VM placement Strategies in Cloud Computing based upon EA

[1]Amrit Kaur, [2]Bhupinder Singh

[1,2]*College Name*

[1]amritkaursidhu99@gmail.com, [2]Guide Email id

Abstract*— This implementation aims towards the establishment of performance qualitative analysis on load sharing in VM to VM and then implemented in CloudSim with Java language. Here major stress is given on the study of load balancing algorithm with heterogeneous resources of the cloud, followed by comparative survey of other algorithms in cloud computing with respect to scalability, homogeneity or heterogeneity and process migration. A previous study also indicates change of MIPS will affect the response time and increase in MIPS versus VM decreases the response time. When image size of VM is implemented against the VM bandwidth then no significant effect is found on response time and it remains constant for which these parameters are investigated. But in case of Cloudlet long length versus Host bandwidth a pattern is observed in which response time increases in proportionate manner.

Keywords: Cloud, VM, Host, VM Placement Schemes

I. INTRODUCTION

With the development of high speed networks, there is an alarming rise in its usage comprised of Web queries a day and thousands of e-commerce transactions. A large scale data centers handle this ever increasing demand by consolidating hundreds and thousands of servers with other infrastructure such as cooling, network systems and storage. The development of this commercialization is named as cloud computing. Clouds are sky rocketing virtualized data centers and applications offered as services on a subscription basis. The characteristics exhibited by Clouds are shown in Fig 1

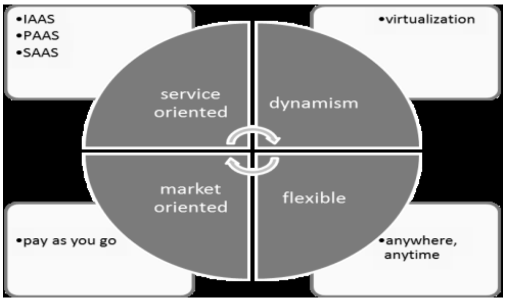

Fig 1: Characteristics of cloud computing

Cloud delivers three type of services such as Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) [2] and can be deployed in three different ways *i.e.* private, public and protected. SaaS mainly deliver the online software application to the client, whereas PaaS gives the capability to create application services as on their desire. It allows users to develop their software using programming languages and tools supported by the provider. Infrastructure as a Service (IaaS) provides the capability to have control over complete cloud infrastructure with CPU processing, storage, networks, and other computing resources.

Cloud is a business model, which provides the on demand computing resources as a service to the clients on the rent basis. Client need to pay only for that amount that is actually used. To generating high revenue, cloud provider use virtualization technology. Virtualization [5-6] is the key technology in cloud computing, which divide the physical resources and allow the sharing of these resources. With the help of virtualization number of user can share the same resources without intervening to each other. Hypervisor is a small process also known as virtual machine monitor (VMM) is used to deploy the virtualization. It behaves like an operating system and responsible for taking all the decision related to the VM. When the user demand for the computer resources (CPU, storages, network), hypervisor create the VM and assign to the user. Number of VM can be created in the single physical machine (PM). In cloud each data center keep number of host. When a request for the VM comes to the hypervisor, where this VM is to be placed is known as VM placement problem. VM placement is a NP hard problem [8]. Therefore finding a suitable host for placing VM is a very challenging task. VM placement are required in two different situation either for placing new VM or to place a migrated VM. Transferring the VM from one host to another host is called VM migration [8]. VM migration is needed to deal with several situations such as server consolidation, load balancing, maintenances, server failure, hot spot mitigation etc. Migrations degrade the system performance, so number of migration should be minimized as possible. VM is the main processing unit in the cloud. User's applications run on the VM and the resource requirement of these applications changes dynamically. Resources in the cloud are multidimensional (CPU, memory, bandwidth etc.). So resource required by the VM may be different in their amount and types.

II. VM PLACEMENT MODEL

In cloud environment each data center can have N heterogeneous physical nodes. Each physical node is characterized by the CPU performance measure in MIPS, RAM size and network bandwidth. Local and global manager are used to place VM. Global manager which is also known as virtual machine manager reside on the master node and collect information from the local manager for the high availability. If local manager fail then all required information can be obtained from the global manager. It divides all available host of the data center into the six clusters according to the remaining capacity of the resources. When global manager receive resource request from the user it send it to the appropriate cluster according to the resource requirement of the VM. Each cluster has its own local manager, which continuously received information from all available hosts in the data center. Based on this information local manager decide where the selected VM is placed. Load balancing and consolidation policy are resided on the local manager. If local manager fail then global manager select lightest host in the cluster, migrate all VM from that host and make it local manager for that cluster.

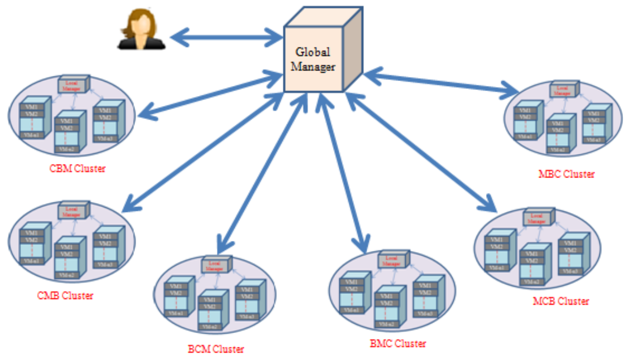

Fig 2: VM Placement Model

III. Classification of VM Placement Algorithms

Goal of the VM placement can either saving energy by shutting down some severs or it can be maximizing the resources utilization. Based on these goal placement algorithm are classified into two type.

I. Power Based approach
II. Application QoS based approach

Main aim of the power based approach is to save the energy. In these approach VM map to the physical machines in such a `way, that each servers can utilized their maximum efficiency and the other servers can be shut down depending on load conditions. While in the Application QoS based approach a VM map to the PM with the aim of maximizing the QoS delivered by the service provider.

IV. Smart Placement Approach

SPA is an approach that takes into account the availability of the whole network resource, while guaranteeing load-balancing and SLAs objectives. To achieve this, migrating Virtual DataCenter Networks (VDNs) should clearly specify its detailed resource requirements (i.e. the resource vector) to the hosting physical network. This can provide for optimal placements and satisfying services. In this context, requirements may vary from a virtual network to another, depending on the considered topologies and the provided services. However, among all the network components, the challenge for the hosting CDNs (i.e. the physical ones) mainly lies in the switching capabilities of its network, more precisely, its path processing capacities. Indeed, where for a packet to get processed through a switching device, certain resources are required. In this context, let us define the physical switch as a set of virtual switches, where each virtual switch operates a set of virtual switching paths. Mainly, a virtual switching path to operate requires a set of: (1) packet processing resources (network processor cycles, search caches, memories); (2) ports; (3) bandwidth over the ports. Typically, for a packet processing task to operate, this requires: (1) processors (for parsing and analysis); (2) memories (for the lookup tables) that can be either internal or external (e.g. TCAMs, SRAMs); (3) queues (for packets' scheduling and storage, and for the process of shaping priorities); (4) bandwidth over the busses that interconnect the aforementioned internal components. Accordingly, such physical resources will be virtually partitioned among the different virtual data paths that are allocated (reserved) to satisfy the requirements of the VDNs topologies. Hence, for efficient allocations and optimal placement decisions, such resource vectors need to be clear in order to check for resource availability at the hosting physical network.

Therefore, to simplify the presentation, the resource requirements of the migrating networks (i.e. the VDNs) will be represented by the virtual data path capacity. At the hosting side (i.e. the hosting CDNs topology), this will be translated to path processing resources, being a parameter for the placement process. Thus, having the resource victor of the migrating VDN, the hosting network administrator can break this down to: (1) ports; (2) processing engine capabilities; (3) memories; (4) internal bandwidth; all constrained by certain speed/delay limits for QoS assurance. Besides, the administrator need to specify the external bandwidth requirements (bandwidth over the links that interconnects the different switching devices), this could be defined by the Network Interface Cards (NICs) capacities.

V. Problem Formulation

Interesting models that tackle the problem of virtual machines placement are already proposed in the literature. However, to the best of our knowledge, no work tackles the problem of placing a full network as one package. In this context, placement models should provide for performance optimality for the whole parties involved in the placement process. Indeed, where is not a wise solution to place a virtual machine in a way that fulfills its performance requirements but causing problems to others. Considering the placement costs is also a crucial factor in such decision, however, other factors like load-balancing and QoS guarantees are also important.

- First, the candidate PMs are chosen in a way that does not consider that fact that reaching those PMs is done through a network that consists of several nodes (i.e. the connecting devices like switches and routers) interconnected via set of links (i.e. bandwidth resources). Checking the resource availability at the servers only is not enough to provide a proper candidate to hold the required applications or the needed services. Typically, this model will result in a non-optimal placement decision. Indeed, where being part of a network, it is some times more crucial to check the resource availability over the network components (e.g. switches and links) that interconnects such servers with the end-users. Without these resources, servers capacity is useless!

- Second, the migration-based placement scenario allows migrating the already placed VMs from one PM to another in a way to make space for a new VM to be place instead. This is not efficent, as those VMs that already been placed and run over a PM may have functional dependencies between each other or with other VMs from different PMs5. Migrating (re-placing) such VMs to new PMs may not be a feasible process, as this may cause several problems (e.g. service failures, interruptions) for many VMs which may violate the Service Level Agreements (SLAs) and the contracted QoS guarantees.

Third, the proposed model does not consider the load-balancing issues. Choosing the candidate PMs according to the total completion time does not necessarily comply with the load-balancing objectives. Administrators of Cloud-service networks always aim to maximize their revenue objectives while not violating the SLAs and the QoS guarantees provided to their customers. To do so, they tend to keep the loads over their networks balanced in a way to avoid congestions or any other problems that may harm the network performance.

VI. Flow Chart

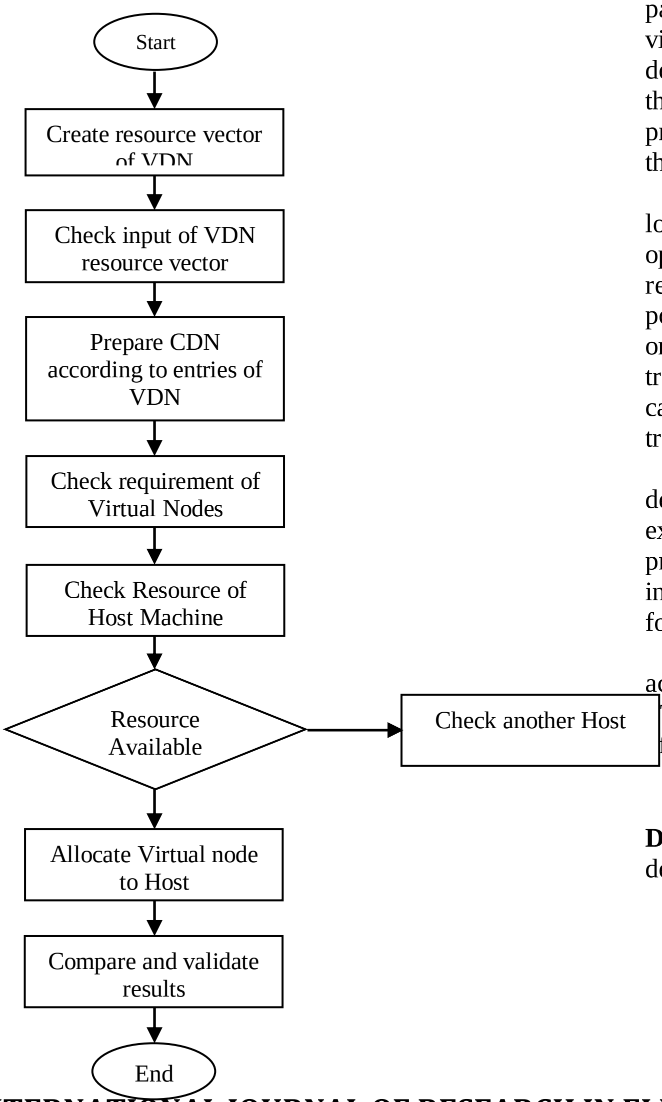

Fig 3: Flow Chart

STEP 1: Virtual machines in idle state. It describes about the main screen that is showing the machines that are in the idle state that is no load is assigned to them. In the proposed scheme the load migrated is done on the basis of parameters like utilization, speed, memory and power where VM is the virtual machine.

STEP 2: Load the machines. It describes about the machines when load is assigned to all the machines. The assigned values described about the load on various machines. When the load is allocated to the various machines continuously and it reaches the threshold value, the load will be migrated from that loaded machine to the other underloaded machine.

STEP 3: Overloaded Machine. It describes about the overloaded machine that is in the red mark. The assumed threshold for the overload condition to occur is above 80%.When the threshold is crossed, the load in the machine is migrated to the optimal destination having less load on it.

STEP 4: According to the priority with respect to the parameters like utilization, memory, speed and power of the virtual machines, the optimal destination is chosen. It describes about the selection of the candidate Virtual Machine that to which the load is to be transferred according to the priority table. In the research a priority table is developed by the algorithm, for the calculation of the destination machine.

STEP 5: Selection of the Virtual Machine that to which the load is to be transferred according to the priority table. The optimal destination is chosen according to the priority with respect to the parameters like utilization, memory, speed and power of the virtual machines. The machine having less load on it and greater speed and better power, the load will be transferred to it. It describes about the selection of the candidate Virtual Machine that to which the load is to be transferred according to the priority table.

STEP 6: Downtime during the load sharing. Downtime is defined as the time at which the virtual machines stop executing. It includes transfer of the processor state. In the proposed approach, the downtime is decreased which results in better performance. The downtime can be calculated by the formula:

Total Downtime = Stop-and-copy + commitment + activation.

STEP 7: Efficiency. In the proposed approach, the efficiency of the system is improved.

VII. Results and Discussion

Down Time: In the proposed approach, the downtime is decreased which results in better performance.

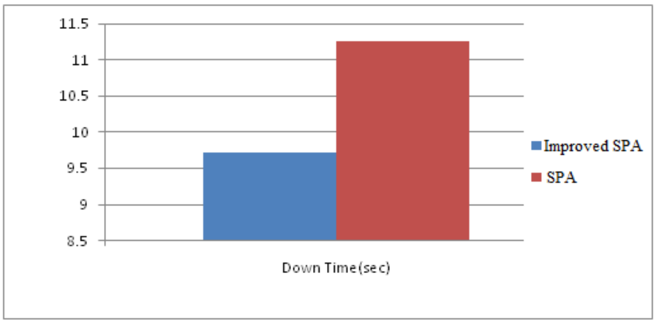

Fig 4: Down time

Efficiency: In the proposed approach, the efficiency of the system is improved.

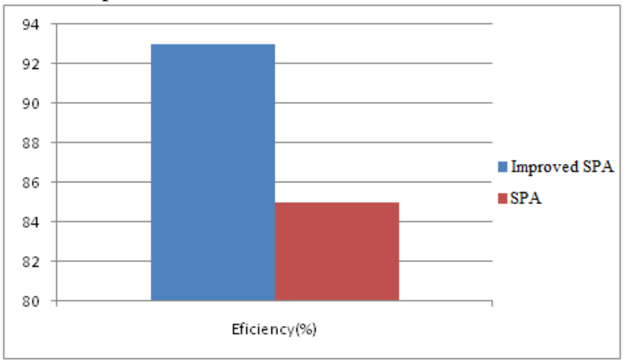

Fig 5 Efficiency

Comparison Table of work performed:

Here the comparison takes place between the base paper and the work performed. The results produced by the work are better than the previous work done.

Table 1: Comparison table of work performed

Approach	Down Time(sec)	Efficiency(%)
SPA	11.3	85
Improved SPA	9.7	93

Downtime is defined as the time at which the virtual machines stop executing. It includes transfer of the processor state. In the proposed approach, the downtime is decreased which results in better performance.

VIII. CONCLUSION

This implementation aims towards the establishment of performance qualitative analysis on load sharing in VM to VM and then implemented in CloudSim with Java language. Here major stress is given on the study of load balancing algorithm with heterogeneous resources of the cloud, followed by comparative survey of other algorithms in cloud computing with respect to scalability, homogeneity or heterogeneity and process migration. A previous study also indicates change of MIPS will affect the response time and increase in MIPS versus VM decreases the response time. When image size of VM is implemented against the VM bandwidth then no significant effect is found on response time and it remains constant for which these parameters are investigated. But in case of Cloudlet long length versus Host bandwidth a pattern is observed in which response time increases in proportionate manner. Using the modified approach the reduction in the down time of the various processes are achieved as shown in results.

IX. REFERENCES

[1]. Ahmad Nahar Quttoum, Mohannad Tomar, Bayan Khawaldeh, Rana Refai, Alaa Halawani, Ahmad Freej, "SPA: Smart Placement Approach for Cloud-service Datacenter Networks", The 10th International Conference on Future Networks and Communications, Vol: 56, 2015, pp: 341-348

[2]. Chonglin Gu, Pengzhou Shi, Shuai Shi, Hejiao Huang, Xiaohua Jia, "A Tree Regression-Based Approach for VM Power Metering", Special Section On Big Data For Green Communications And Computing, IEEE, June 1, 2015

[3]. Zhaoning Zhang, Ziyang Li, Kui Wu, Dongsheng Li, Huiba Li, Yuxing Peng, Xicheng Lu, "VMThunder: Fast Provisioning of Large-Scale Virtual Machine Clusters", IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, Vol. 25, No. 12, DECEMBER 2014

[4]. Jiaxin Li, Dongsheng Li, Yuming Ye, Xicheng Lu, "Efficient Multi-Tenant Virtual Machine Allocation in Cloud Data Centers", TSINGHUA SCIENCE AND TECHNOLOGY, ISSN: 1007-0214, Volume 20, Number 1, February 2015, pp: 81-89

[5]. Gursharan Singh, Sunny Behal, Monal Taneja, "Advanced Memory Reusing Mechanism for Virtual Machines in Cloud Computing", 3rd International Conference on Recent Trends in Computing, Vol: 57, 2015, pp: 91-103

[6]. Narander Kumar, Swati Saxena, "Migration Performance of Cloud Applications- A Quantitative Analysis", International Conference on Advanced Computing Technologies and Applications, Vol. 45, 2015, pp: 823-831

[7]. Aarti Singh, Dimple Juneja, Manisha Malhotra, "Autonomous Agent Based Load Balancing Algorithm in Cloud Computing", International Conference on Advanced Computing Technologies and Applications, Vol: 45, 2015, pp: 832-841

[8]. S. Sohrabi, I. Moser, "The Effects of Hotspot Detection and Virtual Machine Migration Policies on Energy Consumption and Service Levels in the Cloud", ICCS, Vol: 51, 2015, pp: 2794-2798

[9]. Mohammad Mehedi Hassan, Atif Alamri, "Virtual Machine Resource Allocation for Multimedia Cloud: A Nash Bargaining Approach", International Symposium on Emerging Inter-networks, Communication and Mobility, Vol: 34, 2015, pp: 571-576

[10]. Christina Terese Josepha, Chandrasekaran K, Robin Cyriaca, "A Novel Family Genetic Approach for Virtual Machine Allocation", International Conference on Information and Communication Technologies, Vol: 46, 2015

[11]. Antony Thomas, Krishnalal G, Jagathy Raj V P, "Credit Based Scheduling Algorithm in Cloud Computing Environment", International Conference on Information and Communication Technologies, Vol: 46, 2015, pp: 913-920

[12]. Doshi Chintan Ketankumar, Gaurav Verma, K. Chandrasekaran, "A Green Mechanism Design Approach to Automate Resource Procurement in Cloud", Eleventh International Multi-Conference on Information Processing, Vol: 54, 2015, pp: 108-117

[13]. A.I.Awad, N.A.El-Hefnawy, H.M.Abdel_kader, "Enhanced Particle Swarm Optimization For Task Scheduling In Cloud Computing Environments", International Conference on Communication, Management and Information Technology, Vol: 65, 2015, pp: 920-929

[14]. Arunkumar. G, Neelanarayanan Venkataraman, "A Novel Approach to Address Interoperability Concern in Cloud Computing", 2nd International Symposium on Big Data and Cloud Computing, Vol: 50, 2015, pp: 554-559

[15]. Atul Vikas Lakra, Dharmendra Kumar Yadav, "Multi-Objective Tasks Scheduling Algorithm for Cloud Computing Throughput Optimization", International Conference on Intelligent Computing, Communication & Convergence, Vol: 48, 2015, pp: 107-113

[16]. Narander Kumar, Swati Saxena, "A Preference-based Resource Allocation In Cloud Computing Systems", 3rd International Conference on Recent Trends in Computing, Vol: 57, 2015, pp: 104-111

[17]. Nidhi Bansal, Amitab Maurya, Tarun Kumar, Manzeet Singh, Shruti Bansal, "Cost performance of QoS Driven task scheduling in cloud computing", Third International Conference on Recent Trends in Computing, Vol: 57, 2015, pp: 126-130

[18]. Mehmet Sahinoglua, Sharmila Ashokan, Preethi Vasudev, "Cost-Efficient Risk Management with Reserve Repair Crew Planning in CLOUD Computing", Cost-Efficient Risk Management with Reserve Repair Crew Planning in CLOUD Computing, Vol: 62, 2015, pp: 335-342

[19]. Patricia Arroba, Jos´e L. Risco-Mart´ın, Marina Zapater, Jos´e M. Moya, Jos´e L. Ayala, Katzalin Olcoz, "Server Power Modeling for Run-time Energy Optimization of Cloud Computing Facilities", 6th International Conference on Sustainability in Energy and Buildings, Vol: 62, pp: 2014, pp: 401-410