Dynamic Scheduling of Resource Based on Virtual Machine Migration

Kamakshi R[1]
Department of Computer Science and Engineering
Adithya Institute of Technology, Coimbatore
INDIA
kamakshi619@gmail.com,

Sakthivel A[2]
Department of Computer Science and Engineering
Adithya Institute of Technology, Coimbatore
INDIA
asakthivel75@yahoo.com

*Abstract: A*n improved method is proposed for achieving high utilization of server resources and minimizing power usage for cloud Environment. The cloud allows cloud user to scale up and down their resource usage based on the needs. Due to multiple multiplexing and sharing of resource is to decrease the resource utilization of the system and also increase the power utilization. The virtual machine (VM) migration is used to increase the server resource utilization and minimize number of primary machines (PM) is used; it helps to eliminate the hotspots to improve overall performance and minimize the power utilization. It is very useful to various autonomic environments for handling different types of workload efficiently and effectively.

Key-word: Cloud computing, resource management, scheduling techniques, Virtual machine migration.

1 Introduction

A user community who operates on common characteristics undergoes a process of transition. The transition takes place traditional environment to virtual clouds (VCs) environment [1]. The VCs provide many benefits such as Rapid Service, Secure Service, Satisfying User Experience, Lower Costs, Multi User Access, Development Platform and Infinite Storage. The virtual cloud environment [2] is the internet-based storage for files, applications and infrastructure are called as physical resources that provides huge amount of features like as streamline process (Netflix, hulu, big thinks, currentTV, Vimeo), Deploying projects faster (Google Docs), globalize your workspace on the cheap and best (online games), Video conferencing (yahoo messenger, facebook, Hangouts), Synchronization of data across device (Google calendar). Even though it has many benefits and features, it faces some difficulties while allocating those resources to cloud user because lack of mapping virtual machines (VMs) to physical resources. This lack of virtual mapping is caused by unnecessary automatic scale up and scale down of user request in dynamic environment is called work load variation. Due to this work load variation causes resource allocation problem in dynamic changing environment.

The resource allocation is a strategy which may be adopted in adverse circumstances. This problem effectively deals with indirect virtual machine that remains alive and has a movement of mapping from one part to another [3] [4]. The live migration is control by virtual machine monitor like Xen [5] [6] provides a mechanism for mapping virtual machines (VMs) to physical resources [7]. Here the cloud service provider is the physical machines (PMs) have sufficient resources to meet their needs [8]. VM live migration technology makes it possible to change the mapping between VMs and PMs While application running [9] [10]. However it decides the virtual mapping in dynamic network, PMs can be overloaded for satisfying user needs. It may cause sometimes resource overload from another PMs, the resource overload from PMs to PMs and VMs to VMs is called hotpots. The higher chance of hotspots to decreases the overall performances, typically to eliminate hotspot does not need to be moved entire permanent storage to target host. While overloading VMs to another PMs requires that the entire state of the VMs is transferred. The state of VMs comprises all the details of the configuration of drivers and permanent storage. The remapping of status work is to improve overall performance and eliminates hotspots.

The reason for the suggested proposal is to refine all inclusive process of performing a tasks is to improve to eliminate hotspots and possibly achieve power energy conception by minimizing number of PMs [11] [12]. To eliminate the hotspot is used to increase overall performance and to achieve overload avoidance. The benefit of green computing is achieved by enabling sleep mode of idle machines [13] [14] [15] [16]. The VM migration algorithm selects a destination server (i.e., it favors low load) has low load, their algorithm tends to spread the VMs onto idle servers during offloading. Virtual machine live migration provides excellent platform for achieving overload avoidance, improving overall performance, minimizing the downtime and reducing the operating cost. It can improve their scalability in dynamic environment and it can more suitable for wide area network.

The rest of the paper is structured as follows: Section 2 Survey of Literature, Section 3 Related Work and Section 4 Proposed algorithm for dynamically scheduling of resource. The Experimental setup and results is discussed in the Section 5, Section 6 Performance analysis and finally Section 7 concludes.

2 Survey of Literature

Christopher Clark et al [4] have achieved impressive towards the performance along with minimal service downtimes as low as 60ms. They analyzed the concept of X86 virtual machine [1] monitor Writable working set principle [17] for high performance of VM migration [18]. It also separates hardware and software consideration for avoiding problem of residual dependencies [19]. It provides a rapid movement of interactive workload within cluster of data center. Virtual machine migration [20] concept applications are Quake3 game server, Netflix, yahoo screen. This is faster because their search space is smaller, but it may not work correctly under some scenarios. This ends the limitation of writable working set.

Timothy Wood et al [3] have used Virtual machine migration to eliminate hotspots in distributed network environment using sandpiper that proposes two concepts named as black box and white box approaches. Hotpot is the PM overloaded for satisfying user requirements. Black box eliminates hotspots between fully OS and application level agnostic [11]. White box eliminates hotspots between exploits OS and application level statistics [17]. The black box white box algorithms combine multi-dimensional load information into a single Volume metric [3]. It sorts the list of PMs based on their volumes and the VMs in each PM’s volume-to-size ratio (VSR). Sandpiper automates the task of monitoring and detecting hotspot. It also determines a new mapping of virtual resource and finally takes the decision for migration.

David Meisner et al [13] have used PowerNap as an energy conservation approach for heavy load environment. The goal is to obtain minimal idle power and low transition time. PowerNap eliminates the idle power waste in enterprise blade server. It uses the RAILS which provides high energy conservation. RAILS mean Redundant Array for Inexpensive load sharing. PowerNap with RAILS will reduce average server power consumption by 74%. PowerNap eliminates idle power in server by quickly transitioning in and out of ultra-low power state. It improves power conversion efficiency and reduces the cost.

RipalNathuji and Karsten Schwan [16] mentioned how to integrate power management mechanism and policies with the virtualized technology. The goal is to support the isolated independent operation and to control globally the coordinate effects of diverse power management [14][15]. Infrastructure is employed to support rich and effective policies for ensuring fair power allocations across different VMs. It prevents power viruses and used for Intel Core micro architecture. The advantage of virtual power approach is to ensure fair power allocation across different VMs.

3 Related Works

In the recent years, Cloud-based applications have more and more attention in, more and more servers are needed to provide services. In the scenario scheduling is the major problem to satisfy the user request. To meet the user requirements the primary machine can be overloaded (hotspot) [21]. The higher chance of hotspot is to decreases the overall performances and utilization of server resources. In other end the primary machine can switch on during un-utilization of server resources. This case leads to provide high loss of power energy.

To resolve the above problem we introduce dynamic resource scheduler for scheduling the hardware resources to the target host and mapping the state (with memory size of several gigabytes) of the VMs to improve the utilization of server resource. The scheduler detects the hotspot and it eliminates the hotspot. The elimination of the hotspots improves overall performances and utilization of server resources. The scheduler is not only detects and eliminates hotspots it also detects cold spots. The detection of cold spots is used to helps for saving the power energy while switch off the idle machine.

To obtain the solution for the problem using VM migration algorithm this is based on the skewness algorithm [21]. The VM migration algorithm consists of two parts: hot spot mitigation, and green computing. Let p be the number of (CPU, memory, I/O, etc.), k be the virtual machine, k_i be the utilization of the i^{th} resource by k and $\bar{k}$ is the average utilization of all resources for server h. $\bar{k}$ can be defined as:

$$\textit{Average utilization}\,(\bar{k}) = \frac{\sum_{i=1}^{p} k_i}{p} \qquad (1)$$

The resource hotspot of a server h can be defined by using equation (1).

$$\textit{Hotspot (h)} = \sqrt{\sum_{i=1}^{p} (\frac{k_i}{\bar{k}} - 1)^2} \qquad (2)$$

In real time, equation (2) is not fair for all the type of resources. It is only suitable for bottleneck resources. The hotspot (h) value should always less than or equal to 1 (i.e. by minimizing the hotspot value), we can combine different types of workloads nicely and improve the overall utilization of server resources. The maximum hotspot of the server can be determined by using degree of the server can be defined as:

$$\textit{Degree (d)} = \sum_{k \in K} (k - k_d)^2\,, \quad k_d \le 1 \qquad (3)$$

Here K is the set of overloaded resources in server h and k_d is the hot threshold for resource k. The temperature of a hot spot reflects its degree of overload. It can be derived by using equation (3). The resource cold spot of a server h can be defined as:

$$\textit{Coldspot (c)} = 1 - \sqrt{\sum_{i=1}^{p} (\frac{k_i}{\bar{k}} - 1)^2} \qquad (4)$$

The parameter k_i and $\bar{k}$ are as same as hotspot. By using equation (1) we can derive cold spot value of server (c). From the equation (4), the cold spot value of a server is calculated. If the value is 0, that server is the idle mode. So it changes to switch off mode for saving power energy. From the rule of probability based on complement probability, the relationship of the hotspot and cold spot can be defined as follow as:

$$\textit{Hotspot (h)} + \textit{Coldspot (c)} = 1 \qquad (5)$$

From the equation. (5) the server fall in to four different cases as follow as

Case 0: If Hotspot (h) = 0, the Coldspot (c) value goes to 1. At the time the server can automatically change into switch off mode.

Case 1: If Coldspot (c) = 0, the Hotspot (h) value goes to 1 or may be more than one with respect to network work load.

Case 2: If Hotspot (h) = 0, Coldspot (c) = 0, that the case is not applicable.

Case 3: If Hotspot (h) = 0, Coldspot (c) = 0, that the case is not applicable.

The case 0 and case1 is easily addressing the instant changes of the server for the dynamic scheduling.

4 Proposed Work

The proposal work is to improve overall performance by eliminating hotspots and possibly achieve power energy conception using VM migration algorithm. The scheduler plays a major role in proposal work. It plays two roles: hotspot detector and migration manager. The schedulers are used to construct resource usage profiles for each virtual server and aggregate profiles for each physical server. It detects and monist continuously the constructed usage profiles for identifying the hotspots. If the scheduler detects a hotspot, instantly it takes a decision to which VM can be migrated. If the scheduler detects a hotspot, instantly it takes a decision to switch off the corresponding VM, while switch off the VM we can achieve power energy, here it is called as green computing.

The proposal method is to avoid the residual dependency because here no pre-copy when hotspot is occurred. This method is completely eliminates hotspot and maintains VM don't move to hotspot mode further. Here green computing is achieved when no request will arrive. So it effectively improves overall performance, achieve quality of service and increase utilization of server resources.

4.1 Algorithm

Step 1. Initialize the number of VM and number of request.

Step 2. Calculate the hotspot value and pass to scheduler.

Step 3. Call the scheduler function for checking whether the incoming value is hotspot value or cold spot value.

Step 4. If it is the condition true, the scheduler again checks degree of the VM for how many times migration can be taken place.

Step 5. If it is false, the scheduler decides to process the cold spot and make the corresponding VM is to be switch off.

Step 6. The step 2 to step 3 is repeated for every request.

Using this algorithm we can easily detects and monitors continuously the constructed usage profiles for identifying the hotspots. If the scheduler detects a hotspot, instantly it takes a decision to which VM can be migrated. The algorithm provides benefit of green computing is achieved by enabling sleep mode of idle machines. The VM migration algorithm selects a destination has low load, the algorithm tends to spread the VMs onto idle servers during offloading. This algorithm decrease the higher chance of hotspots to increases the overall performances, it improves overall performance and to eliminates hotspot and possibly achieve power energy conception because the algorithms minimizing number of PMs is used. Virtual machine live migration provides excellent platform for achieving overload avoidance, improving overall performance, minimizing the downtime and reducing the operating cost. It can improve their scalability in dynamic environment and it can more suitable for wide area network.

5 Experimental Setup and Result Analysis

For the attainment of the objectives prescribed in our project, we have used different hardware components and tools of software available. They mainly include the java programming language run in the Net-Bean platform for the web pages. The hardware requirements are as Processor Intel core TM 2 duel core, its speed is 1.1 GHz, memory is256 MB (min) RAM and Hard Disk memory is 20 GB. Key Board Standard Windows Keyboard, Mouse Two or Three Button Mouse, Monitor SVGA. The software requirements are as Front End is Java SE 1.6, Tools/ IDE-Net Beans 6.9, Technologies Used as Java and J2EE, Web Technologies-Html, JavaScript, CSS, Operating System can be a Windows XP/7. The resources are CPU utilization, I/O utilization, memory utilization. User request is the input of the project and processing the request is the output of the project.

The parameter p be the number of resources; h be the hotspot values; c be the coldspot values; d be the degree of the server and $\bar{k}$ be the average utilization. The maximum resources are used as *CPU, I/O, and MEMORY* in this project. The value of $h, c, d, \bar{k}$ are calculated by using above mentioned mathematical equations.

Table 1 shows Average Utilization of resources

S. No	k	P			$\sum k_{ji}$	$\bar{k}$
	j↓→ i	*CPU*	*I/O*	*MEMORY*		
1	1	0.1	0.1	0.61	0.81	0.27
2	2	0.1	0.1	0.2	0.5	0.1
3	3	0.6	0.6	0.05	1.25	0.41
4	4	0.3	0.3	0.2	0.8	0.26

The average utilization of resource and values as drawn in the Table 1 which is calculated by using mathematical equation Eq. (1). The table is also expressing the summation of values of k_{ji}. The Table 2 shows hotspot value of the virtual machine is calculated by using Eq. (2) through the help of Table 2 average utilization of the resources.

Table 2 Shows Hot spot value of resources

S. No	k	p			$\bar{k}$	h
	j↓→ i	1	2	3		
1	1	0.1	0.1	0.61	0.27	1.54
2	2	0.1	0.1	0.2	0.1	1
3	3	0.6	0.6	0.05	0.41	1.01
4	4	0.3	0.3	0.2	0.26	0.31

The hotspot value h is always not exceeding 1. In above table VM2 VM4 is in normal mode because their h values are 1and 0.31 respectively. The VM1 and VM3 is in hotspot mode because their h values are 1.54 and 1.01 respectively

Table 3 Shows Cold spot value of resources

S. No	k	p			h	c
	j↓→ i	1	2	3		
1	1	0.1	0.1	0.61	1.54	0.54
2	2	0.1	0.1	0.2	1	0
3	3	0.6	0.6	0.05	1.01	0.01
4	4	0.3	0.3	0.2	0.31	-0.69

The parameter c is the cold spot value of the virtual machine and their values as drawn in the Table 3 which is calculated by using mathematical equation Eq. (4) through the help of Table 3 hotspot values. The cold spot value c is always not exceeding 0. In above table VM2 VM4 is in cold spot mode because their c values are 0and -0.69 respectively. The VM1 and VM3 is becoming cold spot mode because their h values are 0.54 and 0.01 respectively.

6 Performance Analyses

6.1 Average utilization of resource

For average resource utilization, let $p\bar{k}$ be the number of average utilization in the system during a decision. The complexity of each VM's average resource utilization takes O ($p\bar{k}$_log($p\bar{k}$)). To sort the resource based on the average utilization. The Figure 1 shows average utilization of resources by the virtual machines wise VM1, VM2, VM3, VM4. In x-axis, it takes number of virtual machines. In y-axis, it takes average utilization of resources. From this analysis can identify the full-fledged resources utilization. The virtual machine 3 uses the resource CPU and I/O highly compared with others. The virtual machine 1 uses the resource Memory highly compared with other. The virtual machine 2&3 use the all resources medium compared with other virtual machine.

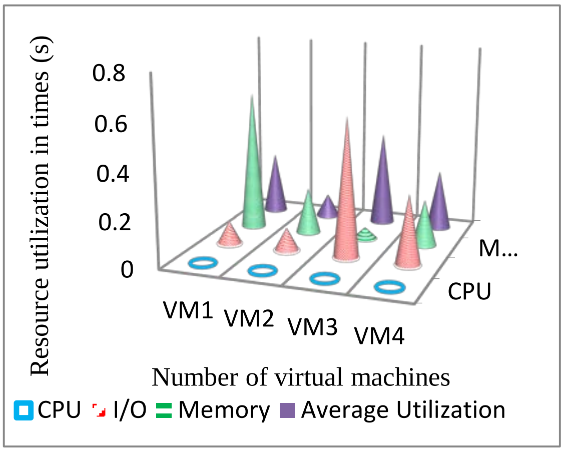

Figure 1: Average Utilization of severs resources.

6.2 Complexity of hot spot mitigation

For hot spot mitigation, let *ph* be the number of hot spots in the system during a decision. Sorting them based on their temperature takes O (*ph* _log (*ph*)). For each virtual machine is in hot spot, to sort the set of running VMs.

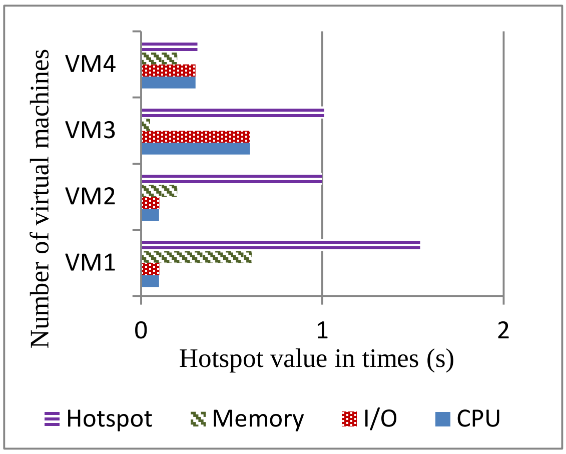

Figure 2: Hotspot value of VMs.

In real-time, the number of VMs that run on a PM is typically limited to a small constant. Hence, the sorting takes a constant amount of time. Under normal condition, the system is likely to have only a few hot spots within the decision time. The Figure 2 shows hotspot values of virtual machines wise VM1, VM2, VM3, and VM4. In x-axis, it takes number of virtual machines. In y-axis, it takes hot spot values of VM. From this analysis can identify the hot spot of VM easily and can eliminate quickly. To achieve 95% of elimination of hotspot while a decision. The graph clearly shows the hotspot value is gradually reduced while resource utilization of each virtual machine is low and the hotspot value is gradually increased while the resources utilization of each virtual machine is high. From the virtual machine VM2&VM4 is in normal mode, virtual machine VM3 is becoming normal mode and VM1 is in hotspot mode.

6.3 Complexity of Green Computing

For the green computing phase, let *pc* be the number of cold spots in the system during a decision run. Switch off them based on their memory sizes, I/O usage and CPU utilization takes O (*pc*_log(*pc*)). For each VM is in a cold spot, to switch off the corresponding VM. In real-time, the number of VMs that run on a PM is typically limited to a small constant. Hence, the switch off the VM takes a constant amount of time. The number of VMs on a PM is no more than a small constant.

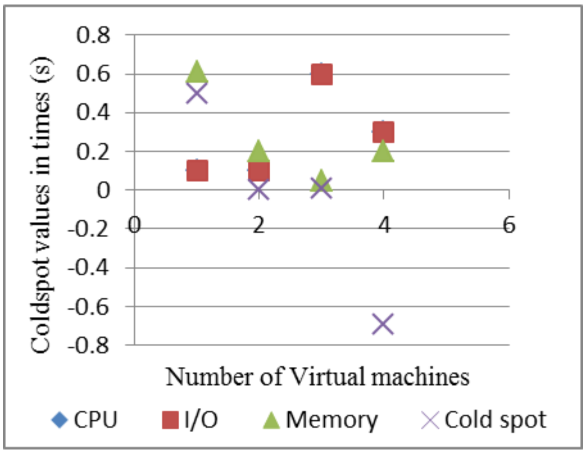

Figure 3: Cold spot value of VMs.

The Figure 3 shows cold spot values of virtual machines named wise 1, 2, 3, and 4. In x-axis, it takes number of virtual machines and in y-axis; it takes cold spot values of VM. From this analysis can identify the cold spot of VM easily and can switch off them quickly. To save 90% of power energy during consideration of cold spot value in a decision time. The graph clearly shows the cold spot value is gradually reduced while resource utilization of each virtual machine is low. From the virtual machine 2&4 is in switch off mode, virtual machine 3 is becoming switch off mode and 1are not in clod spot mode.

7 Conclusion

This paper presents an improved method for system in dynamic environment (Online gaming sever, video streamline server). VM Migration method eliminates 91% of resource overloaded; it helps to achieve higher utilization of server resource and achieves 87% of power energy conception. It can be improved up to 100% in future by using future trends approaches.

Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable commands and feedback. The authors acknowledge the Chairman, the Principal and the Director of the Adithya Institute of Technology for all help render in this research work.

Reference

[1]. Prasad Calyam, RohitPatali, Alex Berryman, Albert M. Lai, Rajiv Ramnath, “Utility-directed resource allocation in virtual desktop clouds”, *The International Journal of Computer and Telecommunications Networking, Vol. 55, No. 18, 29,Pages 4112 - 4130, Dec 2011.*

[2]. Vijay Sundaram, Timothy Wood, PrashantShenoy, “Efficient Data Migration in Self-managing Storage Systems”, *In Proceedings of the 2006 IEEE International Conference on Autonomic Computing* (ICAC *'06) PP. 297-300, June 2006.*

[3]. RosenbulmMenedl, Garfinkel Tal, “Virtual machine monitors: current technology and future trends,” *IEEE Computer, vol. 38, No. 5, PP. 39 - 47, May 2005.*

[4]. Timothy, PrashantShenoy,

ArunVenkataramani, and MazinYousif, “Black-box and Gray-box Strategies for Virtual Machine Migration”, *4th USENIX Symposium on Networked Systems Design & Implementation (NSDI ’07), PP. 229 - 242, Apr 2007.*

[5]. Michael R. Hines and KartikGopalan, “Post-copy based live virtual machine migration using adaptive pre-paging and dynamic self-ballooning,” *In Proceedings of (SIGPLAN/ SIGOPS) international conference on Virtual international conference on Virtual execution environments Vol. 43, No. 3, PP. 14 - 26, July 2009.*

[6]. Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauery, Ian Pratt, Andrew Wareld, “Xen and the Art of Virtualization, ”*In Proceedings of symposium on Operating systems principles (SOSP’03), Vol. 37, No. 5, PP. 164 - 177, Dec 2003.*

[7]. Mark Stillwell, David Schanzenbach, Frederic Vivien, Henri Casanova, “Resource allocation algorithms for virtualized service hosting platforms”, *Journal of Parallel and Distributed Computing, Vol.70, No. 9 , Pages 962 - 974, Sep 2010.*

[8]. Michael Nelson, Beng-Hong Lim, Greg Hutchins, “Fast transparent migration for virtual machines,” *In Proceedings of the annual conference on USENIX Annual Technical Conference (ATEC '05), PP. 391 - 394, Apr 2005.*

[9]. Violeta Medina, Juan Manuel García, “A survey of migration mechanisms of virtual machines,” *Journal of ACM Computing Surveys (CSUR) Vol. 46, No. 3, Article No. 30, Jan 2014.*

[10]. Wei-Tsong Lee, Tain- Wen Suen, Jhihsiang Huang, “An enhanced approach for estimating writable working set by pre-copy and prediction,” *In Proceedings of the 2013 International Conference on Information Networking (ICOIN) PP. 227 - 231, Jan 2013.*

[11]. Constantine P. Sapuntzakis, Ramesh Chandra, Ben Pfaff, Jim ChowMonica, S. LamMendel Rosenblum, “Optimizing the Migration of Virtual Computers,” *ACM SIGOPS Operating Systems Review - Operating systems design and implementation (OSDI '02) Vol. 36, No. 51, PP. 377 - 390, Winter 2002.*

[12]. DaniloArdagna, Marco Trubian, Li Zhang, “SLA based resource allocation policies in autonomic environments,” *Journal of Parallel and Distributed Computing, Vol. 67, No. 3, PP. 259 - 270, Mar 2007.*

[13]. David Meisner, Brian T. Gold, Thomas F. Wenisch, “PowerNap: Eliminating Server Idle Power,” *Architectural support for programming languages and operating systems, Vol. 44, No. 3, PP. 205 - 216, Mar 2009.*

[14]. Annie Cecile Orgeie, Marcos Dias De Assuncao, Laurent Lefevre, “A Survey on Techniques for Improving the Energy Efficiency of Large Scale Distributed Systems,” *Journal of ACM Computing Surveys (CSUR), Vol. 46, No. 4, Apr 2014.*

[15]. Jean-Marc Pierson, Georges Da Costa, Lars Dittmann, “Energy Efficiency in Large Scale Distributed Systems,” *COST IC0804 European Conference, EE-LSDS 2013, Apr 2013.*

[16]. RipalNathuji, Karsten Schwan, “VirtualPower: coordinated power management in virtualized enterprise systems”, *ACM SIGOPS symposium on Operating systems principles Vol. 41, No. 6, PP. 265 - 278, Dec 2007.*

[17]. Kejiang Ye, Xiaohong Jiang, Dawei Huang, Jianhai Chen, Bei Wang, “Live Migration of Multiple Virtual Machines with Resource Reservation in Cloud Computing Environments,” *CLOUD '11 Proceedings of the 2011 IEEE 4th International Conference on Cloud Computing, pp. 267 - 274, Sep 2011.*

[18]. Mayank Mishra, Anwesha Das, Purushottam Kulkarni, and Anirudha Sahoo, “Dynamic resource management using virtual machine migrations,” *IEEE Communications Magazine, Vol. 50, PP. 34 - 40, Sep 2012.*

[19]. Zhang X, Huo Z, Ma J, Meng D, “Exploiting Data Dependancy to Accelerate Live Virtual Machine Migration,” *In IEEE International Conference on Cluster Computing, PP. 88–96, Sep 2010.*

[20]. UmeshDeshpande, Xiaoshuangang, KartikGo palan, “Live gang migration of virtual machines”, *InProceedings of the 20th international symposium on High performance distributed computing (HPDC '11), PP. 135 - 146, June 2011.*

[21]. Zhen Xiao, Weijia Song, and Qi Chen, “Dynamic Resource Allocation Using Virtual Machines for Cloud Computing Environment,” *In IEEE Transactions On Parallel And Distributed Systems, Vol. 24, No. 6, PP. 1107- 1117, June 2013.*