International Journal of Engineering, Economics and Management
ISSN: 2319-7927, Volume 3, Issue 1

Divisible Resource Allocation to Minimize Cloud Task Length

Sneha S. Tile Prof. Naresh Thoutam

Abstract — **Cloud computing is a model which is used basically to get share access to the divisible resources .The user should get convenient access to data in the cloud . Cloud computing environment involves high cost infrastructure on one hand and need high scale computational resources on the other hand. These resources need to be provisioned (allocation and scheduling) to the end users in most efficient manner so that the tremendous capabilities of cloud are utilized effectively and efficiently. But there are some constraints on cloud computing like payment budget and on demand resources. The analysis of the proposed model gives the following 3 important points-1. It derives the upper bound of cloud task length, by taking into account both workload prediction errors and host load prediction errors. 2) Designing of a dynamic version for the algorithm to adapt to the load dynamics over task execution progress, further improving the resource utilization. 3) Build a cloud prototype over a real cluster environment with 56 virtual machines, and evaluate our algorithm with different levels of resource contention**.

Key Words — **ODRA Algorithm, divisible-resource allocation, convex optimization, upper bound analysis, LOAA Algorithm, web composition.**

I. INTRODUCTION

Cloud Computing uses SOA (Service Oriented Architecture) to provide IaaS (Infrastructure as a Service) [1],[2],[3], SaaS (Software as a Service) [1],[2],[3], PaaS (Platform as a Service) [1],[3], DaaS (Data Storage as a Service) [1][2], CaaS (Communication as a Service) [2],[3], HaaS (Hardware as a Service) [1],[2] to cloud users. The end users can use these resources over a network on-demand basis in pay-as-you-say manner.

Cloud computing is an attracting technology in the field of computer science. In Gartner's report, it says that the cloud will bring changes to the IT industry. Traditional task scheduling adopted in distributed systems like grids assumes discrete resource usage model. The processing ability assigned to a task cannot be customized by users elastically. Such an indivisible resource consumption model with discrete computation unit results in a non-trivial problem like binary Integer programming problem, where CPU rates may not be fully utilized. With virtual machine (VM) resource isolation technology the computational resources could be partitioned and reassembled on demand, creating an avenue to improve resource utilization. There is an optimal algorithm (namely local optimal allocation algorithm (LOAA)) [12] minimizing a task's execution length, subject to a set of constraints like user's payment budget and host availability states.

Virtual machine (VM) technology being greater and fully developed, compute resources in cloud systems can be partitioned in fine granularity and allocated on demand, which contributes three technologies such as, Formulating a deadline-driven resource allocation problem based on the cloud environment facilitated with VM resource isolation technology, and also to minimize users' payment. Analyzing the upper bound of task execution length based on the possibly inaccurate workload prediction, it further proposed an error-tolerant method to guarantee task's completion within its deadline. Validating its effectiveness over a real VM-facilitated cluster environment under different levels of competition.Traditional task scheduling adopted in distributed systems like grids assumes discrete resource usage model [1], [2], [3]. The processing ability assigned to a task cannot be customized by users elastically. Such an indivisible resource consumption model with discrete computation units results in a non-trivial problem like binary Integer programming problem, where CPU rates may not be fully utilized.
In general, services can be classified into two categories: a non-delay system (loss system) and a waiting system. A non-delay system allocates a spare resource immediately to the user upon the arrival of the request, and rejects the request if there is no spare capacity. A waiting system allocates a spare capacity to users in the sequence in which their requests have arrived, instead of allocating resources immediately upon the arrival of a request. This paper assumes a service that runs as non-delay. This paper also assumes static resource allocation, which is the most basic form of resource allocation, although dynamic allocation, which uses process migration and bandwidth consolidation, can increase the utilization of resources.

The paper is organize into four phases they are as follows.
1. In first step formulation of the cloud resource allocation issues a convex optimization problem aiming to minimize task length with divisible resource fractions and a set of constraints is performed
2.In second step, LOAA algorithm is used to outline the task processing procedure and .then prove that LOAA algorithm can also minimize user payments meanwhile based on tasks' final real wall-clock lengths.
3. In the third step the upper bound of task execution length considering prediction errors on task workloads and resource availability, as against to the result under the hypothetically precise prediction is derived.
4. In the fourth step, the algorithm is expanded to adapt the volatile states of the system with multiple web services deployed. The experimental results generated over a real-cluster environment.

Traditional task scheduling adopted in distributed systems
like grids assumes discrete resource usage model [1], [2], [3]. The processing ability assigned to a task cannot be customized by users elastically. Such an indivisible resource consumption model with discrete computation units results in a non-trivial problem like binary Integer programming problem, where CPU rates may not be fully utilized.
With virtual machine (VM) resource isolation technology

[4], [5], [6], [7], [8], [9], the computational resources could be partitioned and reassembled on demand, creating an avenue to improve resource utilization. In our previous work [10], we proposed an optimal algorithm (namely local optimal allocation algorithm (LOAA)) minimizing a task's execution length, subject to a set of constraints like user's payment budget and host availability states.

II. Literature Survey

Many related works have been done to achieve efficient resource allocation scheme. Resource prov sioning in cloud computing environment is done with the main aim of achieving load balancing. Based on various factors like spatial distribution of cloud nodes, algorithm complexity, storage/replication, point of failure etc. different techniques have evolved to provision the resources in balanced manner. The provisioning is done taking into account whether the environment is static or dynamic.

In addition, this paper analyze the approximation ratio for the expanded execution time generated by the algorithm to the user-expected deadline, under the possibly inaccurate task property prediction. When the resources provisioned are relatively sufficient, It can guarantee task's execution time[11] always within its deadline even under the wrong prediction about task's workload characteristic.

1) It formulate a deadline-driven resource allocation problem based on the cloud environment facilitated with VM resource isolation technology, and also proposed a novel solution with polynomial time, which could minimize users' payment in terms of their expected deadlines.
2) By analyzing the upper bound of task execution length based on the possibly inaccurate workload prediction, it further proposed an error-tolerant method to guarantee task's completion within its deadline.
3) It validate its effectiveness over a real VM-facilitated cluster environment under different levels of competition.

Resource provisioning in cloud computing environment is done with the main aim of achieving load balancing. Based on various factors like spatial distribution of cloud nodes, algorithm complexity, storage/replication, point of failure etc. different techniques have evolved to provision the resources in balanced manner. The provisioning is done taking into account whether the environment is static or dynamic. Many related works have been done to achieve efficient resource allocation scheme. Statistical based Resource allocation (SLB) introduced by Zhenzhong Zhang in paper [4] is based on prior learning and performance statistics, SLB involves analysis of huge on-line historical data for forecasting resource demands. In paper [5], an algorithm given for load balancing is an inspiration from the honeybee. It is biologically inspired technique that uses behavior of honeybees foraging and harvesting for food. It does not take into consideration the waiting time of the tasks and overall turnaround time. Similar techniques mentioned in paper [6], [7] are used for load balancing but none reduce the overall waiting time of tasks.

Optimal divisible resource allocation(ODRA)
This method focuses on how to make full use of the multi-attribute resources facilitated by such a resource isolation technology to optimize the task's execution efficiency, under users' specific resource demands (such as execution payment and service level). Another challenge about the resource allocation issue is the potentially high-dimensional execution. Since task's workloads as well as computational resources are multi-attribute, the execution will be multi-dimensional in nature. Even through considering only one resource attribute (for example, the task may be computation-intensive application), a task may also be split to multiple sequential execution steps (or phases), each calling for a different resource capacity and price on demand. When user request for files in any server the processor first checks the available resources with request made by the user if it is not feasible then the resources are divided and accommodated to all the requests made by the user. So it will improve the time consumption by the resources and delay in the access to the data.

Implementation of Local optimal allocation algorithm (**LOAA**))
LOAA algorithm is an optimal solution to minimize task length, but it can prove that user payment is also minimized based on task's final wall-clock length.

Minimize the upper bound of task execution length
It is used when task's workload and host's availability will be predicted with errors, which is in the line with reality. For instance, the multi-variant polynomial regression method and Bayes method have been effective in precise workload prediction and host load prediction respectively, yet they are still suffering inevitable margin of prediction errors like 10 percent. On the other hand, the flexible resource partitioning of the cloud systems may definitely result in load dynamics on resource states, and worse still, the collected states are error-prone due to the network propagation delay. The inevitable load prediction errors may significantly affect task's execution in reality. It derive the bound of task length for the LOAA algorithm[4], based on erroneous prediction of task's workload and resource availability, as compared to the theoretically optimal task length with hypothetically accurate information. This is fairly valuable/useful in that users are able to know the worst performance in advance and the resource allocation can be tuned in turn to adapt to user demand based on the bound of task execution length estimated.

Dynamic version for load balancing
The algorithm can further extended to a dynamic version[1],[2] to adapt to the load dynamics over time. Due to the dependency between the subtasks (or web services) of a task, the resource availability states for a particular subtask may not be forecasted upon the task's initial submission. Accordingly, we extend our algorithm to be a dynamic (or adaptive) version, which can tune the resource allocation at runtime based on task's execution progress and updated resource availability states.

A large portion of the work in resource allocation in cloud computing mainly focused on the cost-effectiveness and easy maintenance of the systems [1]. Most of the work has been descriptive in nature. Patricia et al. [2] discusses this process in the context of distributed clouds, which are seen as systems where application developers can selectively lease geographically distributed resources. [2] Highlights and categorizes the main challenges inherent to the resource allocation process particular to distributed clouds, offering a stepwise view of this process that covers the initial modelling phase through to the optimization phase.

A critical evaluation of current network resource allocation strategies and their possible applicability in Cloud Computing Environment which is expected to gain a prominent profile in the Future Internet are presented in work by M. Asad Arfeen [3]. Atsuo Inomata et al. [4] has proposed a dynamic resource allocation method based on the load of VMs on IaaS, abbreviated as DAIaS. This method enables users to dynamically add and/or delete one or more instances on the basis of the load and the conditions specified by the user. It has been believed that a market-based resource allocation will be effective in a cloud computing environment where resources are virtualized and delivered to users as services (Fujiwara et al. [5]) and in such a market mechanism to allocate services to participants efficiently has proposed.

The mechanism enables users to order a combination of services for workflows and coallocations and to reserve future/current services in a forward/spot market. The evaluation shows that the mechanism works well in probable setting. In a cloud computing environment, it is necessary to simultaneously allocate both processing ability and network bandwidth needed to access it. Tomita et al [6] proposed the congestion control method for a cloud computing environment which reduces the size of required resource for congested resource type, instead of restricting all service requests as in the existing networks.

Mochizuki and Kuribayashi [7] presents cloud resource allocation guidelines in the case where there is a limit to electric power capacity available in each area, assuming a cloud computing environment in which both processing ability and network bandwidth are allocated simultaneously. Next, it proposes a method for optimally allocating processing ability and bandwidth as well as electric power capacity. Optimal allocation means that the number of requests that can be processed is maximized, and the power consumed by a request is minimized. It is demonstrated by simulation evaluations that the proposed method is effective.

CONCLUSION

This paper optimize the divisible-resource allocation with in-depth analysis on upper bound of task execution length under prediction errors. It also design a dynamic approach that adapts to the load dynamics over task execution progress. This paper evaluate the performance using a real cluster environment with composite web services. These services are of different execution patterns on multiple types of resources. Experiments show that task execution lengths with our ODRA solution are always close to their theoretically optimal results with resource capacity limitation. S

REFERENCES

[1] Sheng Di, , Cho-Li Wang and Franck Cappello, IEEE "Adaptive Algorithm for Minimizing Cloud Task Length with Prediction Errors" IEEE Trans. Parallel and Distributed Systems, vol. 24, no. 6, pp. 1097-1106,April-June 2014

[2] D. Sheng, D. Kondo, and W. Cirne, "Host Load Prediction in a Google Compute Cloud with a Bayesian Model," Proc. IEEE/ACM 24th Int'l Conf. for High Performance Computing, Networking, Storage and Analysis (SC '12), pp. 21:1-21:11, 2013.

[3] S. Di and C.-L. Wang, "Dynamic Optimization of Multi-Attribute Resource Allocation in Self-Organizing Clouds," IEEE Trans. Parallel and Distributed Systems, vol. 24, no. 3, Mar. 2013.

[4] S. Di and C-L. Wang, "Minimization of Cloud Task Execution Length with Workload Prediction Errors," Proc. 20th High Performance Computing Conf. (HiPC '13), 2013.

[5] Gideon-II Cluster: http://i.cs.hku.hk/_clwang/Gideon-II, 2012.

[6] F. Chang, J. Ren, and R. Viswanathan, "Optimal Resource Allocation in Clouds," Proc. Third IEEE Int'l Conf. Cloud Computing (Cloud '10), pp. 418-425, 2010.

[7] C. Jiang, C. Wang, X. Liu, and Y. Zhao, "A Survey of Job Scheduling in Grids," Proc. Joint Ninth Asia-Pacific Web and Eighth Int'l Conf. Web-Age Information Management Conf. Advances in Data and Web Management (APWeb/WAIM '07), pp. 419-427, 2007.

[8] D. Gupta, L. Cherkasova, R. Gardner, and A. Vahdat, "Enforcing Performance Isolation
Across Virtual Machines in Xen," Proc. Seventh ACM/IFIP/USENIX Int'l Conf. Middleware (Middleware '06), pp. 342-362, 2007.

[9] S. Chinni and R. Hiremane, "Virtual Machine Device Queues," techical report, Virtualization Technology White Paper, 2007.

[10] D. Gupta, L. Cherkasova, R. Gardner, and A. Vahdat, "Enforcing Performance Isolation
Across Virtual Machines in Xen," Proc. Seventh ACM/IFIP/USENIX Int'l Conf. Middleware (Middleware '06), pp. 342-362, 2006.

[11] S. Chinni and R. Hiremane, "Virtual Machine Device Queues," techical report, Virtualization Technology White Paper, 2006.

[12] D. Gupta, L. Cherkasova, R. Gardner, and A. Vahdat, "Enforcing Performance Isolation
Across Virtual Machines in Xen," Proc. Seventh ACM/IFIP/USENIX Int'l Conf. Middleware (Middleware '06), pp. 342-362, 2006.

[13] J.E. Smith and R. Nair, Virtual Machines: Versatile Platforms for Systems and Processes. Morgan Kaufmann, 2005.

[14] Website http://www.net-security.org/secworld.php?id=10886, Article on "Lack of admin rights mitigates most Microsoft vulnerabilities" Posted on 12 April 2011.

[15] Patricia Takako Endo, Andre Vitor de Almeida Palhares, Nadilma Nunes Pereira, 2011. Resource Allocation for Distributed Cloud: Concepts and Research Challenges, IEEE, july 2011.

[16] Hadi Goudarzi and Massoud Pedram University of Southern California, MaximizingProfit in Cloud Computing System via Resource Allocation.

[17] M.Asad Arfeen, Krzysztof Pawlikowski, Andreas Willig .2011,
A Framework for Resource Allocation Strategies in Cloud Computing Environment, 2011 35th IEEE Annual Computer Software and Applications Conference Workshops.

[18] Atsuo Inomata, Taiki Morikawa, Minoru Ikebe. 2011, Proposal
and Evaluation of a Dynamic Resource Allocation Method based on the Load of VMs on IaaS, IEEE 2011.