A Stochastic Reward Nets (SRNs) Model to Investigate Data Center Performance and QoS in IaaS Cloud Computing Systems

Sripada Rajinish[1], Barkat Amirali Jiwani[2]
[1]*M.Tech Student, Vaagdevi Engineering College, Warangal*
[2]*Assistant Professor, Department of CSE, Vaagdevi Engineering College, Warangal*

Abstract- **Cloud data center management is a key problem due to the numerous and heterogeneous strategies that can be applied, ranging from the VM placement to the federation with other clouds. Performance evaluation of Cloud Computing infrastructures is required to predict and quantify the cost-benefit of a strategy portfolio and the corresponding Quality of Service (QoS) experienced by users. Such analyses are not feasible by simulation or on-the-field experimentation, due to the great number of parameters that have to be investigated. In this paper, we present an analytical model, based on Stochastic Reward Nets (SRNs), that is both scalable to model systems composed of thousands of resources and flexible to represent different policies and cloud-specific strategies. Several performance metrics are defined and evaluated to analyze the behavior of a Cloud data center: utilization, availability, waiting time, and responsiveness. A resiliency analysis is also provided to take into account load bursts. Finally, a general approach is presented that, starting from the concept of system capacity, can help system managers to opportunely set the data center parameters under different working conditions.**

Index Terms- **Performance evaluation, Cost-benefit, Quality of service, Cloud-specific, Resiliency analysis**

I. INTRODUCTION

Cloud Computing is a promising technology able to strongly modify the way computing and storage resources will be accessed in the near future [1]. Through the provision of on demand access to virtual resources available on the Internet, cloud systems offer services at three different levels: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). In particular, IaaS clouds provide users with computational resources in the form of virtual machine (VM) instances deployed in the provider data center, while PaaS and SaaS clouds offer services in terms of specific solution stacks and application software suites, respectively. To integrate business requirements and application-level needs, in terms of quality of service (QoS), cloud service provisioning is regulated by service-level agreements (SLAs): contracts between clients and providers that express the price for a service, the QoS levels required during the service provisioning, and the penalties associated with the SLA violations. In such a context, performance evaluation plays a key role allowing system managers to evaluate the effects of different resource management strategies on the data center functioning and to predict the corresponding costs/benefits. Cloud systems differ from traditional distributed systems. First of all, they are characterized by a very large number of resources that can span different administrative domains. Moreover, the high level of resource abstraction allows us to implement particular resource management techniques such as VM multiplexing [2] or VM live migration [3] that, even if transparent to final users, have to be considered in the design of performance models to accurately understand the system behavior. Finally, different clouds, belonging to the same or to different organizations, can dynamically join each other to achieve a common goal, usually represented by the optimization of resources utilization. This mechanism, referred to as cloud federation [4], allows us to provide and release resources on demand, thus providing elastic capabilities to the whole infrastructure. For these reasons, typical performance evaluation approaches such as simulation or on-the-field measurements cannot be easily adopted. Simulation [5], [6] does not allow us to conduct comprehensive analyses of the system performance due to the great number of parameters that have to be investigated. On-the-field experiments [7], [8] are mainly focused on the offered QoS; they are based on a black box approach that makes difficult to correlate obtained data to the internal resource management strategies implemented by the system provider. On the contrary, analytical techniques [9], [10] represent a good candidate, thanks to the limited solution cost of their associated models. However, to accurately represent a cloud system, an analytical model has to be Scalable. To deal with very large systems composed of hundreds or thousands of resources. Flexible. Allowing us to easily implement different strategies and policies and to represent different working conditions. In this paper, we present a stochastic model, based on stochastic reward nets (SRNs) [11], that exhibits the above mentioned features allowing to capture the key concepts of an IaaS cloud system. The proposed model is scalable enough to represent systems composed of thousands of resources and it makes possible to represent both physical and virtual resources exploiting cloud-specific concepts such as the infrastructure elasticity Cloud Computing is a promising technology able to strongly modify the way computing and storage resources will be accessed in the near future [1]. Through the provision of on demand access to virtual resources available on the Internet, cloud systems offer services at three different levels: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). In particular, IaaS clouds provide users with computational resources in the form of virtual machine (VM) instances deployed in the provider data center, while PaaS and SaaS clouds offer services in terms of specific solution stacks and application software suites, respectively. Cloud systems differ from traditional distributed systems. First of all, they are characterized by a very large number of resources that can span different administrative domains. Moreover, the high level of resource abstraction allows to implement particular resource management techniques such as VM multiplexing [2] or VM live migration [3] that, even if transparent to final users, have to be considered in the design of performance models in order to accurately understand the system behavior. Finally, different clouds, belonging to the same or to different organizations, can dynamically join each other to achieve a common goal, usually represented by the optimization of resources utilization. This mechanism, referred to as cloud federation [4], allows to provide and release resources ondemand thus providing elastic capabilities to the whole infrastructure.

Stochastic Reward Nets (SRNs) [11], that exhibits the above mentioned features allowing to capture the key concepts of an IaaS cloud system. The proposed model is scalable enough to represent systems composed of thousands of resources and it makes possible to represent both physical and virtual resources exploiting cloud specific concepts such as the infrastructure elasticity. With respect to the existing literature, the innovative aspect of the present work is that a generic and comprehensive view of a cloud system is presented. Low level details, such as VM multiplexing, are easily integrated with cloud based actions such as federation, allowing to investigate different mixed strategies. An exhaustive set of performance metrics are defined regarding both the system provider (e.g., utilization) and the final users (e.g., responsiveness). Moreover, different working conditions are investigated and a resiliency analysis is provided to take into account the effects of load bursts. Finally, to provide a fair comparison among different resource management strategies, also taking into account the system elasticity, a performance evaluation approach is described. Such an approach, based on the concept of system capacity, presents a holistic view of a cloud system and it allows system managers to study the better solution with respect to an established goal and to opportunely set the system parameters.

With respect to the existing literature, the innovative aspect of the present work is that a generic and comprehensive view of a cloud system is presented. Low-level details, such as VM multiplexing, are easily integrated with cloud-based actions such as federation, allowing us to investigate different mixed strategies. An exhaustive set of performance metrics is defined regarding both the system provider (e.g., utilization) and the final users (e.g., responsiveness). Moreover, different working conditions are investigated and a resiliency analysis is provided to take into account the effects of load bursts. Finally, to provide a fair comparison among different resource management strategies, also taking into account the system elasticity, a performance evaluation approach is described

II.RELATED WORK

Live migration of virtual machines (VM) across physical hosts provides a significant new benefit for administrators of data centers and clusters. Previous memory-to-memory approaches demonstrate the effectiveness of live VM migration in local area networks (LAN), but they would cause a long period of downtime in a wide area network (WAN) environment [9]. This paper describes the design and implementation of a novel approach, namely, CR/TR -Motion, which adopts check pointing/recovery and trace/replay technologies to provide fast, transparent VM migration for both LAN and WAN environments. With execution trace logged on the source host, a synchronization algorithm is performed to orchestrate the running source and target VMs until they reach a consistent state. CR/TR-Motion can greatly reduce the migration downtime and network bandwidth consumption [8].

Modeling VM multiplexing

When VM multiplexing is allowed, the number of running VMs can be greater than N, i.e., $0 \leq P\#run \leq M$ and each PM can be loaded with more than one VM. Assuming an optimal scheduling algorithm able to balance the load among the N PMs, the maximum multiplexing level l reached by each PM (i.e., the maximum number of VMs running on a single PM)

The set J (with cardinality |J| = P#run) of the instantiated VMs can be partitioned into two sets Jl and Jl−1 (with J =Jl ∪ Jl−1) that correspond to the set of VMs running with a multiplexing level equal to l and the set of VMs running with a multiplexing level equal to l − 1, respectively. The cardinality of such sets can be obtained as: To model such a configuration, we need to calculate the equivalent rate of transition Tserv taking into account the parallel execution of the VMs and the performance degradation factor d. We start calculating the expected execution time TP#run averaged among all the P#run running VMs:

Then, in order to also consider the VM parallel execution, in accordance to the infinite server semantic, the marking dependent rate of transition Tservice when P# run VMs are running on N PMs is given by multiplying the number of running VMs by the average execution rate.

It can be noticed that, when $1 < P\# run \leq N$, for an illustrative example of the multiplexing strategy implementation, refer to the Appendix B of the supplementary file.

Understanding the model complexity

In order to respect the scalability requirement of the proposed model, we need to analyze its complexity. In particular, we are interested in the analysis of the state space cardinality that is the parameter that mainly influences the performance of the numerical solution techniques. The state space S of the model is given by the set of all its tangible markings [18]. Considering the SRN of and the corresponding guard functions we can make some considerations on the token distributions. Tokens in places Pqueue, Pres, and Prun are governed by the following rules giving rise to a number of possible combinations of the token distribution in such places equal to M +Q+1. Each of these M+Q+1 combinations can be obtained with zero or one token in place Pmmpp and with a number of tokens in place Psend ranging from 0 to D. The cardinality of the state space S is then given by:

$$|S| = 2 \cdot (D + 1) \cdot (M + Q + 1)$$

and the model complexity is $O(D \cdot (M + Q))$. However, being normally M _ D,Q we can state that the model complexity grows linearly with the number of logical resources of the cloud system under exam and we can definitely assert that the model is scalable and that it is also suitable for on-line performance analyses.

Cloud computing aims to power the next generation data centers and enables application service providers to lease data center capabilities for deploying applications depending on user QoS (Quality of Service) requirements. Cloud applications have different composition, configuration, and deployment requirements. Quantifying the performance of resource allocation policies and application scheduling algorithms at finer details in Cloud computing environments for different application and service models under varying load, energy performance, and system size is a challenging problem to tackle [11].Cloud computing is an emerging infrastructure paradigm that promises to eliminate the need for companies to maintain expensive computing hardware. Through the use of virtualization and resource time-sharing, clouds address with a single set of physical resources a large user base with diverse needs.

Advanced computing on cloud computing infrastructures can only become viable alternative for the enterprise if these infrastructures can provide proper levels of non functional properties (NPFs). A company that focuses on service-oriented architectures (SOA) needs to know what configuration would provide the proper levels for individual services if they are deployed in the cloud. In this paper we present an approach for performance evaluation of cloud computing configurations. While cloud computing providers assure certain service levels, this it typically done for the platform and not for a particular service instance [15].Cloud Computing is emerging today as a commercial infrastructure that eliminates the need for maintaining expensive computing hardware. Through the use of virtualization, clouds promise to address with the same-shared set of physical resources a large user base with different needs. Thus, clouds promise to be for scientists an alternative to clusters, grids, and supercomputers. However, virtualization may induce significant performance penalties for the demanding scientific computing workloads.

In this work we present an evaluation of the usefulness of the current cloud computing services for scientific computing. We analyze the performance of the Amazon EC2 platform using micro-benchmarks and kernels. While clouds are still changing, our results indicate that the current cloud services need an order of magnitude in performance improvement to be useful to the scientific community [18].

III.PROPOSED SYSTEM

Here is the proposed system based on Stochastic Reward Nets (SRNs), that exhibits the above mentioned features allowing capturing the key concepts of an IaaS cloud system [29]. The proposed model is scalable enough to represent systems composed of thousands of resources and it makes possible to represent both physical and virtual resources exploiting cloud specific concepts such as the infrastructure elasticity. With respect to the existing literature, the innovative aspect of the present work is that a generic and comprehensive view of a cloud system is presented. Low level details, such as VM multiplexing, are easily integrated with cloud based actions such as federation, allowing to investigate different mixed strategies. An exhaustive set of performance metrics is defined regarding both the system provider (e.g., utilization) and the final users (e.g., responsiveness).

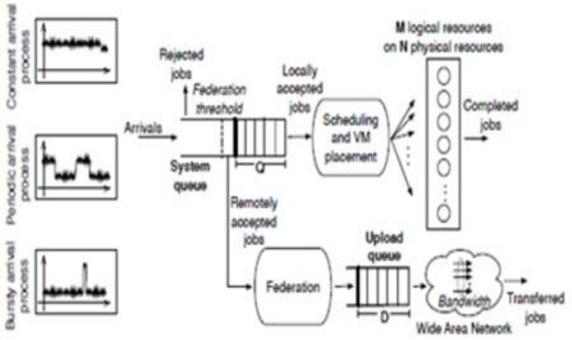

Fig.1. An IaaS Cloud System with federation

There are five modules in this system:

1. System Queuing
2. Scheduling Module
3. VM Placement Module
4. Federation Module
5. Arrival Process

A. System Queuing:

A queue management system is used to control queues. Queues of people form in various situations and locations in a queue area. The process of queue formation and propagation is defined as queuing. Job requests (in terms of VM instantiation requests) are en-queued in the system queue. Such a queue has a finite size Q, once its limit is reached further requests are rejected. The system queue is managed according to a FIFO scheduling policy.

B. Scheduling Module:

Scheduling is the process of arranging, controlling and optimizing work and workloads in a production process or manufacturing process. Scheduling is used to allocate plant and machinery resources, plan human resources, plan production processes and purchase materials. When a resource is available a job is accepted and the corresponding VM is instantiated. We assume that the instantiation time is negligible and that the service time (i.e., the time needed to execute a job) is exponentially distributed with mean $1/\mu$.

C. VM Placement:

Virtual Machine (VM) placement is a critical operation which is conducted as part of the VM migration and aimed to find the best Physical Machine (PM) to host the VMs. It has a direct effect on the performance, resource utilization and power consumption of the data centers and can reduce the maintenance cost of the data centers for cloud providers. According to the VM multiplexing technique the cloud system can provide a number M of logical resources greater than N. In this case, multiple VMs can be allocated in the same physical machine (PM), e.g., a core in a multi-core architecture. Multiple VMs sharing the same PM can incur in a reduction of the performance mainly due to I/O interference between VMs.

D. Federation Module:

Federation Module allows organizations to manage and secure access to cloud-based services by federating existing user credentials. Cloud federation allows the system to use, in particular situations, the resources offered by other public cloud systems through a sharing and paying model. In this way, elastic capabilities can be exploited in order to respond to particular load conditions. Job requests can be redirected to other clouds by transferring the corresponding VM disk images through the network.

E. Arrival Process:

Finally, we respect to the arrival process we will investigate three different scenarios. In the first one (Constant arrival process) we assume the arrival process be a homogeneous Poisson process with rate λ. However, large scale distributed systems with thousands of users, such as cloud systems, could exhibit self-similarity/long-range dependence with respect to the arrival process. The last scenario (Bursty arrival process) takes into account the presence of a burst with fixed and short duration and it will be used in order to investigate the system resiliency.

IV.ANALYTICAL MODEL

We consider an IaaS cloud system composed of N physical resources (see Fig. 1). Job requests (in terms of VM instantiation requests) are enqueued in the system queue. Such a queue has a finite size Q; once its limit is reached, further requests are rejected. The system queue is managed according to a FIFO scheduling policy. When a resource is available, a job is accepted and the corresponding VM is instantiated. We assume that the instantiation time is negligible and that the service time (i.e., the time needed to execute a job) is exponentially distributed with mean 1=_. According to the VM multiplexing technique the cloud system can provide a number M of logical resources greater than N. In this case, multiple VMs can be allocated in the same physical machine (PM), for example, a core in a multicore architecture. Multiple VMs sharing the same PM can incur in a reduction of the performance mainly due to I/O interference between VMs. We define the degradation factor d (0) as the percentage increase in the expected service time experienced by a VM when multiplexed with another VM.

The performance degradation of multiplexed VMs depends on the multiplexing technique and on the VM placement strategy. We assume that, to reduce the degradation and to obtain a fair distribution of VMs, the system is able to optimally balance the load among the PMs with respect to the resources required by VMs (e.g., trying to multiplex CPU-bound VMs only with I/ O-bound VMs), thus reaching a homogeneous degradation factor. Then, indicating with T ¼ 1=_ the expected service time of a VM in isolation, we can derive the expected time needed to execute two multiplexed VMs as T2

V.CONCLUSION

In this paper, we have presented a stochastic model to evaluate the performance of an IaaS cloud system. Several performance metrics have been defined, such as availability, utilization, and responsiveness, allowing to investigate the impact of different strategies on both provider and user point-of-views. In a market-oriented area, such as the Cloud Computing, an accurate evaluation of these parameters is required in order to quantify the offered QoS and opportunely manage SLAs. Future works will include the analysis of autonomic techniques able to change on-thefly the system configuration in order to react to a change on the working conditions. We will also extend the model in order to represent PaaS and SaaS Cloud systems and to integrate the mechanisms needed to capture VM migration and data center consolidation aspects that cover a crucial role in energy saving policies.

REFERENCES

[1] R. Buyyaet al., "Cloud computing and emerging it platforms :Vision, hype, and reality for delivering computing as the 5thutility," FutureGener. Comput. Syst., vol. 25, PP. 599–616, June 2009.

[2] X. Menget al., "Efficient resource provisioning in compute clouds via vm multiplexing," in Proceedings of the 7th internationalconference on Autonomic computing, ser. ICAC '10. New York, NY, USA:ACM, 2010, pp. 11–20.

[3] H. Liu et al., "Live virtual machine migration via asynchronous replication and state synchronization," Parallel and Distributed Systems,IEEE Transactions on, Vol. 22, No. 12,PP. 1986 –1999, December - 2011.

[4] B. Rochwergeret al., "Reservoir - when one cloud is not enough, "Computer, vol. 44, no. 3, pp. 44 –51, march 2011.

[5] R. Buyya, R. Ranjan, and R. Calheiros, "Modeling and simulation of scalable cloud computing environments and the cloud sim toolkit:Challenges and opportunities," in High Performance Computing Simulation, 2009. HPCS '09. International Conference on,PP. 1 – 11,June - 2009.

[6] A. Iosup, N. Yigitbasi, and D. Epema, "On the performance variability of production cloud services," in Cluster, Cloud and GridComputing (CCGrid), 2011 11th IEEE/ACM International Symposiumon, may 2011, pp. 104 –113.

[7] K.G.S. Venkatesan and M. Elamurugaselvam, "Design based object oriented Metrics to measure coupling & cohesion", International journal ofAdvanced & Innovative Research, Vol. 2, Issue 5, PP. 778 – 785, 2013.

[8] Teerawat Issariyakul • Ekram Hoss, "Introduction to Network Simulator NS2".

[9] S. Sathish Raja and K.G.S. Venkatesan, "Email spam zombies scrutinizer in email sending network Infrastructures", Internationaljournal of Scientific & Engineering Research, Vol. 4, Issue 4, PP. 366 – 373, April 2013.

[10] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," IEEE J. Sel. Areas Communication., Vol. 18, No. 3, PP. 535–547, Mar. 2000.

[11] K.G.S. Venkatesan, "Comparison of CDMA & GSM Mobile Technology", Middle-East Journal of Scientific Research, 13 (12),PP. 1590 – 1594, 2013.

[12] P. Indira Priya, K.G.S.Venkatesan, "Finding the K-Edge connectivity in MANET using DLTRT, International Journal of Applied Engineering Research, Vol. 9, Issue 22, PP. 5898 – 5904, 2014.