ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

Efficient Resources Allocation and Reduce Energy Using Virtual Machines for Cloud Environment

R.Giridharan

M.E. Student, Department of CSE, Sri Eshwar College of Engineering, Anna University - Chennai, India[1]

ABSTRACT - The rapid growth in demand for computational power driven by modern service applications combined with the shift to the Cloud computing model have led to the establishment of large-scale virtualized datacenters. Such datacenters consume enormous amounts of electrical energy resulting in high operating costs and carbon dioxide emissions. Dynamic consolidation of virtual machines (VMs) using live migration and switching idle nodes to the sleep mode allow Cloud providers to optimize resource usage and reduce energy consumption. However, the obligation of providing high quality of service to customers leads to the necessity in dealing with the energy-performance trade-off, as aggressive consolidation may lead to performance degradation. Due to the variability of workloads experienced by modern applications, the VM placement should be optimized continuously in an online manner. To understand the implications of the online nature of the problem, we conduct competitive analysis and prove competitive ratios of optimal online deterministic algorithms for the single VM migration and dynamic VM consolidation problems. Furthermore, we propose novel adaptive heuristics for dynamic consolidation of VMs based on an analysis of historical data from the resource usage by VMs. The proposed algorithms significantly reduce energy consumption, while ensuring a high level of adherence to the Service Level Agreements (SLA).

KEYWORDS - Cloud Computing, Resource Management, Virtualization, Green Computing.

I. INTRODUCTION

Cloud computing delivers infrastructure, platform, and software that are made available as subscription-based services in a pay-as-you-go model to consumers. These services are referred to as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) in industries. The importance of these services was highlighted in a recent report from the University of Berkeley as: "Cloud computing, the long-held dream of computing as a utility has the potential to transform a large part of the IT industry, making software even more attractive as a service".

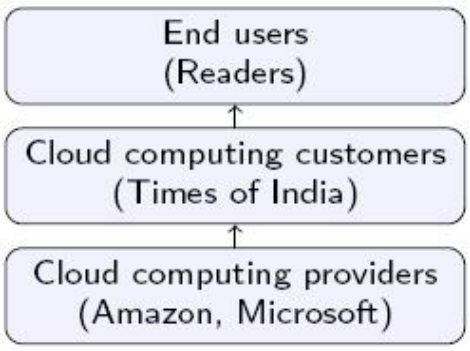

Figure 1 Cloud computing overview

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

Cloud computing(Figure 1) can be defined as "a type of parallel and distributed system consisting of a collection of inter-connected and virtualized computers that are dynamically provisioned, and presented as one or more unified computing resources based on service-level agreements established through negotiation between the service provider and consumers". Some of the examples for emerging Cloud computing infrastructures/platforms are Microsoft Azure, Amazon EC2, Google App Engine, and Aneka.

One implication of Cloud platforms is the ability to dynamically adapt (scale-up or scale-down) the amount of resources provisioned to an application in order to attend variations in demand that are either predictable, and occur due to access patterns observed during the day and during the night; or unexpected, and occurring due to a subtle increase in the popularity of the application service. This capability of clouds is especially useful for elastic (automatically scaling of) applications, such as web hosting, content delivery, and social networks that are susceptible to such behavior.

These applications often exhibit transient behavior (usage pattern) and have different QoS requirements depending on time criticality and users' interaction patterns (online/offline). Hence, the development of dynamic provisioning techniques to ensure that these applications achieve QoS under transient conditions is required. Even though Cloud has been increasingly seen as the platform that can support elastic applications, it faces certain limitations pertaining to core issues such as ownership, scale, and locality. For instance, a cloud can only offer a limited number of hosting capability (virtual machines and computing servers) to application services at a given instance of time, hence scaling application's capacity beyond a certain extent becomes complicated. Therefore, in those cases where the number of requests overshoots the cloud's capacity, application hosted in a cloud can compromise on overall QoS delivered to its users.

One solution to this problem is to inter-network multiple clouds as part of a federation and develop next-generation dynamic provisioning techniques that can derive benefits from the architecture. Such federation of geographically distributed clouds can be formed based on previous agreements among them, to efficiently cope with variation in services demands. This approach allows provisioning of applications across multiple clouds that are members of a/the federation. This further aids in efficiently fulfilling user SLAs through transparent migration of application service instance to the cloud in the federation, which is closer to the origins of requests.

A hybrid cloud model is a combination of private clouds with public clouds. Private and public clouds mainly differ on the type of ownership and access rights that they support. Access to private cloud resources is restricted to the users belonging to the organization that owns the cloud. On the other hand, public cloud resources are available on the Internet to any interested user under pay-as-you-go model. Hence, small and medium enterprises (SMEs) and governments have started exploring demand-driven provisioning of public clouds along with their existing computing infrastructures (private clouds) for handling the temporal variation in their service demands. This model is particularly beneficial for SMEs and banks that need massive computing power only at a particular time of the day (such as back-office processing, transaction analysis). However, writing the software and developing application provisioning techniques for any of the Cloud models – public, private, hybrid, or federated – is a complex undertaking. There are several key challenges associated with provisioning of applications on clouds: service discovery, monitoring, deployment of virtual machines and applications, and load-balancing among others. The effect of each element in the overall Cloud operation may not be trivial enough to allow isolation, evaluation, and reproduction.Cloud computing infrastructure models:

1.1 PUBLIC CLOUDS:

These are managed by third parties, and the applications requested from different customers are liable to be mixed together on the cloud's servers, storage systems, and networks. If a public cloud is employed with performance, security, and data locality in mind, the existence of other applications running in the cloud should be transparent to both cloud architects and end users. In reality, one of the benefits of public clouds is that they are much larger than a company's private cloud , offering the ability to provide on demand, and shifting infrastructure risks from the enterprise to the cloud provider, if even just temporarily.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

1.2 PRIVATE CLOUDS

These are built for the limited use of one client, providing the utmost control over data, security, and quality of service. The company owns the infrastructure and has control over how applications are deployed on it. These types of clouds can be built and managed by a company's own IT organization or by a cloud provider. This model gives companies a high level of control over the use of cloud resources while bringing in the capability needed to establish and operate the environment.

1.3 HYBRID CLOUDS

This type combines both the public and private cloud models. They can help to provide on-demand, externally provisioned scale. The ability to augment a private cloud with the resources of a public cloud can be used to maintain service levels in the face of rapid workload fluctuations. Sometimes called "surge computing," a public cloud can be used to perform periodic tasks that can be installed easily on a public cloud. A hybrid cloud is designed (Fig 1.1) by carefully determining the best split between public and private cloud components. One of the problems to face is to determine when and how to split a workflow, which is composed of dependent tasks, to execute in private re- sources and in public resources. We aim to achieve two goals in our algorithm:

- Overload avoidance. The capacity of a PM should be sufficient to satisfy the resource needs of all VMs running on it. Otherwise, the PM is overloaded and can lead to degraded performance of its VMs.
- Green computing. The number of PMs used should be minimized as long as they can still satisfy the needs of all VMs. Idle PMs can be turned off to save energy.

II. SYSTEM ANALYSIS

The main contributions of this work are the following.

1. Formal definitions of optimal online deterministic and offline algorithms for the single VM migration and dynamic VM consolidation problems.

2. A proof of the cost incurred by the optimal offline algorithm for the single VM migration problem.

3. Competitive analysis and proofs of the competitive ratios of the optimal online deterministic algorithms for the single VM migration and dynamic VM consolidation problems.

4. Novel adaptive heuristics for the problem of energy and performance efficient dynamic consolidation of VMs that outperform the optimal online deterministic algorithm.

Proposed system has conducted competitive analysis of the single VM migration and dynamic VM consolidation problems. We have found and proved competitive ratios for the optimal online deterministic algorithms for these problems. We have concluded that it is necessary to develop randomized or adaptive algorithms to improve upon the performance of the optimal deterministic algorithms. We have proposed novel adaptive heuristics that are based on an analysis of historical data on the resource usage for energy and performance efficient dynamic consolidation of VMs.

A. Create Cloud Setup

A simulation toolkit enables modeling and simulation of Cloud computing systems and application provisioning environments. The CloudSim toolkit supports both system and behavior modeling of Cloud system components such as data centers, virtual machines (VMs) and resource provisioning policies. It implements generic application provisioning techniques that can be extended with ease and limited effort. Currently, it supports modeling and simulation of Cloud computing environments consisting of both single and inter-networked clouds (federation of

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

clouds). Moreover, it exposes custom interfaces for implementing policies and provisioning techniques for allocation of VMs under inter-networked Cloud computing scenarios. In this module we are creating cloud users and datacenters and cloud virtual machines as per our requirement.

B. Predicting Future Resource Needs

We need to predict the future resource needs of VMs. As said earlier, our focus is on Internet applications. One solution is to look inside a VM for application level statistics, e.g., by parsing logs of pending requests. Doing so requires modification of the VM which may not always be possible. Instead, we make our prediction based on the past external behaviors of VMs. Our first attempt was to calculate an exponentially weighted moving average (EWMA) using a TCP-like scheme

$$E(t) = \alpha \times E(t-1) + (1-\alpha) \times O(t), 0 \leq \alpha \leq 1$$

where E(t) and O(t) are the estimated and the observed load at time t, respectively. α reflects a tradeoff between stability and responsiveness.We use the EWMA formula to predict the CPU load on the DNS server in our university. We measure the load every minute and predict the load in the next minute.

C. To measure the uneven utilization of a server

We introduce the concept of skewness to quantify the unevenness in the utilization of multiple resources on a server. Let n be the number of resources we consider and r_i be the utilization of the ith resource. We define the resource skewness of a server p as

$$skewness(p) = \sqrt{\sum_{i=1}^{n} (\frac{r_i}{\bar{r}} - 1)^2}$$

where $\bar{r}$ is the average utilization of all resources for server p. In practice, not all types of resources are performance critical and hence we only need to consider bottleneck resources in the above calculation. By minimizing the skewness, we can combine different types of workloads nicely and improve the overall utilization of server resources.

D. Hot and Cold Spots

Our algorithm executes periodically to evaluate the resource allocation status based on the predicted future resource demands of VMs. We define a server as a hot spot if the utilization of any of its resources is above a hot threshold. This indicates that the server is overloaded and hence some VMs running on it should be migrated away. We define the temperature of a hot spot p as the square sum of its resource utilization beyond the hot threshold:

$$temperature(p) = \sum_{r \in R} (r - r_t)^2$$

where R is the set of overloaded resources in server p and r_t is the hot threshold for resource r. (Note that only overloaded resources are considered in the calculation.) The temperature of a hot spot reflects its degree of overload. If a server is not a hot spot, its temperature is zero.

We define a server as a cold spot if the utilizations of all its resources are below a cold threshold. This indicates that the server is mostly idle and a potential candidate to turn off to save energy. However, we do so only when the average resource utilization of all actively used servers (i.e., APMs) in the system is below a green computing threshold. A server is actively used if it has at least one VM running. Otherwise, it is inactive. Finally, we define the

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

warm threshold to be a level of resource utilization that is sufficiently high to justify having the server running but not so high as to risk becoming a hot spot in the face of temporary fluctuation of application resource demands.

E. Eliminate all hot spots using Mitigation

We sort the list of hot spots in the system in descending temperature (i.e., we handle the hottest one first). Our goal is to eliminate all hot spots if possible. Otherwise, keep their temperature as low as possible. For each server p, we first decide which of its VMs should be migrated away. We sort its list of VMs based on the resulting temperature of the server if that VM is migrated away. We aim to migrate away the VM that can reduce the server's temperature the most. In case of ties, we select the VM whose removal can reduce the skewness of the server the most. For each VM in the list, we see if we can find a destination server to accommodate it. The server must not become a hot spot after accepting this VM. Among all such servers, we select one whose skewness can be reduced the most by accepting this VM. Note that this reduction can be negative which means we select the server whose skewness increases the least. If a destination server is found, we record the migration of the VM to that server and update the predicted load of related servers. Otherwise, we move onto the next VM in the list and try to find a destination server for it. As long as we can find a destination server for any of its VMs, we consider this run of the algorithm a success and then move onto the next hot spot. Note that each run of the algorithm migrates away at most one VM from the overloaded server.

F. Utilizations of all resources on active servers using green computing algorithm

Our green computing algorithm is invoked when the average utilizations of all resources on active servers are below the green computing threshold. We sort the list of cold spots in the system based on the ascending order of their memory size. Since we need to migrate away all its VMs before we can shut down an underutilized server, we define the memory size of a cold spot as the aggregate memory size of all VMs running on it. Recall that our model assumes all VMs connect to a shared back-end storage. Hence, the cost of a VM live migration is determined mostly by its memory footprint.

For a cold spot p, we check if we can migrate all its VMs somewhere else. For each VM on p, we try to find a destination server to accommodate it. The resource utilizations of the server after accepting the VM must be below the warm threshold. While we can save energy by consolidating underutilized servers, overdoing it may create hot spots in the future. The warm threshold is designed to prevent that. If multiple servers satisfy the above criterion, we prefer one that is not a current cold spot. This is because increasing load on a cold spot reduces the likelihood that it can be eliminated. However, we will accept a cold spot as the destination server if necessary. All things being equal, we select a destination server whose skewness can be reduced the most by accepting this VM. If we can find destination servers for all VMs on a cold spot, we record the sequence of migrations and update the predicted load of related servers. Otherwise, we do not migrate any of its VMs. The list of cold spots is also updated because some of them may no longer be cold due to the proposed VM migrations in the above process.

G. Efficient Dynamic Consolidation of Virtual Machines

We analyze a more complex problem of dynamic VM consolidation considering multiple hosts and multiple VMs. For this problem, we define that there are n homogeneous hosts, and the capacity of each host is A_h. Although VMs experience variable workloads, the maximum CPU capacity that can be allocated to a VM is A_v. Therefore, the maximum number of VMs allocated to a host when they demand their maximum CPU capacity is $m=A_h/A_v$. The total number of VMs is nm. VMs can be migrated between hosts using live migration with a migration time t_m.

The cost of power is C_p, and the cost of SLA violation per unit of time is Cv. Without loss of generality, we can define $C_p = 1$ and $C_v = s$, where $s \in R^+$. This is equivalent to defining $C_p = 1/s$ and $C_v = 1$. We assume that when a host is idle, i.e. there is no allocated VMs, it is switched off and consumes no power, or switched to the sleep mode with negligible power consumption. We call non-idle hosts active. The total cost C is defined as follows:

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

$$C = \sum_{t=t_0}^{T} (C_p \sum_{i=0}^{n} a_{ti} + C_v \sum_{j=0}^{n} v_{tj})$$

where t_0 is the initial time; T is the total time; $a_{ti} \in \{0,1\}$ indicating whether the host i is active at the time t; $v_{tj} \in \{0,1\}$ indicating whether the host j is experiencing an SLA violation at the time t. The problem is to determine what time, which VMs and where should be migrated to minimize the total cost C.

We split the problem of dynamic VM consolidation into four parts: (1) determining when a host is considered as being overloaded requiring migration of one or more VMs from this host; (2) determining when a host is considered as being under loaded leading to a decision to migrate all VMs from this host and switch the host to the sleep mode; (3) selection of VMs that should be migrated from an overloaded host; and (4) finding a new placement of the VMs selected for migration from the overloaded and under loaded hosts.

III. CONCLUSION

To maximize their ROI Cloud providers have to apply energy-efficient resource management strategies, such as dynamic consolidation of VMs and switching idle servers to power-saving modes. However, such consolidation is not trivial, as it can result in violations of the SLA negotiated with customers. In this work we have conducted competitive analysis of the single VM migration and dynamic VM consolidation problems. We have found and proved competitive ratios for the optimal online deterministic algorithms for these problems. We have concluded that it is necessary to develop randomized or adaptive algorithms to improve upon the performance of the optimal deterministic algorithms. According to the results of the analysis, we have proposed novel adaptive heuristics that are based on an analysis of historical data on the resource usage for energy and performance efficient dynamic consolidation of VMs.

In order to evaluate the proposed system in a real Cloud infrastructure, we plan to implement it by extending a real-world Cloud platform, such as OpenStack. Another direction for future research is the investigation of more complex workload models, e.g. models based on Markov chains, and development of algorithms that will leverage these workload models.

REFERENCES

[1] Agarwal .Y, Savage .S and Gupta .R (2010), 'Sleepserver: A SoftwareOnly Approach for Reducing the Energy Consumption of PCS within Enterprise Environments' Proc. USENIX Ann. Technical Conf.
[2] Armbrust et al .M (2009), 'Above the Clouds: A Berkeley View of Cloud Computing' technical report, Univ. of California, Berkeley.
[3] Barham .P, Dragovic .B, Fraser .K, Hand .S, Neugebauer .R and Warfield .A (2003), 'Xen and the Art of Virtualization' Proc. ACM Symp. Operating Systems Principles(SOSP '03).
[4] Bila .N, Joshi .K, Lagar-Cavilla .H.A, Hiltunen .M and Satyanarayanan .M (2012), 'Jettison: Efficient Idle Desktop Consolidation with Partial VM Migration' Proc. ACM European Conf. Computer Systems (EuroSys '12).
[5] Bobroff .N, Kochut.A and Beaty .K (2007), 'Dynamic Placement of Virtual Machines for Managing SLA Violations' Proc. IFIP/IEEE Int'l Symp. Integrated Network Management (IM '07).
[6] Chase .J.S, Anderson .D.C, Thakar .P.N, Vahdat .A.M, and Doyle .R.P(2001), 'Managing Energy and Server Resources in Hosting Centers' Proc. ACM Symp. Operating System Principles (SOSP '01).
[7] Clark .C,Fraser .K,Hand .S, Jul .E,Limpach .C,Pratt .I, and Warfield.A (2005),'Live Migration of Virtual Machines' Proc. Symp. Networked Systems Design and Implementation (NSDI '05).
[8] McNett .M, Gupta .D, Vahdat .A, and Voelker .G.M (2007), 'Usher: An Extensible Framework for Managing Clusters of Virtual Machines' Proc. Large Installation System Administration Conf (LISA '07).
[9] Nelson .M, Lim .B.H and Hutchins .G(2005), 'Fast Transparent Migration for Virtual Machines' Proc. USENIX Ann. Technical Conf.
[10]Waldspurger .C.A (2002), 'Memory Resource Management in VMware ESX Server' Proc. Symp. Operating Systems Design and Implementation (OSDI).
[11] Wood .T, Shenoy .P, Venkataramani .A and Yousif .M (2007), 'Black-Box and Gray-Box Strategies for Virtual Machine Migration' Proc. Symp. Networked Systems Design and Implementation (NSDI '07).