Survey on VM Placement Algorithms

B. Benita Jacinth Suseela

Post Graduate Student, Department of Computer Science and Engineering,
Karunya University, Coimbatore, Tamil Nadu, India.

***Abstract*— In cloud computing, there are many strategies used for virtual machine (VM) placement. Objectives for VM placement are to reduce the number of physical machines required, VM allocation time and to reduce resource and power wastage. This paper surveys various VM placement algorithms for reducing the required number of physical machines or for efficient usage of resource consumption and power consumption.**

***Keywords*— Cloud Computing, Multi-objective, Virtual Machine, Virtual Machine Placement.**

I. Introduction

Cloud computing refers to distributed environment where computing resources are distributed over a network. The main three service models are Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). Due to virtualization in cloud infrastructure, the placement of VM in an appropriate server is one of the major research problems [1]. VM placement is done by considering many objectives like usage of minimum number of servers, efficient use of power consumption and resource (e.g. CPU, memory, bandwidth) consumption. VM placement is also done by not violating the Service Level Agreement (SLA).

II. VM Placement Algorithms

The following are the VM placement algorithms which are surveyed on.

A. Static Server Allocation Problem (SSAP)

SSAP is framed for VM placement by Martin .B et al in [2]. The main objective of SSAP is to reduce the number of servers used for VM placement because it will minimize the overall cost of servers. In the pre-processing step of SSAP, all virtual machines (VMs) are assigned to servers manually. After manual placement of VMs to servers, the capacity of servers is reduced. Through VM monitor (i.e. hypervisor) lower and upper bound for capacity are set for each VM. The lower and upper bound capacity of VM specifies the minimum and maximum capacity that a VM can utilize respectively. But SSAP cannot handle the situation when the workload of each VM varies over time.

B. Static Server Allocation Problem with variable workload (SSAPv)

SSAPv [2] overcomes the disadvantage of SSAP by considering varying workloads of VM, when assigning VMs to servers. In SSAPv, a matrix u_{jt} is maintained for describing the maximum capacity of VM j needed in a time t.

C. Dynamic Server Allocation Problem (DSAP)

In cloud environment, VMs can be migrated to different servers over a time period for balancing the workload in datacenters. So, the number of servers needed and allocation of VMs to servers in a time period are considered and VM placement is done in DSAP [2]. By considering the history of workloads, DSAP is better when compared to assigning a VM to a particular server.

D. Multi-objective Ant Colony Optimization (ACO) Algorithm

Fei MA et al [3] proposed a multi-objective ACO algorithm to minimize SLA violation, resource wastage and power wastage in servers. In this multi-objective ACO algorithm, each ant constructs a solution for assigning VM to server. The constructed solution is evaluated by the suitable function specified in equation 4. The solution is more effective when the suitable function is bigger.

Suitable function for SLA violation

$$f_{SLA}(u_{CPU}) = 1\text{-}(1/(1+e^{uCPU\text{-}0.9})) \quad (1)$$

Suitable function for resource consumption

$$f_{resource}(u_{CPU},u_{mem},u_{bw}) = u_{CPU} * u_{mem} * u_{bw} \quad (2)$$

Suitable function for power consumption

$$f_{power}(u_{CPU}) = (u_{CPU}/(P_{idle}+(P_{busy} - P_{idle} * u_{CPU}))) * P_{busy} \quad (3)$$

Therefore, the total suitable function is

$$f = f_{SLA}(u_{CPU}) + f_{resource}(u_{CPU},u_{mem},u_{bw}) + f_{power}(u_{CPU}) \quad (4)$$

After the evaluation of constructed solution, if the constructed solution is in the list of best solutions, then its pheromone level is increased by increasing the pheromone evaporation rate ρ whose value lies between 0 and 1. Otherwise, the pheromone level is decreased by decreasing the value of pheromone evaporation rate ρ. Finally, the pheromone update rule is applied. By updating pheromone level, the optimal solution for VM placement was found.

E. Novel Vector Based Approach for Static VM Placement

Mayank .M et al [4] proposed a novel vector based approach for static VM placement for minimizing the total number of servers required for VM placement. The set of VMs with their resource requirements and set of servers with their capacity are given as inputs for this algorithm. In this algorithm, the VM which is the least imbalanced is placed in the first server and resource utilization vector (RUV) of that server is calculated. RUV is the vector addition of normalized utilization vector of each type of resource.

Each VM is taken and checked whether the server can host this VM. If the server can host this VM, then the vector addition of resource imbalance vector (RIV) of that server and RIV of that VM is calculated and finally the magnitude *M* of this vector addition is also calculated. At last, the VM which has the least magnitude *M* on the current server is placed on the current server and this VM is marked as ‘placed’. After placing each VM, the new RUV of the current server is calculated. After that if there is no place for VM in current server and there is one or more VMs have to be placed, then a new server is introduced and it is checked whether the remaining VMs can be placed in this new server. The above steps are repeated until all VMs are placed in given set of servers.

F. Novel Vector Based Approach for Dynamic VM Placement

The paper [4] discusses a novel vector based approach for dynamic VM placement which is used when a new VM reaches. The inputs for this algorithm are a set of servers with their current resource utilization and a new VM.

The goal of this dynamic VM placement algorithm is either to provide load balancing or server consolidation. According to the goal (e.g., load balancing or server consolidation), a list of potential servers is selected for placing the new VM. While selecting the potential servers, if the goal is to support load balancing then the servers with low resource utilization are selected and checked whether they support this new VM (checking of servers which support the new VM, starts from low to high resource utilization). If the goal is to provide server consolidation then the high utilization servers are selected for checking whether this server supports the new VM (checking of servers which support the new VM, starts from high to low resource utilization).

After getting a list of potential servers, take each server and compute the vector addition of RIV of server and VM. Also, the magnitude *M* of the vector addition is calculated. After the vector addition and their magnitude *M* are calculated, the server which has the lowest magnitude *M* is marked as the server for the new VM.

G. VM Scheduler Algorithm

Sameer K.M. et al [5] have designed a VM placement algorithm which represents the list of resources in a binary search tree (BST) instead of representing them in a queue. By this, the resource utilization is improved and VM allocation time is reduced. The input for the VM Scheduler algorithm is BST of VMs and the output is allocated servers.

The goals of this algorithm are to reduce the time of allocation of VM to server and to optimize the resource utilization. In various algorithms, the VM manager sends VM specifications to VM scheduler where as in this algorithm a BST is created for VM specifications and sent to VM scheduler. Servers are also listed in a BST. Using the BST of server and BST of VM, the VM scheduler will take the VM which has the maximum requirement and searches for a server which best fit the requirement of VM.

The server which best fit the VM specification is found by dividing required virtual machine specification (RVMS) by available host specification (AHS). If the value is 1 then that server perfectly fit the VM specification. If the value lies between 0 and 1 then that server is taken as a possible candidate and search for the best fit server continues. If the value is greater than 1 then that server will not fit the VM and so that server is rejected and search for the best fit server continues.

III. Comparison

The objectives, input, output for each VM algorithm are listed in Table 1.

TABLE I
COMPARISON OF DIFFERENT VM PLACEMENT ALGORITHMS

Algorithm	**Input**	**Output**	**Static/Dynamic**	**Objectives/ Goals**	**Workload considered**
Static Server Allocation Problem (SSAP) [2]			Static	To reduce number of servers required for reducing the overall server cost	
Static Server Allocation Problem with variable workload (SSAPv) [2]			Static	To reduce number of servers required for reducing the overall server cost	Yes
Dynamic Server Allocation Problem (DSAP) [2]	1. Number of servers needed 2. History VM allocation in a time period		Dynamic	To reduce number of servers required for reducing the overall server cost	Yes
Multi-objective Ant Colony Optimization (ACO) Algorithm [3]	1. VMs with specific resource needs		Static	To reduce resource wastage (CPU, memory, bandwidth), power wastage and SLA violation (QoS considered)	
Novel Vector Based Approach for Static VM Placement [4]	1. set of VMs with their resource requirements 2. set of servers with their capacity		Static	To reduce number of servers required	
Novel Vector Based Approach for Dynamic VM Placement [4]	1. set of servers with current resource utilization 2. new VM		Dynamic	To reduce number of servers required and load balancing and server consolidation were done	
VM Scheduler Algorithm [5]	1. Binary Search Tree (BST) of VMs	Allocated servers	Dynamic	To reduce VM allocation time and improve resource utilization	

IV. Conclusions

The VM placement is one of the research problems in cloud infrastructure. Some VM placement algorithms with different objectives were surveyed. VM placement should be done by considering two or more goals at the same time which will improves the overall performance of servers in a datacenter.

REFERENCES

[1] Q. Zhang, L. Cheng, R. Boutaba, Cloud computing: state-of-the-art and research challenges, J. Internet Services Appl. 1 (1), 2010, pp. 7–18.

[2] M. Bichler, T. Setzer, B. Speitkamp, Capacity planning for virtualized servers, in: Workshop on Information Technologies and Systems (WITS), 2006.

[3] Fei MA, Feng LIU, Zhen LIU, Multi-objective Optimization for Initial Virtual Machine Placement in Cloud Data Center, in: Journal of Information & Computational Science, 2012, pp. 5029–5038.

[4] Mayank Mishra, Anirudha Sahoo, On Theory of VM Placement: Anomalies in Existing Methodologies and Their Mitigation Using a Novel Vector Based Approach, in: IEEE International Conference on Cloud Computing (CLOUD), 2011, pp. 275–282.

[5] Sameer Kumar Mandal, Pabitra Mohan Khilar, Efficient Virtual Machine Placement for On-Demand Access to Infrastructure Resources in Cloud Computing, in: International Journal of Computer Applications, 2013, pp. 6–11.