ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering
(IJERCSE)
Vol 4, Issue 12, December 2017

Managing task allocation in Cloud Computing environment by Multi-agent system

[1] Manoj Sharma, [2]Dr. Manoj Kumar Sharma, [3] Dr. S. Srinivasan
[1] Research Scholar, Department of C.S.E., Suresh Gyan Vihar University Jaipur, Rajasthan
[2] Professor, Department of C.S.E., Suresh Gyan Vihar University Jaipur, Rajasthan
[3] Professor, PDM University, Bahadurgarh, Haryana.

***Abstract*— Task allocation is an important problem in grid/cloud environments in both research and applications. With the rapid development of grid/cloud environments, the features of openness and dynamism of environments put two new challenges to the development of task allocation approaches and strategies in such environments. Firstly, the participants in the environments normally have only local views about the environments due to the administrative independencies between the participants and the limited communication abilities of the participants. Secondly, task allocation methods/ approaches have to handle the dynamism and openness of the environments. In particular, task allocation methods/approaches have to respond to and be resilient from the unpredicted changes in the environments in a quick manner. In the proposed method, both consumers and providers only have local views about the environment. Consumers and providers trade with each other through negotiations in which they make their offer (count-offer) decisions strategically through taking the issues that they are concerned with into account. The experimental results show that the proposed method can achieve desirable performances in terms of the success rate of and profit obtained from the task allocation.**

Keywords— Cloud computing; intelligent agent; load balancing; fitness value; load percentage;

I. INTRODUCTION

Cloud computing provide elastic services, high performance and scalable data storage to a large and everyday increasing number of users [1]. Cloud computing enlarged the arena of distributed computing systems by providing advanced Internet services that complement and complete functionalities of distributed computing provided by the Web, Grid computing and peer-to-peer networks. In fact, Cloud computing systems provide large-scale infrastructures for high-performance computing that are dynamically adapt to user and application needs.

Today Clouds are mainly used for handling highly intensive computing workloads and for providing very large data storage facilities. Both these goals are combined with the third goal of potentially reducing management and use costs. At the same time, multi-agent systems (MAS) represent another distributed computing paradigm based on multiple interacting agents that are capable of intelligent behavior.

Multi-agent systems are often used to solve problems by using a decentralized approach where several agents contribute to the solution by cooperating one each other. One key feature of software agents is the intelligence that can be embodied into them according to some collective artificial intelligence approach that needs cooperation among several agents that can run on a parallel or distributed computer to achieve the needed high performance for solving large complex problems keeping execution time low.

Although several differences exist between Cloud computing and multi-agent systems, they are two distributed computing models, therefore several common problems can be identified and several benefits can be obtained by the integrated use of Cloud computing systems and multi-agents. The research activities in the area of Cloud computing are mainly focused on the efficient use of the computing infrastructure, service delivery, data storage, scalable virtualization techniques, and energy efficiency. In summary, we can say that in Cloud computing the main focus of research is on the efficient use of the infrastructure at reduced costs. On the contrary, research activities in the area of agents are more focused on the intelligent aspects of agents and on their use for developing complex applications. Here the main problems are related to issues such as complex system simulation, adaptive systems, software-intensive applications, distributed computational intelligence, and collective learning.

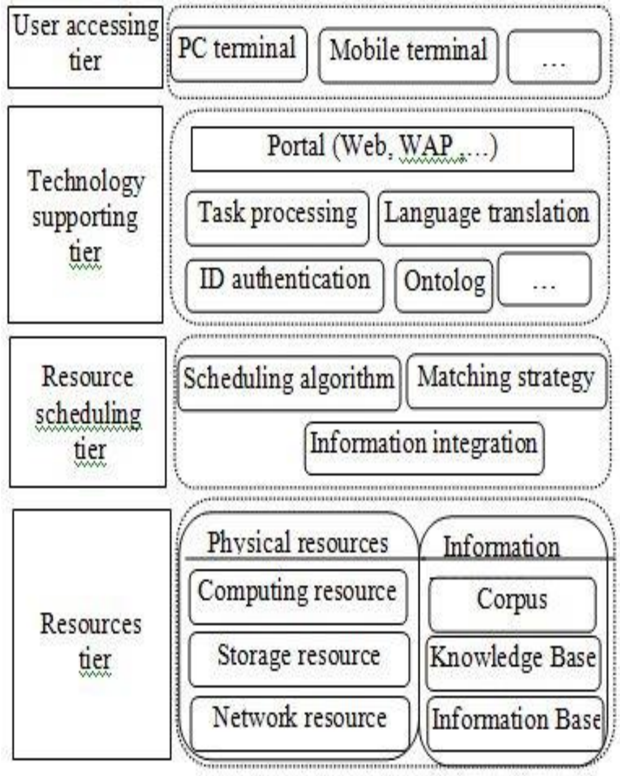

Fig. 1. Scheduling architecture

Despite these differences, Cloud computing and multiagent systems share several common issues and research topics in both areas have several overlaps that need to be investigated. In particular, Cloud computing can offer a very powerful, reliable, predictable and scalable computing infrastructure for the execution of multi-agent systems implementing complex agent-based applications such when modeling and simulation of complex systems must be provided. On the other side, software agents can be used as basic components for implementing intelligence in Cloud computing systems making them more adaptive, flexible, and autonomic in resource management, service provisioning and in running large-scale applications.

For these reasons and for others that we discuss later, this paper investigates research work in the two areas and point out potential synergies that deserve to be analyzed. The paper discusses Cloud computing models and architectures, their use in parallel and distributed applications, and examines analogies, differences and potential synergies between Cloud computing and multi-agent systems. Analysis is led having in mind the goal of implementing high-performance complex systems and intelligent applications by using both Cloud computing systems and software agents. Section II introduces loud computing concepts and reviews some research activities. Section III discusses multi-agent systems and some research topics that are related to Cloud computing. Section IV presents some ideas on using intelligent software agents to improve the performance and functionality of Clouds, whereas Section V discusses how Cloud computing platforms can be used for the efficient execution of MAS. Section VI concludes the paper.

II. RELATED WORK

The section throws light on the work of some renowned researchers who had been pillars and founders of the current research work. Research on resource management strategies in different fields (Chia-Ming et al., 2014) of distributed computing with different policies is not new. However in CC, dynamic resource provisioning (Quang-Hung et al., 2014) without delay or any compromise on delay is of utmost concern. Since, ubiquity and cost-effectiveness are two keywords describing CC, cost effectiveness centers on optimal resource allocation. Literature has been reviewed to explore existing strategies of resource allocation and scope of improvement. Buyya et al. (2002, 2003) presented resource allocation frameworks which could optimize the objective function for users and resource providers. Li et al. (2009) offered scheduling and optimization techniques based on Service Level Agreement (SLA) ignoring the throughput and response time of data centers. Bennani and Menasce (2005) presented a predictive multi-class queuing network model for computing the mean response time but the model was not good enough to evaluate the cost in case server switches from one application to another. Singh et al. (2015) have presented an agent based load balancing mechanism. Arfeen and his coworkers (Arfeen et al., 2011) focused on network awareness and consistent optimization of resource allocation strategies and highlighted the research issues prevailing in this field. Zhang et al. (2010) emphasized that more efforts are required to make the existing performance models predictive and responsive. Zheng et al. (2009) proposed a binary integer programming method to solve independent optimization but linear problems only and is not suitable for dynamic and complex problems. Also, a few authors (Christodoulou et al., 2007; Doulamis et al., 2007) had proposed the game theoretic method to solve the optimization of resource allocation in network systems from the resource providers' perspectives. Ji et al. (2014) proposed a job scheduling algorithm based on greedy approach. Authors have implemented their algorithm in cloud environment and indicated success in reducing completion time of a task. Their implementation divides the tasks based on completion time and bandwidth requirements. However, in case resources are not found in a particular data center, this issue has not been paid attention. Hassan and Alamri (2014) proposed a resource allocation mechanism based on Nash Bargaining system for multimedia cloud computing, their scheme provides dynamic resource allocation with reduced cost. Authors have compared their algorithm with greedy approach of migration in case of overloaded VMs and indicated better results. However there is no bargaining for resource utilization. Marrone and Nardone (2015) proposed a model driven approach for resource allocation.

Authors have deployed an automatic negotiation model using UML and Bayesian Network modeling approach. This works is completely based on negotiations. However, response time and cost optimization have been left unattended. Xiao et al. (2013) has introduced concept of skewness to measure unevenness in multi-dimensional resource utilization of a server. Different types of workloads can be combined to minimize skewness and improve overall server resource utilization. This mechanism provides overload avoidance while concerning green computing. Yee-Ming and Hsin-mie (2010) provided an allocation and pricing mechanism as a market-based model for allocating resources in a cloud computing environment. But this model is also not able to handle large scale problems adequately. Many more resource allocation mechanisms are available in Buyya et al. (2008), Jung and Sim (2011), Stoesser et al. (2007), Streitberger et al. (2007) which reflects that substantial efforts had already been put toward resource management in cloud computing but to the best of our knowledge none has proved to be suitable under all conditions. From the literature review it is clear the main purposes of scheduling algorithms are to minimize the resource starvation and to ensure the effective and fair resource scheduling (Singh and Malhotra, 2013). In fact, optimal resource allocation strategies have always been of utmost concern for researchers and hence the need to pay more attention on the resource scheduling policies is chirping in. Traditionally optimal resource allocation makes use of the Hungarian algorithm, which can work only on symmetric number of resources and requests. But, cost sharing model of CC deploys multi-tenancy, thus resource scheduling cannot be optimized using the Hungarian method always. This gave us motivation for the present work which focuses on an intelligent agent-based automated scheduling and service composition framework for cost optimization of resource provisioning in cloud computing. Next section elaborates the proposed framework.

III. PROPOSED INTELLIGENT AGENT BASED FRAMEWORK

From the literature review it is clear that limited work has been done for load balancing in cloud computing environment and those existing mechanisms do have limitations that need to be addressed. Thus there is need of an algorithm which can offer maximum resource utilization, maximum throughput, minimum response time, dynamic resource scheduling with scalability and reliability. This work proposes an autonomous agent based load balancing algorithm (A2LB) to address above issues. Whenever a VM becomes overloaded, the service provider has to distribute the resources in such a manner that the available resources will be utilized in a proper manner and load at all the virtual machines will remain balanced. A2LB mechanism comprises of three agents: Load agent, Channel Agent and Migration Agent. Load and channel agents are static agents whereas migration agent is an ant, which is a special category of mobile agents. The reason behind deploying ants is their ability to choose shortest/best path to their destination. Ant agents are motivated from biological ants which seek a path from their colonies to the food source. While doing so they secrete a chemical called pheromone on ground [16] thus leaving a trail for other colleagues to follow. However this chemical evaporates with time. Initially the ants start searching a food source randomly, thus they may follow different paths to the same source, however with passage of time, density of pheromone on the shortest path increase and thus all follower ants start following that path resulting in increase of pheromone density even further. An appealing property of ants is that they move from source to destination for collecting desired information or performing a task but they do not necessarily come back to their source rather they destroy themselves at the destination only thereby reducing unnecessary traffic on the network. Since load balancing in CC would require searching for under loaded servers and resources, ant agents suit the purpose and fulfill it appropriately without putting additional burden on network.

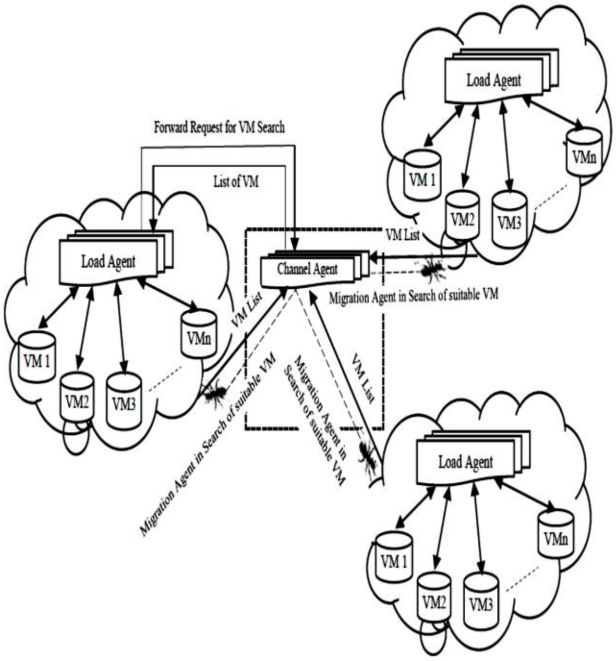
Forward Request for VM Search
List of VM
Load Agent
Load Agent
Load Agent
Channel Agent
VM List
VM List
VM List
Migration Agent in Search of suitable VM
Migration Agent in Search of suitable VM
Migration Agent in Search of suitable VM
VM 1
VM2
VM3
VMn
VM 1
VM2
VM3
VMn
VM 1
VM2
VM3
VMn

Figure 2 Working of proposed system

Description of various agents deployed in A2LB is as follows:

- Load Agent (LA):It controls information policy and maintains all detail of a data centre. The major work of a load agent is to calculate the load on every available virtual machine after allocation of a new job in the data centre. This agent is supported with table termed as VM_Load_Fitness table.
- VM_Load_Fitness table: It is used for maintaining record of specifications of all virtual machines of a data centre. It contains virtual machine id, status of its memory consumed along with CPU utilization, fitness value and load status of all VMs..

IV. IMPLEMENTATION AND RESULTS

In order to illustrate the performance of the agent based resource monitoring system, a system with hardware configuration of CPU with 4 x 2.44 GHz, Memory (RAM) with 3.41 GB and local disk with 82 GB, installed with Linux 2.6.1.8-238.el5Xen (X86_64) was considered. Our agent is implemented using java programming. In future, we are planning to use mobile agents for resource monitoring for proper resource utilization. Eucalyptus was used to create virtual machines with a centos image customized with java and tomcat server. On top of each VM, CPU intensive application and memory intensive application such factorial calculation and finding paths in a graph were considered to test the performances of virtual machines. The CPU and memory utilization factors were discussed in table 1 using EC2 image of VM type small (1 CPU, 128 MB RAM and 3GB hard disk).
Table 1

V. CONCLUSION

Cloud providers must ensure that end user receives the services with reliable and optimal business experience. It is essential to monitor the actual resource usages to avoid over and under estimation of resource levels. So, in this paper, architecture is presented for resource monitoring using agent based system. Since, agents can be loaded anywhere it is easy to use in a real cloud environment for monitoring purposes. As part of our future work, we would like to work on mobile agents that take automatic decisions based on resource usage statistics so that it is beneficiary to both the Cloud Provider and Cloud Consumer.

REFERENCE

[1] Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., et al.: A view of cloud computing. Commun. ACM 53(4), 50–58 (2010)

[2] Beloglazov, A., Buyya, R.: Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. Concurr. Comput. 24(13), 1397–1420 (2012)

[3] Bonvin,N., Papaioannou,T.G.,Aberer, K.:Autonomic SLA-driven provisioning for cloud applications. In: Proceedings of the 2011 11th IEEE/ACM international symposium on cluster, cloud and grid computing, IEEE Computer Society, pp. 434–443 (2011)

[4] Calheiros, R.N.,Ranjan, R.,Beloglazov, A.,DeRose, C.A.,Buyya, R.: Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning

algorithms. Software 41(1), 23–50 (2011)

[5] Chatterjee, S., Hadi, A.S.: Regression analysis by example.Wiley, Hoboken (2013)

[6] Duong, T.N.B., Li, X., Goh, R.S.M.: A framework for dynamic resource provisioning and adaptation in iaas clouds. In: Proceedings of the IEEE Third International Conference on Cloud Computing Technology and Science (CloudCom), 2011, pp. 312–319 (2011). IEEE

[7] Hategan, M., Wozniak, J., Maheshwari, K.: Coasters: uniform resource provisioning and access for clouds and grids. In: Proceedings of the Fourth IEEE International Conference on Utility and Cloud Computing (UCC), pp. 114–121 (2011). IEEE

[8] Herbst, N.R., Kounev, S., Reussner, R.: Elasticity in cloud computing: what it is, and what it is not. In: Proceedings of the 10th International Conference on autonomic computing (ICAC 2013), San Jose, CA (2013)

[9] Huang, H., Wang, L.: P&p: a combined push-pull model for resource monitoring in cloud computing environment. In: Proceedings of the Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on, pp. 260–267 (2010). IEEE

[10] Jararweh,Y., Jarrah, M.,Kharbutli, M.,Alsaleh,M.N., Al-Ayyoub, M.: CloudExp: a comprehensive cloud computing experimental framework. Simul. Model. Pract. Theory 49, 180–192 (2014)

[11] Marshall, P., Keahey, K., Freeman, T.: Elastic site: using clouds to elastically extend site resources. In: Proceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, EEE Computer Society, pp. 43–52. I (2010)

[12] Siddiqui, U., Tahir, G.A., Rehman, A.U., Ali, Z., Rasool, R.U., Bloodsworth, P.: Elastic jade: dynamically scalable multi agents using cloud resources. In: Proceedings of the Cloud and Green Computing (CGC), 2012 Second International Conference on, pp. 167–172 (2012). IEEE

[13] 13. Vaquero, L.M., Rodero-Merino, L., Buyya, R.: Dynamically scaling applications in the cloud.ACMSIGCOMMComput.Commun. Rev. 41(1), 45–52 (2011).