 2007.

[7] U. Deshpande, X. Wang, and K. Gopalan, "Live Gang Migration of Virtual Machines," *High-Performance Parallel and Distributed Computing*, June 2011.

[8] M. Mishra and A. Sahoo, "On Theory of VM Placement: Anomalies in Existing Methodologies and Their Mitigation Using a Novel Vector Based Approach," *Proc. 4thInt'l. Conf. Cloud Computing*, 2011, pp. 275–82.

[9] R. M. Karp, M. Luby, and A. Marchetti-Spaccamela, "A Probabilistic Analysis of Multidimensional Bin Packing Problems," *Proc. 16th Annual ACM Symp. Theory of Computing*, 1984, pp. 289–98.

[10] T. Wood *et al.*, "Memory Buddies: Exploiting Page Sharing for Smart Colocation in Virtualized Data Centers," *Proc. ACM SIGPLAN/SIGOPS Int'l. Conf. Virtual Execution Environments, VEE*, 2009, pp. 31–40.