A Novel Virtual Machine Placement in Cloud Computing

[1]Eslam Mohammadi, [2]Mohammadbager Karimi, [3]Saeed Rasouli Heikalabad

[1,2]Technical and Engineering Dept., Tabriz Branch, Islamic Azad University, Tabriz, Iran.
[3]Young Researchers Club, Tabriz Branch, Islamic Azad University, Tabriz, Iran.

Abstrac: In this paper a novel virtual machine placement mechanism for cloud computing is proposed based on network-aware virtual machines (VMs). VMs are widely applied to modern data center for cloud computing as a key technology to realize energy-efficient operation of servers. In such a scenario, applications and data thereof can be hosted by various networked virtual machines. As applications, especially data-intensive applications, often need to communicate with data frequently, the network input and output performance would affect the overall application performance significantly. Therefore, placement of virtual machines which host an application and migration of these virtual machines while the unexpected network latency or congestion occurs is critical to achieve and maintain the application performance. Our paper proposes a virtual machine placement to minimizing the data transfer time consumption. The simulation results show that the proposed approach is effective in optimizing the data transfer between the virtual machine and data, thus helping optimize the overall application performance.

Key words: cloud computing; virtual machines; network; placement.

INTRODUCTION

Cloud computing (Armbrust *et al*., 2009) has recently received considerable attention in both academic community and industrial community as a new computing paradigm to provide dynamically scalable and virtualized resource as a service over the Internet. By this means, users will be able to access the resources, such as applications and data, from the cloud anywhere and anytime on demand. Currently, several large infrastructure companies, such Amazon (2010) Google (2010) Yahoo! (Cooper *et al*., 2009) Microsoft (2010) IBM (2010) and Sun, are developing cloud platforms for consumers and enterprises to access the cloud resources through services. Data center is traditional concept that provides powerful computing and storage capacity for crucial areas, such as particle physics, scientific computing and simulation, earth observation and oil prospecting and so on. A data center usually deploys hundreds or thousands of blade servers which are densely packed to maximize the space utilization and management efficiency. As the rapid growth of server quantity and scale, the energy consumed by the data center, which is directly related to the number of hosted servers and their workloads, is becoming a great challenge. As reported in (Raghavendra *et al*., 2008) the power consumption of worldwide enterprises exceeds $30 billion in 2008. The rated power consumptions of servers have increased by 10 times over the past ten years. What's more, it's worth noting that the server management and maintenance costs and electricity and cooling costs in modern data center have exceeded the server equipment costs. Due to the huge energy cost in data center, there is an urgent need of designing and deployment of energy-efficient technologies for building a green data center (Liu *et al*., 2009) Fortunately, The energy consumption problem has already attracted enough attention (The green 2010) Many efforts have been made to improve the energy efficiency of data center from different aspects including processor energy efficiency (Brooks and Martonosi 2001) storage power management (Carrera Pinheiro and Bianchini 2003) and network power management (Nedevschi *et al*., 2008). Recently, with the rapid development of virtualization technology, such as VMware (Waldspurger 2002) Xen (Barham *et al*., 2003) KVM (Kivity 2007) OpenVZ (Openvz 2010) more and more data centers use this technology to build new generation data center architecture to support cloud computing due to the benefits such as improving resource utilization, reducing costs, easing server management. What's more, server consolidation and live migration of virtual machine are two crucial methods to achieve load balancing and energy saving. Server consolidation which allowing multiple servers running in a single physical server simultaneously is a main approach to achieve better energy efficiency of data center. It is because in doing so, server consolidation allows more physical servers to be turned off via live migrating the virtual machines to other unsaturated physical servers.

In such an environment, the VM that executes an application is placed on a physical machine in order to use the local computation resources to execute the required tasks. At the same time, data can be stored with some geographical or logical distances (i.e. Amazon S3 etc) and these data are accessible to cloud-based applications (Brooks and Martonosi 2001) For a data-intensive application in cloud computing, the requested data might be spread in a number of vastly distributed data centers. As an application, especially a data-intensive application, often needs to communicate with related data frequently, the network I/O performance between the data centers that store the data and VMs that execute the applications could affect the performance of the applications

Corresponding Author: Eslam Mohammadi, Technical and Engineering Dept., Tabriz Branch, Islamic Azad
University, Tabriz, Iran
E-mail: eslammohammadi@gmail.com

significantly. Current VM placement policy mainly focuses on the effectiveness and efficiency of the computing resources utilization (Buyya Ranjan and Calheiros 2009: Tang *et al.*, 2007: Van Tran and Menaud 2009). whereas the network aspects are largely ignored. This might make a VM that executes an application be placed on physical machines that are far away from the data centers that store the related data. As a result, the overall application performance and the system overhead would eventually deteriorate due to the costly data transfer time between the application and the data storage. Furthermore, the virtualization and processor sharing over physical machines often result in the instability of the communication within a cloud computing environment. For example, the TCP/UDP throughput between the small instances in Amazon EC2 varies between 1Gb/s and 0 frequently (Guohui *et al.*, 2010). The unanticipated network congestion and latency places another challenge to optimizing the data transfer time between VMs and the related data. This research addresses the above issues and proposes a policy to place the VM with consideration of the network I/O requirement. In addition, a VM migration policy is presented to deal with the situation in which the unstable network connection deteriorates the application performance and likely to jeopardize the existing agreement between the cloud service provider and the end user. The rest of this paper is organized as follows. In Section 2, the related work is discussed. Section 3 explains the proposed network-aware VM placement and migration approach. After that, Section 4 presents the experimental results, followed by the conclusions and future work in Section 5.

Background And Motivation:

Various management strategies have been developed to effectively reduce the power consumption from different aspects; however they cannot be directly applied to today's data centers that rely on virtualization technologies. Virtual machine technology can efficiently manage the server consolidation, and improve the total power efficiency in data center (Nathuji and K. Schwan 2007).

Usually, contemporary VM placements approaches focus on optimizing the efficiency and effectiveness of computation resources utilization, allowing VMs to share the capacity of computation resources on the physical hosts. The VM placement problem is modeled as either the Constraint Satisfaction Problem (CSP) (Van Tran and Menaud 2009) or the Constraint Multiple Knapsack Problem (CMKP) (Tang *et al.*, 2007) to maximize the utility of computing resource. Other approaches focus on minimizing the cost of using computing resources (Kim and Ellis 2005). These approaches largely ignored the network performance and its impact on the overall application performance. VMs would be allocated with a non-optimized physical distance to the relevant data. This would then cause an extra data transfer overhead and eventually lead to the degradation of application completion time.

This drawback makes the statistic method based VM allocation or migration approaches hard to fit a runtime circumstance. The optimization approach should be implemented on the applications that interact with certain data in a relatively long term. Yet, if the applications communicate with data in a short time, the lack of statistics may cause these allocation or migration approaches inapplicable.

Proposed Approache:
Problem Statement:

Our problem statement can be briefly described as follows:

1- ' M' physical machines are available and their resource capacities given along memory, CPU and Network bandwidth dimensions.
2- There are 'N' virtual machines to be placed. The requirements of these virtual machines are given along the dimensions of memory, CPU and network bandwidth.
3- We have to find a mapping between VMs and PMs that satisfies the VMs' resource requirements while minimizing the number of physical machines used.

While finding such a mapping, we have to take care that the total resource requirement of the VMs placed on a PM should not exceed its capacity. Also, we plan to include the information of affinity between two virtual machines while obtaining such a mapping. We also plan to handle the case of availability of only a restricted set of PMs for a given VM for placement.

Scenario:

Data centers that are operated by a cloud not only provide a flexible data storage space (e.g Amazon S3 etc.) but also support the underlying virtualization infrastructure. This innovated technology facilitates the user to request data storage space and computation resources to form one or more VMs with arbitrary computational capacity.

Figure 1 shows the scenario in which users request the data storage space and VMs to store data and host applications to process these data, respectively. In this scenario, the applications request to access the related data across the Internet or Intranet and the link between the clouds might be logical or physical.
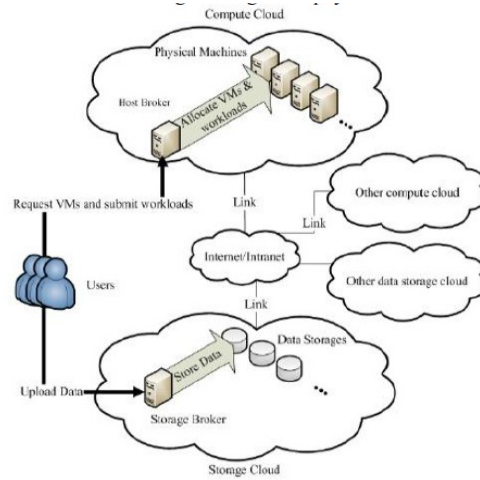
Fig. 1: The Studied Scenario.

Under current VM allocation policy, it can be seen that the data is stored arbitrarily and can be distributed vastly across the whole storage cloud or even over several storage clouds. The applications are allocated by the broker regardless of the data access time. This will result in the problem that the application may access its related data over a unnecessary distance. To solve this problem, the proposed VM placement and migration approach could be deployed in the host broker to allocate the VMs to the physical machine with a minimized data access time.

The VM placement Approach:
Within the context of cloud computing, the data for a given application could be distributed in several blocks and the blocks could be stored with logical/physical distances. According to the status of data distribution, a data distribute matrix $M_{i,j}$ can be implemented as:

$$M_{i,j} = \begin{bmatrix} m_{1,1} & m_{1,2} & ... & m_{1,j} \\ m_{2,1} & m_{2,2} & ... & m_{2,j} \\ ... & ... & ... & ... \\ m_{i,1} & m_{i,2} & ... & m_{i,j} \end{bmatrix}$$

Here, the column number j represents the total number of data which will be accessed by the applications and the row number n represents the number of data storage nodes.
In addition, for each column of the matrix $D_{i,j}$ each data should satisfy the following relationship:

$$Size_j = \sum_{c=1}^{i} m_{c,j} \tag{1}$$

The network speed between a physical machine which hosts the VM and a data storage node is represented by function *Speed*, namely

$$S = \text{Data transfer rate}(s, \Delta t) \tag{2}$$

Here, s represents the size of the package and Δt is the package transfer time slot. Within a certain time interval, the network speed could be fluctuated. In addition, the data transfer speed could be different for different applications.
The value of each element in matrix $S_{n,m}$ is the inverse of the *Speed(s, Δt)* between the physical machine and the data storage node. Therefore, the matrix $S_{n,m}$ can be obtained as follows:

$$S_{n,m} = \begin{bmatrix} S_{1,1} & S_{1,2} & \cdots & S_{1,m} \\ S_{2,1} & S_{2,2} & \cdots & S_{2,m} \\ \cdots & \cdots & \cdots & \cdots \\ S_{n,1} & S_{n,2} & \cdots & S_{n,m} \end{bmatrix}$$

One of our matrixes are the resource requirements of each virtual machine to be placed. To capture these requirements along various dimensions, we define a requirements matrix as follows:

$$\text{Requirements Matrix} = \begin{bmatrix} R_{1,1} & R_{1,2} & \cdots & R_{1,d} \\ R_{2,1} & R_{2,2} & \cdots & R_{2,d} \\ \cdots & \cdots & \cdots & \cdots \\ R_{n,1} & R_{n,2} & \cdots & R_{n,d} \end{bmatrix}$$

Where each r_{ij} indicates the requirement of VM_i along the dimension j.

A sample requirements matrix is shown below:

$$\begin{bmatrix} 0.2 & 0.4 & 0.3 \\ 0.2 & 0.4 & 0.5 \\ 0.6 & 0.2 & 0.3 \\ 0.6 & 0.2 & 0.5 \end{bmatrix}$$

In this matrix, each row corresponds to one virtual machine's requirements along the dimensions of CPU, memory and network I/O. For example, the first row specifies that the corresponding VM uses 20% of CPU, 40% of memory and 30% of network capacity of a physical machine.

The data access time matrix $T_{n,i}$ represents the data access time from each physical machine to the related data. The matrix $T_{n,i}$ can be obtained by following formula:

$$T_{n,i} = M_{i,j} \times S_{n,m} \times R_{n,d} \tag{3}$$

Assuming that the computing resource set:

$$C = \{P, M\} \tag{4}$$

Indicates the total computation capacity on a physical machine, where the notation P represents the processor capacity on the physical machine and M denotes the memory capacity on the physical machine. The set:

$$C_{occupied} = \{P_{VMi}, M_{VMi}\} \tag{5}$$

Represents the computation capacity requirement of VMi. Therefore, the available computing resource set can be represented as:

$$C_{Available} = \{P_{Available}, M_{Available}\} = \left\{ P - \sum_{i=1}^{n} P_{vmi}, M - \sum_{i=1}^{n} M_{vmi} \right\} \tag{6}$$

Where n indicates the number of running VMs. Using

$$C_{VMarriving} = \{P_{VMarriving}, M_{VMarriving}\} \tag{7}$$

Denotes the computation capacity requirement of the arriving VM, if the available computation resource on the physical machine satisfies

$$P - \sum_{i=1}^{n} P_{vmi} > P_{vmi} \tag{8}$$

And

$$M - \sum_{i=1}^{n} M_{vmi} > M_{vmi} \tag{9}$$

Then the VM would be placed on this physical machine. For simplicity, we use $C_{Available} \_\_ C_{VMArriving}$ to indicate the relationship in (10) and (11). After the placement of the arriving VM, the new available computation resource can be obtained by the formula:

$$C_{VMarriving} = \left\{ P_{VMarriving} - P_{VMi}, M_{VMarriving} - M_{VMi} \right\} \tag{10}$$

For α given application _ that requests data d1, d2,..., d_, the process of proposed network-aware VM placement algorithm can be demonstrated by the following pseudo code:

Input: *data storage matrix* $M_{i,j}$
Input: *network status matrix* $S_{n, m}$
Input: *network status matrix* $R_{n, d}$
Input: *a coming application requested data set*
1. Calculating data transfer time matrix
2. Traverse all of the column in the matrix $T_{n,i}$ *to find the*

$$minimum \sum_{y=1}^{\omega} T_{x,y} \ \&\& C_{Available} > C_{VMarriving}$$

return x;
Allocating the arriving VM on the host h_x.
if *(the VM is allocated succeed)*
$C_{Available} = \left\{ P_{Available} - P_{VMi}, M_{Available} - M_{VMi} \right\}$
end if

Experiment:

To evaluate the effectiveness of the proposed approach, an experiment has been designed and conducted. The proposed VM placement policy is implemented and tested using Cloudsim 2.0 which supports modeling and stimulating one or more VMs on a stimulated data center (Armbrust *et al.,* 2009; http://www.amazon.com/ec2, 2010). In the experiment, we focused on the average task completion time and compared our policy with the default VM placement policy adopted by Cloudsim 2.0. Besides, the matrix ($S_{n,m}$) is modified in the experiment to stimulate a degradation of data access status.

In the initial phase of the experiment, 2 files (i.e., data), *file1, file2,* are stored in 2 storages, storage0 and storage1. The size of the files is 6000MB, 8000MB respectively. Here, *file1* is stored in *storage0*, whereas *file2* stored in *storage1*. We implemented one data center, *datacenter0*, with 2 hosts (i.e., physical machines), *host0, host1*. Each host has a fixed computation capacity. The connection speeds between the hosts and storages are represented by a 3x 2 matrix, and the storage of data is defined by a 2x3 matrix. Based on the VM allocation policy, we can calculate a 3x3 data transfer time matrix. In this experiment, the data transfer time is set as:

$$\begin{bmatrix} 480.0 & 160.0 & 80.0 \\ 240.0 & 240.0 & 120.0 \\ 540.0 & 480.0 & 240.0 \end{bmatrix}$$

The comparisons of the average task completion time in two groups are shown in Figures 2, 3, respectively. It is clear that the proposed approach resulted in shorter average task completion time in all two groups. In fact, the decline of the average task completion time is caused by the optimized location of the hosted VMs, comparing

with two different VM allocation policies. The results proved that the proposed network aware VM allocation policy can lead to improved task completion time.
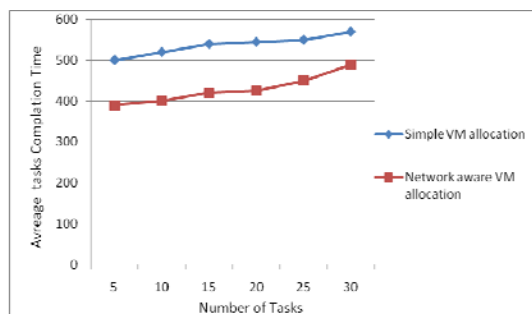

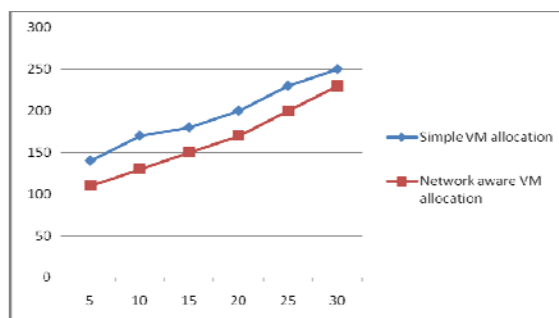**Fig. 2:** The Results of Cloudlets Requesting *file1*.

Fig. 3: The Results of Cloudlets Requesting *file2*.

Conclusions and Future Work:

While the data-intensive applications interact with distributed data in a cloud computing environment, the network status between the application and data could influence the performance of the application significantly. Thus, it is necessary to control the location of the VMs so that the applications hosted by the VM can obtain a shorter data access time. To address this issue, this paper presents a network aware VM placement and migration approach for data intensive applications in cloud computing environments. The proposed approach places the VMs on physical machines with consideration of the network conditions between the physical machines and the data storage.

Therefore, it is possible that a host is perfect for several requests simultaneously but the system would not be able to schedule all of the tasks in the cloud. This may lead to a possibility that some users' tasks always occupy a faster link on the network. To solve this issue, it is necessary to extend a payment mechanism so that the system can set priority to the users according to their payment.

REFERENCES

"Amazon elastic compute cloud (ec2)," *http://www.amazon.com/ec2*, 2010.

"Google app engine," *http://code.google.com/appengine*, 2010.

"Ibm 2010. blue cloud project," *http://www.ibm.com/ibm/cloud*.

"Microsoft 2010 azure cloud platform," *http://www.microsoft.com/windowsazure*.

"Openvz: Server virtualization open source project," *http://openvz.org*, 2010.

"The green 500 list," *http://www.green500.org/*, 2010.

Armbrust, M,. A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, 2009. *et al.*, "Above the clouds: A berkeley view of cloud computing," *EECS Department, University of California, Berkeley, Tech. Rep*.

Barham, P., B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield, 2003. "Xen and the art of virtualization," in *Proceedings of the nineteenth ACM symposium on Operating systems principles*, p: 177.

Brooks, D. and M. Martonosi, 2001. "Dynamic thermal management for high-performance microprocessors," in *The Seventh International Symposium on High-Performance Computer Architecture*, pp: 171-182.

Buyya, R., R. Ranjan and R.N. Calheiros, 2009. Modeling and simulation of scalable Cloud computing environments and the CloudSim toolkit: Challenges and opportunities. High Performance Computing & Simulation, HPCS '09. International Conference on.

Carrera, E., E. Pinheiro, and R. Bianchini, 2003. "Conserving disk energy in network servers," in *Proceedings of the 17th annual international conference on Supercomputing*, p: 97.

Cooper, B., E. Baldeschwieler, R. Fonseca, J. Kistler, P. Narayan, C. Neerdaels, T. Negrin, R. Ramakrishnan, A. Silberstein, U. Srivastava, *et al.*, 2009. "Building a Cloud for Yahoo!" *Data Engineering*, p: 36.

Guohui, W. and T.S.E. NG, 2010. The Impact of Virtualization on Network Performance of Amazon EC2 Data Center. INFOCOM, Proceedings IEEE.

Kim, J.L. and R.D. Ellis, 2005. A Framework for Integration Model of Resource-Constrained Scheduling using Genetic Algorithms. Simulation Conference, Proceedings of the Winter, 2119 – 2126, Orlando, Florida, USA.

Kivity, A., Y. Kamay, D. Laor, U. Lublin, and A. Liguori, 2007. "kvm: the Linux virtual machine monitor," in *Proceedings of the Linux Symposium*, 1: 225-230.

Liu, L., H. Wang, X. Liu, X. Jin, W. He, Q. Wang, and Y. Chen, 2009. "GreenCloud: a new architecture for green data center," in *Proceedings of the 6th international conference industry session on Autonomic computing and communications industry session*, pp: 29-38.

Nathuji, R. and K. Schwan, 2007. "VirtualPower: coordinated power management in virtualized enterprise systems," in *Proceedings of twenty-first ACM SIGOPS symposium on Operating systems principles*. ACM, p: 278.

Nedevschi, S., L. Popa, G. Iannaccone, S. Ratnasamy, and D. Wetherall, 2008. "Reducing network energy consumption via sleeping and rate-adaptation," in *Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation*, pp: 323-336.

Raghavendra, R., P. Ranganathan, V. Talwar, Z. Wang, and X. Zhu, 2008. "No power struggles: Coordinated multi-level power management for the data center," *ACM SIGPLAN Notices*, 43(3): 48-59.

Tang, C., M. Steinder, M. Spreitzer, and G. Pacifici. 2007. A Scalable Application Placement Controller for Enterprise Data Centers. Proceedings of the 16th international conference on World Wide Web, 331 – 340, Banff, Alberta, Canada.

Van, H.N., F.D. Tran, and J.M. Menaud, 2009. Autonomic Virtual Resource Management for Service Hosting Platforms. Software Engineering Challenges of Cloud Computing, CLOUD '09. ICSE Workshop on, Vancouver, Canada, 1-8.

Waldspurger, C., 2002. "Memory resource management in VMware ESX server," *ACM SIGOPS Operating Systems Review*, 36:SI, p. 194.