Proceedings of the Fifth International Conference on Communication and Electronics Systems (ICCES 2020)
IEEE Conference Record # 48766; IEEE Xplore ISBN: 978-1-7281-5371-1

Optimizing Resource Utilization by Vm Migration Among Virtual Machines of a Cloud Server

Nandan Adeshara
School of Computer Engineering and Technology
MIT Academy of Engineering
Pune, India
nandanadeshara23@gmail.com

Ajinkya Rede
School of Computer Engineering and Technology
MIT Academy of Engineering
Pune, India
redeajinkya@gmail.com

Suhani Jain
School of Computer Engineering and Technology
MIT Academy of Engineering
Pune, India
suhanijain456@gmail.com

Krishna Dhoot
School of Computer Engineering and Technology
MIT Academy of Engineering
Pune, India
dhootkrishna18@gmail.com

Sunil Mhamane
Assistant Professor, School of Computer Engineering and Technology
MIT Academy of Engineering
Pune, India
sunil.mhamane015@gmail.com

***Abstract*- In a cloud data center server, there could be many physical machines connected as clients. Each physical machine can hold multiple virtual machines. The effective working of the system is determined majorly on the parameters such as the utilization of the system which depends upon resource utilization by single Virtual Machine and Physical Machine resource utilization in total. To optimize the utilization, it is crucial to fetch the data, take the decision considering the situation, and finally take action by migrating the VM to decide the PM if needed.**
To automate this process and to make such decisions, it is necessary to appoint a monitor machine which can monitor the VM and PM individually, take and implement the actions needed. However, the monitor machine must implement multi-objective optimization thus considering other factors as well such as Qos, Energy saving, and reduction of time and distance while migrating.

***Keywords* - Optimization, VM migration, cloud server, monitor machine, cloud computing, FTP server.**

I. INTRODUCTION

To achieve the proficiency of a cloud server it is crucial to understand, monitor, and manage different utilizations such as compute resources, memory, etc.[1] There are different courses of action by which it can be carried out but to choose between them one needs to be aligned with one s goal. To pull off the aim of the project, there is a need to access all the relevant parameters further using them to the benefit of taking a precise and acceptable decision to finally implement the decision. In this case, the goal of VM migration and VM farming on the targeted location is to be carried out considering the multi-objective optimization.[2]

This will contribute in multiple constructive ways such as energy-saving, resources optimal usage, saving manpower and increasing accessibility and usability of the technology

II. LITERATURE SURVEY

Generally, depending upon the available free resources, there are certain conditions of virtual machines in cloud computing, overloaded, and underloaded[3]. When any of the hosts or server in the cloud network face one of these conditions, the most appropriate virtual machine is selected on some VM Selection policies and then is transferred or migrated to some other Hosts or Server and the Virtual Machine is transferred or migrated based on some VM Migration policies:

A. Candidate Virtual Machine Selection Policy:

This step gives us a final virtual machine to be transferred as an output. This is important as an appropriate VM should be transferred otherwise any bad decision can cause wastage of resources, time, and energy.

T.Wood et al.[4] proposed a method of using a Volume to Size Ratio(VSR). Virtual Machines with the highest VSR value are selected to be transferred to another host or server, where size is the total available memory of virtual memory, and volume is directly corresponding to the load on virtual machine.

To solve the issue of load balancing, Beloglazov et al.[5] proposed a method in which a threshold value is calculated which is the upper required threshold of the host, and then a virtual machine with size greater than this threshold is selected. If there is no such virtual machine then multiple machines are to be selected.

The Minimum Migration Time (MMT) Policy [6] proposed by Beloglazov and Buyya states that the Virtual Machine with the smallest value of migration time from source host to the destination host and also with minimum memory utilization is selected to be migrated.

B. Virtual Machine Migration Policy:

This step includes the actual migration of the Virtual Machine with its memory and CPU state from the current Host to the destination Host. Now, this can be executed in many ways using many policies or methods which can be distinguished as Live Migration [7]and Cold Migration[8][9]:

Stop and Copy: In the above method, it has the need to stop the virtual machine and copy the memory pages and CPU statistics to the destined physical machine, and after successfully transferring, run the VM on the new host. This method falls under the Cold Migration category. This method gives the desired best total migration time but at the same time gives a gradual increase in the downtime of the system.[7]

Post Copy Migration [10]: In the above method, the primary focus is on minimizing the data to be transferred. This can be implemented by minimizing the number of memory pages and CPU states which are sufficient to run the virtual machine on the host machine. In this process also the VM can be stopped on the current host and then copy the stats. This method falls under the Live Migration.[8] [9]

This method reduces the downtime to the fullest but increases the migration time.

Pre Copy Migration[11]: In the above method, the running virtual machine remains non-stop on the source host, a copy of system memory pages can be made and the CPU states and transfer it to the new receiving host, now during transfer, some new memory pages may be created or modified, these memory pages are called as Dirty Pages. So these sets of Dirty Pages are transferred in the second round, and this process goes on until the size of these sets of dirty pages minimizes to some predefined allowed size say 2 MB. When these conditions are met, it stops the VM running on the source host and initiate the VM on the destination host.

This method is considered only if the rate of dirty page transfer is greater than the rate of growth of the dirty page. This method also falls under Live Migration.[7][8]

Pre-Copy Migration method is the most reliable method for migration of VM, it reduces the downtime of the system as well as reduces the migration time. This also keeps records on the source host so that if there is an issue running VM on the destination host, the data can be retrieved from the source host.

III. RELATED SOFTWARES AND TECHNIQUES FOR MIGRATION -

A. VMWARE WORKSTATION PRO:

Offered by VMWare, it is a hypervisor on which one can use and configure multiple VM s (Virtual Machines). The winning service of this software is that one can use all the different VM s along with the underlying Physical Machine. There are two different options to migrate the VM either to the same host or to a different host depending upon the need for migration. However, a major setback of this system of optimization is that the user needs to operate manually. To do so it is important to sync the path of the directory along with the path of the config file of VM. The user needs to manually cut the file containing all the VM files and paste it to the desired location on the host (same or different). After successfully doing so, if the VM functions properly, the original file needs to be deleted from the initial location. This whole process is very tedious, inefficient, and prone to errors and can even lead to loss of data or malfunctioning of the VM.

B. COLD MIGRATION:

It means migration of a suspended or in other words powered off a virtual machine from one PM to another to achieve different goals. To carry out this process it is crucial for the VM to not be working else the process will get suspended. Also, the location to which one wishes to migrate must be ready to accommodate. Cold migration can happen between data centers, different hosts, or instances of vCenter Server.

C. USING VSPHERE SUITE:

To make use of the vSphere suite it is essential to have it s components onboard. For small environments, vSphere Client can directly access the ESXi hypervisor[12] present in each PM while for large environments, it is required to download VCenter Server in the VMware as a VM to allow centrally access and manage all the ESXi present in the data center/ environment through vSphere Client. The VM to be migrated can be done on a VM which is powered off or suspended or even in working state using VMotion.

D. VIRTUALBOX MIGRATION:

To migrate one VM from PM1 toPM2 within a VirtualBox, there are three ways :

Using the Import or Export function provided - Using this it throws all the images and if the process is done incorrectly can cause serious loss of data.

Coping the whole folder containing all the .vdi and.vbox files - This is a safe and easy method to migrate however while copying keep in mind to put all the VDIfiles in the correct directory.
Cloning and recreating on the target site.

IV. PROPOSED ARCHITECTURE

The system architecture as shown in Fig.(1) is proposed such that the global agent or the host server can monitor all the PMs by sending commands to the Hypervisor or the VMM of the Virtual Machine to collect the Virtual Machine data consisting of RAM usage, storage, number of processors running, CPU utilization, Computational power, bandwidth, etc according to the need.

Fig. (1): Proposed Architecture

The Hypervisor then requests the local agent which is the OS of the Physical Machine to collect, compile and send the data to the host via the network. This is done. This is done because a single physical machine may contain multiple VM s hence making it crucial to collect all the data in one place. This data is then sent to a Prediction module, located inside the Global Agent. This module suggests which VMs should be migrated, so that system will work most efficiently. The suggestions by Prediction module are carried out by the Global Agent. These instructions are then passed to the local agents and the migration is carried out.

V. PROPOSED SYSTEM AND ALGORITHM

Ant Colony Optimization is a powerful technique to find the length of the shortest path in the most efficient and effective manner. It has 2 scenarios: Ideal and Non-Ideal.

One of the applications of the Ant Colony Optimization theory can be found in the Cloud domain where resource allocation and management plays an important role. Now you ask how is optimization related to the shortest path? To answer this question, it is known that the Cloud servers on Datacenters host many resources, eg. AWS hosts a plethora of services and keeps its servers free by locating and relocating unwanted applications to other servers. Optimization helps the servers in responding to incoming requests faster and reducing the load on the centers which saves them millions annually and also helps in achieving efficient energy utilization. So back to how ACO is connected to the current setting? ACO algorithm works in the following way -
Step-1: Select N number of ants and initialize the matrix of pheromone deposits, it is the same shape as the distance matrix. And coordinates respond to the same paths.

Step-2: If distances[2,5] = 35 the distance from 2 to 5 is 35, and if pheromone[2,5] = 0.8 the level of pheromone deposited on the path between 2 and 5 is 0.8. The pheromone matrix is initialized with small variables all of the same value.

Step-3: Exploring paths:
Ant decides on what places to go to use this:
path_to_path_score = pheromone ** alpha * (1.0 / distance) ** beta, where alpha and beta act as weights on pheromone and distance respectively.

Calculate path_to_path_score for all the available paths (ignoring paths which are already visited). The probability of going to the next path is:
prob_of_going_to_path(i) = path_to_path_score(i) / sum_of_all_available_path_to_path_scores

For example, if an ant is at path 2, and available paths are 4,7,8. The scores are computed for those paths as:
{4: 0.2, 7: 0.4, 8: 0.8} the probability of going to 4 is 0.2 / (0.2 + 0.4 + 0.8) = 0.142857 and so on. An ant keeps going from path to path according to the above choosing rule until it visits all paths. If 20 ants start, they will have 20 paths at the end of this group of ants traveling generation.

Since in the initial step the pheromone levels are the same, the choices are made on distances + some noise. Keeping track of the successful routes is when ants deposit pheromone. On the way back all ants or selected numbers of the best ants deposit pheromone on the paths they traveled.

Step-4: They deposit:
1 / (distance between two paths)
For example: an ant traveled a path: [(0 -> 3) (distance: 8), (3 -> 5) (distance: 2)] 0.125 units of pheromone would be deposited on pheromone[0,3] += 0.125 and pheromone[3,5] += 0.5 This is done to encourage ants to give more priority to shorter routes between paths.

The final piece have to let pheromone decay (considering Ideal case), so old pheromone does not confuse the next generations of ants. Multiplying the pheromone matrix by decay rate. Right after they deposit. Therefore pheromone that has been sitting for a while has been subject to many decays and should be small. And repeating these steps for required iterations.

The following algorithm is proposed considering a small data center consisting of a single server to manage multiple clients all connected in a single network either wired or non-wired. The proposed algorithm is based on client-server

architecture having a server working as a catalog as well as a communication server. The server acts as a global agent that contains in itself the client information and is an art of the monitor machine as shown in Fig. (2)

No.of PCs	No. of Servers	No. of Data Centres	No. of Networks	No. of Jobs (VMs)
3	1	1	1	3

Table (1): List of Components

The first step towards implementing the algorithm is making sure the client s machines are connected to the monitor machine sharing the same network. The Concept of VMM can be introduced here as it covers the fundamentals of monitoring and validating the successful migration of resources over the network. When the resources of the client exceed a specific threshold, this will be reflected in data sent to the server, periodically, by the client. Once the data is onboard the monitor machine uses them as input to the decision making algorithm. The decision is taken considering various parameters such as CPU utilization, the shortest path of migration (calculated using Ant - Colony Optimization[13]). The Parameters have got their priorities which tells the decision code to decide to take into account the priorities of the parameters while comparing them.

Once the decision is taken, it is circulated to the two physical machines which will be involved in the migration. The migration is a tedious process as it involves deciding the path and other parameters to make sure the process is beneficial in the end. Migration also ensures that the virtual machine is deleted from the initial machine completely and is not holding any resource of the initial location as it should also be transferred completely in the same state as it was in before migration and must function without any errors.

As discussed earlier, both machines are in the same network, hence the user will be able to access the VM situated in another PM after migration from the previous host location using the VMRUN commands provided by the VM Workstation Pro. This algorithm is hence, dynamic and saves time and energy.

Python Language was used to execute the process starting from client data fetch to VM migration and optimizing the PMs. Different libraries and modules were taken into account for successful task completion. Following list of modules were used to build an actual working model and the modules are:

-Python Standard Lib [includes all the necessary libs]

-ftplib [to create a network for a client s entry point]

-tkinter [GUI build above the server architecture]

-VMRUN commands

RESULTS

As explained in the proposed system and architecture section, building a model that takes the client-side resources(in this case CPU parameters, other consolidation parameters) and performing necessary action to consolidate the required client by fetching it in the runtime and bringing the machine in an efficient state of use for end-users.

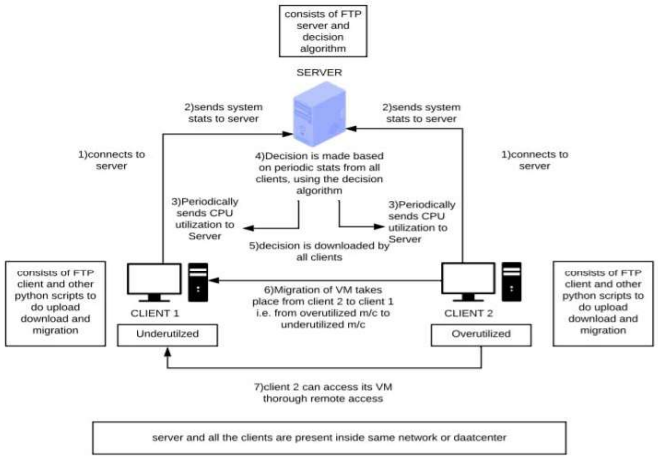

Fig.(2). Design of The Experiment

Hence, from Fig.(2), the VM on overutilized machine is migrated to the underutilized machine. The migrated VM is accessed remotely by its former host. Thus energy efficiency of the system increases, at the same time, resources utilization is smartly managed while maintaining the overall performance.

Here in Table (2). 'S' stands for Sender, 'R' for Receiver, and 'I' for Idle.

PCNO.	PC IP	CPU Usage (%)	Threshold (%)	Transfer To	Status	Migration Time (in seconds)
1	127.168.0.1	20	60		R	
2	127.168.0.2	70	60	1	S	300 approx.
3	127.168.0.3	30	60		I	

Table (2): Results Table

CONCLUSION

The paper purposefully demonstrates the strategy to develop a dynamic, reliable, time-saving program which will help to save and utilizing the resources in a better way by migrating the Virtual machines from burdened PM to another potential PM without the loss of state and data thus saving a considerable amount of energy and resource utilization.

Fig. (3): Proposed System

FUTURE WORK

The designed system, Fig.(3), has three basic components for the sake of experiment, one server and two clients. The server has GUI interface, through which information about the connected clients can be accessed. The GUI also provides a start and stop button for the server. The server has a python script, which upon receiving usage statistics from the connected clients, decides whether to migrate any VM from one machine to another. The clients are the physical machines or PC on which a VM is running. Each client runs its code which connects it with the server and enables it to communicate with the same.

The future scope of this method is vast that may include migration among various servers or even among different data centers. Also hosting multiple VMs on the same PM can be taken into consideration.

VI. REFERENCES

[1] A. Beloglazov and R. Buyya, "Energy-Efficient Resource Management in Virtualized Cloud Data Centers," 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, Melbourne, VIC, 2010, pp. 826-831.

[2] A. Beloglazov and R. Buyya, "Energy Efficient Allocation of Virtual Machines in Cloud Data Centers," 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, Melbourne, VIC, 2010, pp. 577-578.

[3] D. Deng, K. He and Y. Chen, "Dynamic virtual machine consolidation for improving energy efficiency in cloud data centers," 2016 4th International Conference on Cloud Computing and Intelligence Systems (CCIS), Beijing, 2016, pp. 366-370.

[4] T. Wood et al., Black-Box and Gray-Box strategies for virtual machine migration , proc. 4th conference on symposium on network system design and implementation, 2007.

[5] A. Beloglazov et al. Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing , Elsevier journal of Future Generation Computer Systems, pp. 755 768, 2012.

[6] A. Beloglazov and R. Buyya, Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in Cloud data centers, Concurrency Comput., Pract. Exper., vol. 24, no. 13, pp. 1397 1420, 2012.

[7] C. Clark et al., Live migration of virtual machines, in Proc. 2nd Symp. Netw. Syst. Design Implement. (NSDI), Boston, MA, USA, 2005, pp. 273 286.

[8] D.Perez Botero, A brief tutorial on live migration of virtual machine from a security perspective , 2011.

[9] H. Jin et al., Live virtual machine migration with adaptive memory compression proceeding of the IEEE international conference on cluster computing, pp. 1-10, 2012.

[10] F. Zhang, G. Liu, X. Fu and R. Yahyapour, "A Survey on Virtual Machine Migration: Challenges, Techniques, and Open Issues," in IEEE Communications Surveys & Tutorials, vol. 20, no. 2, pp. 1206-1243, Second Quarter 2018.
[11] Yi Zhong, Jian Xu, Qianmu Li, Hong Zhang and Fengyu Liu, "Memory state transfer optimization for pre-copy based live VM migration," 2014 IEEE Workshop on Advanced Research and Technology in Industry Applications (WARTIA), Ottawa, ON, 2014, pp. 290-293.
[12] D. T. Vojnak, B. S. Đorđević, V. V. Timčenko and S. M. trbac, "Performance Comparison of the type-2 hypervisor VirtualBox and VMWare Workstation," 2019 27th Telecommunications Forum (TELFOR), Belgrade, Serbia, 2019, pp. 1-4.
[13] S. Jain, K. Dhoot, A. Rede, N. Adeshara and S.Mhamane, "Optimization of resources in Cloud Computing using Virtual Machine Consolidation," 2019 International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 2019, pp. 1285-1288.
[14] G. Sun, D. Liao, D. Zhao, Z. Xu and H. Yu, "Live Migration for Multiple Correlated Virtual Machines in Cloud-Based Data Centers," in IEEE Transactions on Services Computing, vol. 11, no. 2, pp. 279-291, 1 March-April 2018.