

Efficient Migration –A Leading Solution for Server Consolidation

R Suchithra,
MCA Department
Jain University
Bangalore,India

N.Rajkumar, PhD.
Department of Software Engineering
Ramakrishna College of Engineering
Coimbatore,India

ABSTRACT

Server consolidation of virtual machines is very much essential in a cloud environment for energy conservation and cost cutting. Consolidation can be achieved through live migration of virtual machines. We propose a modified bin packing algorithm for Server Consolidation that avoids unnecessary migrations and minimizes the instantiation of new physical servers. We implement ideas from the First Fit algorithm (decreasing strategy) for live migration of virtual machines. We have simulated our algorithm using java with multiple test cases. The proposed system consequently results in Server Consolidation through minimal migrations.

Keywords

Virtualisation, migration, Server Consolidation

1. INTRODUCTION

During recent times, Cloud service providers like go grid, Amazon EC2 rent resources like CPU, memory to users based on pay as you use model [1], [17]. These providers provide Infrastructure as a service (IAAS). These providers use virtualisation techniques instead of physical services as virtualisation optimise complex resources in a scalable manager so that cloud computing service becomes very cost effective.

Virtualization replaces the old concept of “One Server One application Model” where multiple virtual machines can run in one physical server. The clients can deploy and destroy virtual machines depending upon his changing business and computing requirements through an API or web portal provided by the cloud service provider. In addition to the amazing benefits of reduction in IT cost, capital cost and operational expenses, Virtualisation increases the utilization rate of every server which in turn drastically reduces the number of servers the cloud services need to buy and minimises loss in revenue due to downtime failures.

Virtualisation eliminates server sprawling and under utilization of resources in a cloud environment. Excessive power consumption by data centers has become a critical issue, since utility computing has become popular. To reduce the energy cost and consumption, the unneeded physical servers where the virtual machines are deployed should be intelligently powered off without effecting the applications and users.

In this paper, we study on the deployment of virtual machines with dynamic resource requirements. We use the term “jobs” which indicates virtual machines. There are four kind of major resources like CPU, disk, memory, network bandwidth etc. We consider only CPU as the resource requirement in our

paper. For instance if we want to get virtual machines for Amazon EC2 [18], Amazon provides a simple web service interface through which the user can specify the resource requirements. The user has to pay only for the capacity he has used.

Energy management is a critical problem faced by most of the IT service centres. Huge consumption of energy resource is due to the inefficient utilization of the resources. According to [19] in service centres 30% of energy is consumed by idle servers which are not doing any work. A technique for energy conservation is to power on only few servers and loading the work on them. This term is defined as Server Consolidation.

The server consolidation problem, which has to decide how to rebalance the job loads within the existing servers without instantiating new ones, is an instance of the decision version of the classical NP-complete bin packing problem. The problem can be defined as follows: Consider a set of p servers, each with capacity c_1, \dots, c_p and a set of n jobs, where each job has a weight w_1, \dots, w_n representing the amount of resource that the job demands of server p . Virtual machines are used to run jobs. The key point is to map the resources in the configuration of the virtual machines to the configuration of the physical server. The total amount of required resources from virtual machines mapped to the same server will not exceed the amount of resource of that physical server. Exceeding the capacity would cause the overloaded physical server to fail or have unacceptable performance. The Server consolidation problem is also NP hard problem [20] but it slightly differs from the traditional bin packing problem where resource utilization has positive or negative correlation with other resource utilization.

To minimize the number of physical servers, much study has been done on Server Consolidation. We need an optimum heuristic algorithm for the Server Consolidation problem. Numerous bin packing algorithms have been used for the Server Packing Problem and First Fit Decreasing (FFD) [21] is the best among all. Items are packed into the existing bins. New bins are added only if the existing bins cannot accommodate new items. That is why we are using FFD for Server consolidation.

An open question is how the static bin packing algorithms (First Fit) can be used for Server Packing in a cloud computing environment where the resource requirements are dynamic. So we have to reformulate the algorithm so that virtual machines can be efficiently migrated among the physical servers so that we can give the solution for Server Packing efficiently. This is illustrated in fig 1. Migrating Virtual machines incur overhead cost. The solution we propose should migrate only minimal number of virtual

machines as migrating more virtual machines will increase the overhead cost and consumes more power.

The algorithm we propose will work in the following scenario. We assume that initially virtual machines are allotted to physical servers in First Fit order depending on the resource requirements. When there is no sufficient space in the instantiated servers to allocate a new job, our algorithm will migrate virtual machines, thus preventing turning on of new servers. This results in Server consolidation. We conducted a series of experiments using java to analyze the efficiency of our Consolidation algorithm. The results indicate that our algorithm minimizes the number of migrations.

The rest of the paper is organized as follows. Section II describes background work and study. Section III describes the mathematical formulation of our model. Section IV describes the proposed algorithm. Section V gives the experimentation results and analysis of our algorithm. Section VI gives a summary of our present work and suggests possible future extensions.

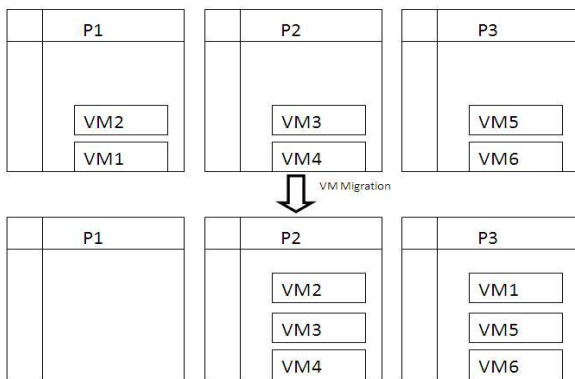


Figure 1: Server Consolidation

2. BACKGROUND AND RELATED WORK

The research regarding Server Consolidation and energy Conservation has shown significant advances in the past few years in both academics and industries [7], [8], [9]. This section discusses the work from both of these groups. The modified First Fit algorithm, proposed by B. Xia and Z. Tan. [7] uses the classical one dimension bin packing problem. The main aim of their algorithm is to decrease the gap between the upper and lower bound of absolute performance ratio. Beloglazov and Buyya [2] proposes an energy efficient system through dynamic reallocation of virtual machines by setting up lower and upper thresholds. Srikantiah et al. [3] have studied the relationship between resource utilization and consolidation of work loads. The authors have proposed multidimensional bin packing algorithm for server consolidation. Their algorithm is application dependent whereas ours is independent. Huang et al. [5] improved the efficiency of VM migration by using RDMA (Remote Direct Memory Access) as the protocol to transfer VM migration traffic. In the cited paper [4] X. Liao et al. proposes encloud that enables dynamic application live placement using virtual machines. The applications are dynamically abstracted using bin packing algorithms. Other relevant work in the area of resource allocation, reservation and scheduling can be found in [6].

Precopy algorithm is proposed for migration in [10]. A Server consolidation tool called Recon is introduced in [11]. It considers the migration cost and provides VM to server mapping. The greedy algorithms discussed in [15], [16] has drawbacks of more resource wastage as they are centralized. In the cited paper [12] wood has used the black box and grey box approach for monitoring and detecting CPU, memory and other resources and the algorithms helps in remapping of virtual machines. The migration aware server consolidation is used in [14]. Sanders .P, Sivadasan.N in [13] has proved that migration can reduce the job completion time by relocating jobs to the underutilized machines .

In all research works mentioned above, it has been concluded that, for the specific set of applications are considered, though migration has its adverse effects like increasing migration cost, it is acceptable and the bad effects can be ignored in consideration to the benefits that can be achieved through server consolidation.

3. PROBLEM FORMULATION

The Server Consolidation problem can be mathematically expressed as follows:

We start with a cloud computing environment consists of p physical servers on which virtual machines are hosted. Each virtual machine denotes one resource requirement (in our case CPU is considered as the resource) from the client. Assume that the set of physical servers in a cloud data center be P_1, P_2, \dots, P_n and resource capacity(CPU) of each server is taken as 100%. Let the jobs requesting for resources be J_1, J_2, \dots, J_k and the weights of the jobs is denoted by w_1, w_2, \dots, w_k . If a new job w_i cannot be allotted in anyone of the instantiated physical servers P_i as defined in (1) then we have to apply the condition of migration.

$$w_i > \left(100 - \sum_{j=1}^n w_j \right) \quad (1)$$

Where w_j is the weight of the new job.

Migration can be achieved if the below condition is satisfied (2). The free space of all p servers should be greater than the new job w_i .

$$\sum_{p=1}^n \left[\left(100 - \sum w_j \right) \right] > w_i \quad (2)$$

We use the following notation identify the target server P_k . The jobs in the targeted server is sorted using First in First Out(decreasing strategy) and a job is selected for migration. It is an obvious condition (3) that the server P_i with the maximum free space is selected as the targeted server P_k .

$$p_k = \text{Max} \left(100 - \sum_{j=1}^k [w_j] \right) \quad (3)$$

We use a criterion to identify the untargeted server p_j .

$$p_j = w_j \leq \left(100 - \sum [w_i] \right) \quad (4)$$

Where w_j is the job selected from p_k for migration

The condition to identify the job to be migrated from target to untargeted is yielded in the below equation.

$$\text{Max}(w_j) \text{ in } p_k \quad (5)$$

We can use a special condition to check the efficiency of migration which can be represented as follows:

$$(100 - p_k) + \text{Max}(w_j) \text{ in } p_k > w_i \quad (6)$$

If condition (6) is true then the new job w_i is accommodated in p_k (7) and w_j is relocated to p_j .

$$p_k = w_i \quad (7)$$

$$p_j = \text{Max}(w_j) \quad (8)$$

4. PROPOSED ALGORITHM FOR SERVER CONSOLIDATION

General Bin packing algorithms has many algorithms like First Fit, Worst Fit etc and we used the First Fit decreasing order algorithm. The FFD orders the items in descending order before packing the items to the bins. If items cannot be packed in the existing bins, then a new bin is opened to fit the new item into it. We used the ideas of FFD and propose a modified bin packing algorithm for Server Consolidation.

Our algorithm minimises the number of physical servers in a cloud data centre by packing the jobs in the existing servers that bears the heaviest work load that results in Server Consolidation. The aim is not only to keep minimum physical servers but also to minimise the number of migrations. Hence our model behaves differently from the FFD bin packing algorithm.

Input: S, T

Where

S → List of all servers

T → Task List

T_i → ith Task in T

BEGIN

allocated → false

sorted server list ← sort S by CPU descending

target ← Find the server with maximum CPU available

i.e., the first server in sorted server list

task list ← List all the tasks except from target

sorted task list ← sort T By CPU descending

for each selected task belongs to sorted task list

if (total available CPU > CPU of selected task)

then for each task belongs to target

if (task is migratable and CPU on target > CPU of selected task)

then migrate the task

allocate the selected task on to target

allocated ← **TRUE**

break (for each loop)

end if

end if

end for each

END

Our algorithm can be demonstrated in simpler terms like this :The algorithm iterates over a list of servers sorted based on free CPU available selected, the algorithm makes a list of running tasks on all servers leaving the tasks from the target server. Then algorithm sorts this list in ascending order based on the CPU requirement of the tasks. A task from this sorted task list is chosen as the task to be migrated. Once the task is chosen, the algorithm checks whether the task can be migrated to target server. If it can be migrated, then it checks whether the migration results in making enough free CPU available to allocate the current task from waiting task list. If the migration results in making enough free CPU available for allocation of current task, then the algorithm goes on to choose the next from the sorted task list to try with migration. Apparently the algorithm iterates over all the tasks from the sorted task list to find a feasible migration that results in allocation of the current task. If no feasible migration is possible, then it's concluded that there's no way to allocate the current task.

5. EXPERIMENTATION RESULTS

Our algorithm is simulated using java. We ran our algorithm for about ten times and made an average of the results. The results clearly show that our Server Consolidation algorithm is more promising than the FFD algorithms in terms of resource utilization of the physical servers as well as minimal migration of jobs (VMs).

To test the performance of Server utilisation before and after migration of jobs, we placed jobs (VMS) on four physical servers. For simplicity we assume that each physical server has CPU as the resource with 100% capacity. For better evaluation we vary the number of resources among the four

servers. Fig. 2 illustrates the performance status of server before the execution of our algorithm where Server 3 and server 4 are underutilized. Fig. 3 shows the implementation of migration technique for better server utilization.

Migration results in Server utilization where it tries to pack the jobs on the maximally utilized Servers thereby the minimally utilized server can be powered off. Fig 4 shows that migration has resulted in the fullest utilization of Server 1 and Server 2. The experimental results shows an interesting trade off that exists between migration and server consolidation. If high level of server utilization is needed, then more efficient migrations are important.

Figure 3 and Figure 4 shows that Servers are utilized efficiently after application of our Server consolidation algorithm.

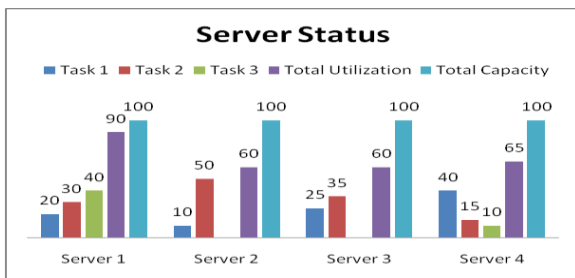


Figure 2: Graphical Representation of Server Status before migration

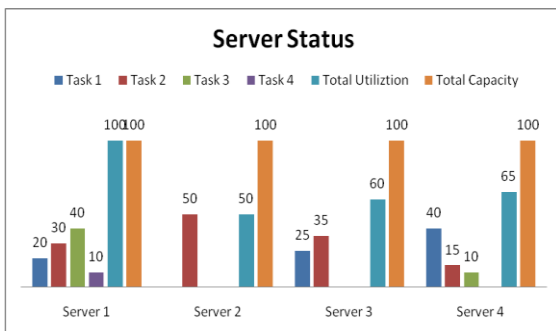


Figure 3: Graphical representation of Server Status after first iteration

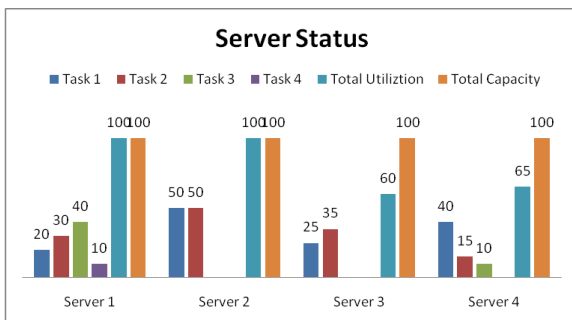


Figure 4: Graphical representation of Server Status after second iteration

6. CONCLUSION AND FUTURE WORK

The detailed experiments with different types of data have proved that Server Consolidation algorithms reduce the usage of more physical servers in a Cloud Data Centre. The experimental results partly validate our algorithmic model as the last two servers are still underutilized. This issue can be rectified in our future work where we are planning to implement parallel migration of more than one job at a time. This work concentrated only on the migration concept. Many other important issues like completion time of job, time taken for migration, and migration costs etc are not considered in this work. As a future work we would like to consider these issues and test our algorithm with complex cases and in better practical environment for actual evaluation.

7. REFERENCES

- [1] www.amazon.com
- [2] A. Beloglazov and R. Buyya. Energy efficient resource management in virtualized cloud data centers. Cluster Computing and the Grid, IEEE International Symposium on, 0:826–831,2010.
- [3] Srikantaiah, A. Kansal, and F. Zhao. Energy aware consolidation for cloud computing. Cluster Computing, 12:1– 15, 2009.
- [4] X. Liao, L. Hu, and H. Jin. Energy optimization schemes in cluster with virtual machines. Cluster Computing, 13:113–126, June 2010.
- [5] W. Huang, Q. Gao, J. Liu, and D. Panda, “High performance virtual machine migration with RDMA over modern interconnects,” in Proceedings of the IEEE International Conference on Cluster Computing, 2008, pp. 11–20.
- [6] J.M. Schopf and L. Yang, “Using Predicted Variance for Conservative Scheduling on Shared Resources,” in Grid Resource Management, eds. J. Nabryski, J.M. Schopf, and J.Weglarz, Kluwer Academic, 2003/Virtualization.aspx (2009).
- [7] B. Xia and Z. Tan. Tighter bounds of the first fit algorithm for the bin-packing problem. Discrete Appl. Math., 158:1668–1675, August 2010
- [8] M. Bichler, T. Setzer, B. Speitkamp, Capacity planning for virtualized servers, in: Proceedings of the 16th Annual Workshop on Information Technologies and Systems (WITS’06), 2006.
- [9] B. Speitkamp, M. Bichler, A Mathematical Programming Approach for Server Consolidation Problems in Virtualized Data Centers, IEEE transactions on Services Computing,2010
- [10] Y. Luo, B. Zhang, X. Wang, Z. Wang, Y. Sun, and H. Chen, “Live and incremental whole-system migration of virtual machines using block- bitmap,” in Proceedings of the IEEE International Conference on Cluster Computing, 2008, pp. 99–106.
- [11] S. Mehta, A. Neogi, ReCon: A Tool to Recommend Dynamic Server Consolidation in Multi-Cluster Data Centers, in: Proceedings of the IEEE Network Operations and Management Symposium (NOMS’08),Salvador, Bahia, 2008.

- [12] T. Wood, P. J. Shenoy, A. Venkataramani, M. S. Yousif, Sandpiper:Black-box and gray-box resource management for virtual machines, *Computer Networks* 53 (17) (2009) 2923– 2938.
- [13] SANDERS, P., SIVADASAN, N., AND SKUTELLA, M. Online scheduling with bounded migration. *Mathematics of Operations Research* 34, 2 (2009), 481–498
- [14] Murtazaev A, Oh s. Sercon:Server Consolidation Algorithm using Live migration of Virtual Machines for Green Computing , <http://tr.ijetejournals.org>,2011
- [15] M. Y. Lim, F. Rawson, T. Bletsch, and V. W. Freeh. Padd: Power aware domain distribution. *Distributed Computing Systems, International Conference on*, 0:239–247, 2009
- [16] B. Li, J. Li, J. Huai, T. Wo, Q. Li, and L. Zhong. Enacloud: An energy-saving application live placement approach for cloud computing environments. In *Proceedings of the 2009 IEEE International Conference on Cloud Computing, CLOUD '09*, pages 17–24, Washington, DC, USA, 2009. IEEE Computer Society.
- [17] www.gogrid.com
- [18] <http://aws.amazon.com/ec2/>
- [19] M.Uddin,A.Rahman “Server Consolidation: An approach to make Data Centers Energy Efficient and Green” , *International Journal of Engineering and Scientific Research* ,Volume 1,Issue 1,2010
- [20] V. V. Vazirani. *Approximation Algorithms*, chapter Bin Packing, pages 74–78. Springer Verlag, 2001.
- [21] A.Lodi,S.Martello and D.Virgo, Recent Advances on Two-dimensional Bin Packing Problem, *Discrete Applied Mathematics* 123:379- 396,2002