Provenance based Adaptive Heuristic Scheduling in Cloud Environment

I.M.Maywish Rajakumari, Mrs.R.Narayani

Abstract— Cloud computing can be partitioned as a fine granularity and pool of resources to serve multiple consumers. The on-demand delivery of cloud computing is mainly based on internet services. The remote users can access the cloud systems by IaaS service. It provides the full infrastructure of the cloud environment. The multiuser parallel jobs are allocated to the host is really a challenging task. So, the flow of jobs can be handled by adaptive heuristic scheduling process. It helps to allocate the consumer jobs to the most appropriate fitted servers. In this paper, our system proposes to create a single datacenter with five servers which have static memory size. In adaptive heuristic scheduling processing, the parallel jobs have to be allocated to the servers. The frequent of server usage reductions are controlled using flag values. Then the system tracks all the information about the servers and its allocation process and stored it in a separate database. This database is referred as a provenance database. Finally a new jobs comes, it directly allocated to the fitted server with the reference of provenance database. This system helps to avoid the frequent allocation of particular server and also minimize the total execution time.

Index Terms— adaptive heuristic scheduling , Cloud computing, data tracking, fitted server, flag values, frequent allocation, provenance

1 INTRODUCTION

THE distributed computing is considered as the business model for cloud computing. It handles the next generation of computing as a storage platform. The basic characteristic of cloud computing are virtualization, agility, resource pooling and dynamic extensibility. The resources can be virtualized and managed on a cloud computing platform includes software, hardware, operating system and network storage. There are three main services delivers the platform to the client such as IaaS, PaaS and SaaS. Many IT industries merely utilizes the wide range of facilities provided by the cloud computing, so that the number of services are easily increase or decrease as per user requirements. These issues in cloud computing still hinges up fully understanding and managing the challenges that the public concerns, for instance, confidentiality and privacy problems. The multiple jobs are coming from the consumers are one of the complex task in cloud environment. Effective scheduling algorithms are required for us to make effective use of parallel job allocations in tremendous capabilities of the cloud. There is a need to optimizing the jobs to be dispatched.

Scheduling algorithm try to minimize the total execution time of the jobs in the cloud by finding the most suitable server to be allocated. The frequency of server usage can be reduced by fixing the flag values, it triggers the server as an idle position once the flag value becomes zero. The overall information about the server and its allocation process are stored in a separate database. The reference of database is considered as a provenance database.

Oxford defines provenance as “the place of origin or earliest known history of something”. The important aspect of provenance based on verification, audit trails, reproducibility, privacy and security, trust and reliability. There are many fields such as art, science and real provenance is considered as the class data for tracing an object from its origin. The properties of provenance are about the process, time and input and evaluated data. Provenance is aim to deliver the answers for few questions such as when the data was created, what is the content of data, purpose of creation and who is the authorized person for the data. Provenance is also important for data forensics to provide digital evidence for post investigation.

In cloud computing, provenance is used to collect, evaluate, verify store the sequence of metadata. These metadata contains the functional and non-functional data required to track the data from the beginning and review its result. The word lineage is sometimes called as a provenance data. In the lineage, the metadata is detailing of a derivation about the objects of data.

In this paper, our system explores the elasticity characteristic for optimal allocation of parallel jobs in cloud environment. Differently from the users of grid and cluster, cloud users beneficially from the elasticity concept. Due to the processing demands, the resources are easily scale up and scale down virtually. The multiuser parallel jobs are scheduling by adaptive heuristic scheduling. The information about the servers and its placements of jobs is maintained in a separate database. This database is referred as provenance database and it helps to analyze the data from previous executions. The main contribution of this system is to minimize the total execution time and finding the frequent usage of servers.

- *I.M.Maywish Rajakumari is currently pursuing master degree program in B S Abdur Rahman University, India, PH-7708519705. E-mail: jmaywish@gmail.com*
- *Mrs.R.Narayani working as an assistant professor in B S Abdur Rahman University, India, PH-9884709306. E-mail: rvnarayani@gmail.com.*

2 RELATED WORK

In recent years, there is growing interest in developing the tracking mechanism for server usage consolidation. The problem of parallel job scheduling is derived using adaptive heuristic approach. It is based on the scheduling and provenance model. In the scheduling model, the parallel jobs are allocated to the servers without any collision occurs. The server usage is recduced by flag concept. The provenance model references the new job to be place in server with a short completion of time. Our system has structured the related work section in two parts as an existing scheduling systems and the provenance usage.

Pardeep Kumar and Amender Verma, et al [6] focused some works on improved genetic scheduling algorithm. In general, the genetic algorithm was used to merge the Min-Max and Max-Min approach and improved the genetic algorithm. It is able to schedule the multiple jobs in multiple machines according to user requirements with the minimum completion of time. The improved genetic algorithm is another part of this existing system. The initial population is randomly generated and mutated with each other. There are very less chance to produce a better child than themselves. Using Min-Min and Max-Min, it provides an idea for generating initial population. The genetic operators tend to give the solution that are fit and better generation and hence produce the individual generations by Min-Min and Max-Min. This technique is used to derive a better scheduling of tasks to resources. So, the user tasks can be completed in a minimum time. However no work was found that consider the frequent server usage.

Chun Hui Suen and Ryan K L Ko, et al [2] Provenance model recently gained attention to maintain the previous execution results in an efficient way. The stake holders felt difficult to track the data from the concern cloud computing environment. The lack of data tracking tool built for cloud. There is a required to logging mechanism data centric technique to track data activities within all cloud services. The data movement in cloud will be enable for transparency and accountability The S2 logger is used as the data logging mechanism which could capture, analyses and virtualize the data events in cloud. The resulting sequence of data events referred as the cloud provenance records. This information helps to overcome the problem like malicious action.

Jinhai Wang and Chuanhe Huang et al [5] investigated energy consumption has become a big concern the deployment of cloud datacenters. There are many techniques helps to reduce energy consumption for cloud data center that consist of a large number of identical se the servers memory is a dominant factors of the performance and energy consumption of cloud system. The existing system proposed a heuristic greedy algorithm to implement virtual machine deployment and live migration to increase total resource utilization of the servers, which is based on energy-aware predict the workload. The heuristic makes the CPU services and mapped to the same physical server. These experiment result shows that there is an important in aspect of energy saving, workload balancing and scalability.

3 SYSTEM DESIGN

The allocation jobs to the most fitted server using the provenance database.

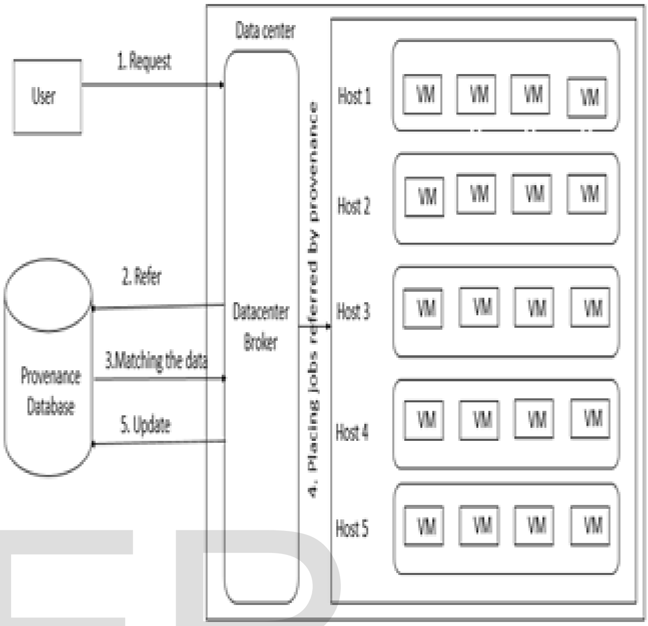

Fig. 1. System architecture of provenance based scheduling

Fig1 is the outline process involved in here is the analysis of the data sent by the user and allocating them to the desired resource locations without any collision and also reducing the usage of the server. Firstly the user sends the data request to the datacenter broker. Then, the datacenter broker does all the coordinating work and schedules the job to the desired server by using by using adaptive heuristic scheduling. Frequent use of the same server can be idle after some task by applying flag concept. Then this information is saved in the provenance database and it is updated for each process. When the datacenter broker get to know a new job which is similar to the one stored on the provenance database, it does the same procedure of the previously executed relevant data and thereby reducing the total execution time.

4 PROPOSED STRATEGY

As well known, the scheduling of parallel job is really a challenging task, especially, it is very intractable to find the optimal solution for allocation of multiple jobs in complex cloud environment. Our system finding the optimal adaptive heuristic scheduling through the following job allocation using adaptive heuristic scheduling, frequent server usage reduction process and allocation of jobs referred by provenance data.

4.1 Job Allocation Using Adaptive Heuristic Scheduling

The adaptive heuristic algorithm is a method of scheduling in which the jobs are assigned to the servers according to individual solutions. It tells about which server is most optimal way to adapt the consumer jobs. Algorithm is slightly based on the first come first serve for server selection process. The jobs are processed in a queue and then schedule one by one to the most suitable prior free server. Initialize the datacenter and physical machines (servers) which have a static memory size. The user sends the parallel request to the cloud environment. Once the datacenter broker analyzes the job and allocates the job to the first free server. The datacenter broker schedules the job to the most adaptive server. Then the jobs are placed and instance the server as many virtual machines using virtualization concept. The server and its job allocation details are tracked and stored it in a separate database.

Algorithm 1: job allocation using adaptive heuristic scheduling

```
INPUT: user task, time of task
OUTPUT: allocate the job in suitable server
Begin
Initialize the physical machine size, datacenter and time slot for each server as static
Evaluate each user jobs based on scheduling mechanism
Schedule the job with adaptive heuristic scheduling
if flag = 1
server has enough resource to allocate job then
server becomes active
allocate job → server
else
flag = 0
server has frequently used then
server becomes idle
allocate job → next server
store the retrieved server information
maintain the records as provenance datasets
check if new job exists then
referred the provenance
check if similar job resides in provenance then
allocate jobs directly
End
```

The above algorithm proceeds under some static conditions. Firstly, our system should create a datacenter, partition the physical machines with equal memory sizes and fix a time slot for each server. The multiple user jobs are allocate using adaptive scheduling algorithm with the aspect of first come first serve mechanism. The user jobs should have a time slot for each job while entering into the datacenter. A set of jobs are get into the cloud, the datacenter broker analyze the job size and its time slot. The jobs starts allocate the first job in a first free server. There are no resources to allocate in first server means its move on to next server.

Likewise all jobs are scheduled and allocated according to the user specifications. In that way of proceeding, the frequent usage of server is reduced by utilization detection using flag concept. Our system proposed a heuristic for deciding the time to migrate jobs from a server based on frequent utilization. It is based on the idea of setting upper and lower frequent utilization server and keeping the total utilization of the memory. If the server utilized frequently then of a host all jobs have to be migrated from this server and the server has to be switched to the sleep mode in order to eliminate the idle reduction of frequent usage. Once the job allocated to the server the time slot starts decremented. When the time slot becomes zero, the flag has to be set as '0', the server switch to idle mode for short period of time. In that period of time, the idle server should not be allocated for any job. After a few time slots, the server again switches on to active mode and starts allocating jobs.

The records of all the information about the server and its utilization requirements are keep tracked from the origin. The retrieved information are stored and maintained as a provenance dataset. It is described clearly about that job and its related records. When the datacenter broker get to know a new job, it starts referred the storage one has already resides in the provenance datasets. Now the jobs are allocated directly with the reference of provenance datasets.

4.2 Frequent Server Usage Reduction Process

According to the adaptive heuristic scheduling, the optimal server tends to allocate the frequent job to the most adaptive server. The purpose of avoiding the frequent usage of server can be dealt with flag concept. The timer is fixed for each server, once the jobs get into the server, it starts decremented in timer. Likewise, the timer reaches the value zero referred as "flag 0". When the server is considered as idle server, switch them into sleep mode or standby mode till the timer value is changed as "flag 1" as active server. Then it again starts allocating a job to the server. This system helps to find out the frequency of server usage and tends to reduce the servers utilization in the datacenter.

4.3 Allocation of Jobs Referred by Provenance Database

Provenance is a type of metadata to track the information about the data from its origin and transformed it. These can be used to protect the intellectual property and its relevance from the perspective of regulatory mechanism. Provenance reduces the growing number of dataset in the computer storage, a better information retrieval quality is needed in order to increase its quality, integrity, confidentiality and authenticity. A detailed history of information will be used to perform the reference and auditing activities. The provenance collection is really a challenging in cloud environment because it is difficult to collect and is often incomplete.

In our proposed system, provenance is used to track the record of job allocation and its partitions of memory in server as a virtual machine according to the jobs arrival from the beginning process. After collecting the set of records and maintain as a separate provenance datasets. When the datacenter broker get to know a new job which is similar to the job one has already stored in the provenance datasets. The provenance logs is not only showing the result with a certain job but also linking all together log records and complete history of information for that particular job. The new job is allocated to the server directly with a reference by provenance datasets. These system leads to minimize and reduce the overall total execution of time.

5 CONCLUSION

This paper presents an efficient solution for allocating parallel jobs by handling adaptive heuristic scheduling. With our proposed system, it can make the allocation of multiple jobs without collision. By using flag concept the active server have a better consolidation and performance improving and the fragments of active servers can be used sufficiently much more and reduce the frequent usage of server to switch server into an idle mode. The provenance datasets is used as a reference for new jobs. The storage of provenance has to support both the storage requirements of job and the database functionality requirements of provenance. The jobs are directly allocated to the server referred by provenance datasets. The experiment result shows that the reduction of frequent server usage and minimize the total execution of time.

REFERENCES

[1] Daniel de Oliveira and Fernanda Baioa, "A Provenance-based Adaptive Scheduling Heuristic for Parallel Scientific Workflows in Clouds", J Grid Computing, vol.10, pp.521–552, 2012.

[2] Chun Hui Suen and Ryan Ko, "S2logger: end-to-end data tracking mechanism for cloud data provenance" Proceedings of the 12th IEEE International Journal, pp.594-604, 2013.

[3] Madhusree Kuanr, Prithviraj Mohanty and Suresh Chandra Moharana, "Grouping-Based Job Scheduling in Cloud computing using Ant Colony Framework", International Journal of Engineering Research and Applications (IJERA), 2013.

[4] S. Mukute and G. Hapanyengwi, "Scheduling in instance-intensive cost-constrained workflows in a cloud", International Journal of Scientific & Engineering Research, vol.4, pp.755- 789, 2013.

[5] Jinhai Wang and Chuanhe Huang, "An Energy-awared Resource Allocation Heuristics for VM Scheduling in Cloud", IEEE International Conference on High Performance Computing and Communications, vol.10, 2013.

[6] Pardeep Kumar and Amandeep Verma, "Independent Task Scheduling in Cloud Computing by Improved Genetic Algorithm", vol.2, 2012.

[7] Shamsollah Ghanbari and Mohamed Othman, "A priority based job scheduling algorithm in cloud computing", International Conference on Advances Science and Contemporary Engineering , vol.50, pp.778 – 785, 2012.

[8] Sen Su, Jian Li, Qingjia Huang, Xiao Huang, Kai Shuang and Jie Wang, "Cost-efficient task scheduling for executing large programs in the cloud", parallel computing, vol.39, pp.177-188, 2013.

[9] Musajjul Ali and Luc Moraen, "A provenance –aware policy language (cprovl) and a data traceability model (cprov) for the cloud", IEEE Third International Conference on Cloud and Green Computing, vol.13, 2013.

[10] Muhammad Imran and Helmut Hlavacs, "Application of provenance data for cloud infrastructure", Eighth International Conference on Semantics, Knowledge and Grids, vol.12, 2013.

[11] Mohd Izuan Mohd Saa and Kamarularifin Abd Jalil, "Data provenance trusted model in cloud computing", International Conference on Research and Innovation in Information Systems (ICRIIS'13), 2013.

[12] Chunyan Zhao and Enzhao Hu, "Job scheduling algorithm based on Berger model in cloud environment", advances in engineering software, vol.42, pp.419-425, 2011.

[13] Indhu Ganrotra, Pawanesh Abrol and Pooja Gupta, "Cloud computing over cluster, grid computing: a comparative analysis", International Journal of Advanced Research in Computer and Communication Engineering, vol.3, 2014.

[14] R.Gogulan, A.Kavitha and U.Karthick Kumar, "A multiple pheromone algorithm for cloud scheduling with various Qos requirements", IJCSI International Journal of Computer Science Issues, vol.9, Issue 3, No 1, May 2012.