Smart Task Assignment Model for Cloud Service Provider

Mainak Adhikari
Department of Computer Science and Engineering
University of Kalyani
Kalyani, Nadia, West Bengal, 741235, India

Sourav Banerjee
Department of Computer Science and Engineering
K.G.E.C
Kalyani,Nadia,West Bengal, 741235, India

Utpal Biswas
Department of Computer Science and Engineering
University of Kalyani
Kalyani, Nadia, West Bengal, 741235, India

ABSTRACT

Cloud computing is one of the upcoming latest technology which has been developing drastically. Today lots of business organizations and educational institutions using Cloud environment. But one of the most important thing is to increase the Quality of Service (QoS) of the system. The cloud environment is divided into two parts mainly, one is Cloud User (CU) and another is Cloud Service Provider (CSP). CU sends service requests to the CSP and all the requests are stored in a Request Queue (RQ) inside CSP which directly communicates with Smart Job Scheduler (SJS). SJS communicates with Resource Pool (RP) and tries to assign each of these jobs as per there requirement to the Resources. The main purpose of SJS is to optimal assignment of the tasks in the RP. This particular procedure is called Task Assignment Approach (TAA). The main objective of this topic is to depict one particular model of CSP and two algorithms related to TAA, one of them is Serial Task Assignment Approach (STAA) and another one is Optimal Task Assignment Approach (OTAA).

Keywords: Cloud computing, Quality of Service, Cloud User, Cloud Service Provider, Request Queue, Smart Job Scheduler, Resource Pool, Task Assignment Approach, Serial Task assignment Approach, Optimal Task Assignment Approach.

2. INTRODUCTION

Cloud computing [1,2,3,4] is the process of delivering computing as a service rather than a product, whereby shared resources, software, and information are provided to users and other devices as a utility over the network. Cloud computing environment is highly dynamic: the system load and computing resource utilization exhibit a rapidly changing characteristic over time. Therefore Cloud service provider normally over-position computing resources to accommodate the peak load and computing resources are typically left under-utilize in nonpeak time. Cloud environment allows users to use applications without installation and access their personal files at any computer with Internet access.

End users access cloud based applications through a web browser or a light weight desktop or mobile app while the business software and data are stored inside CSP at a remote location. Cloud application providers [6, 7] strive to give the better service and performance than if the software programs were installed locally on end-user machines. Cloud environment [5] is used in lot of fields like in IT industries, educational institute as well as in other industries.In this paper we proposed Cloud Service Provider (CSP model and CSP has mainly three parts- Request Queue (RQ), Smart Job Scheduler (SJS) and Resource pool (RP). All service requests which are come from Cloud Users are stored in RQ. Now the requested processes must communicate with SJS and SJS communicate with RP and tries to assign each of these jobs as per their requirement to the resources. But the main problem here to assign the jobs to the resources and the total time require to execution of those jobs. The jobs assignment task is done by SJS. So SJS must need to assign the task such a way that assignments of the jobs to the resources must be fruitful as per as CU requests and the total execution time must be optimal of the whole operations. In next two sections discuss about our proposed model of CSP and two different Task Assignment Approach algorithms which assigns the task to the resource as per the CU's demand and also to optimize the total time for execution.

3. TASK ASSIGNMENT MODEL FOR CLOUD SERVICE PROVIDER

In this section introduce a task assignment model for Cloud Service Provider. But before starting our discussion about the CSP model we have cited some relevant issues related to our model.

Queueing model [5, 10**]** is used to approximate a real queueing situation or system, so the queueing behavior can be analyzed mathematically. Queueing models allow a number of useful steady state performance measures to be determined. Concept of queueing theory concept comes from Kendall's notation.

Kendall's notation [11] is a standard notation for classifying queueing systems into different types. Kendall's notation mainly described by the notation A/B/C/D/E. **A-** Distribution of inter arrival times of customers; **B-** Distribution of service times; **C-** Number of servers; **D-** Maximum total number of customers which can be accommodated in system, i.e. system capacity; **E-** Queuing discipline. Let's take an example where- M/M/m/K/N- this would describe a queuing system with an exponential distribution for the inter arrival times of customers and the service times of customers, m servers, a maximum of K customers in the queueing system at once, and potential customers in the calling population. There are a lot of models available like M/M/1, M/M/2, etc. however we describe our model in M/M/2 structure in this paper to avoid calculation overhead.

An **M/M/2 queue** [5,10] represents the queue length in a system having two servers, where arrivals are determined by a Poisson process and job service times have a exponential distribution rate. An M/M/2 queue is a stochastic process whose state space is the set {0, 1, 2, 3...} where the value corresponds to the number of customers in the system, including any currently in service. Mathematical formula of M/M/2 queueing model is shown in figure 1 and figure 2.

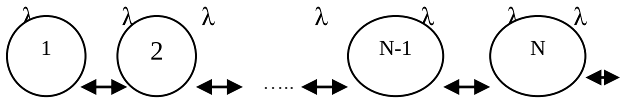

Figure 1: Diagram of M/M/1 Queueing model

$$Q = \begin{pmatrix} -\lambda & \lambda & & & \\ -\mu & -(\mu+\lambda) & \lambda & & \\ & -\mu & -(\mu+\lambda) & \lambda & \\ & & -\mu & -(\mu+\lambda) & \lambda \end{pmatrix}$$

Figure 2: Mathematical formulation of M/M/1 Queuing model

For easy description as well as easy calculation we proposed our CSP model based on M/M/2 queueing model diagrammatically which is shown in figure 3.

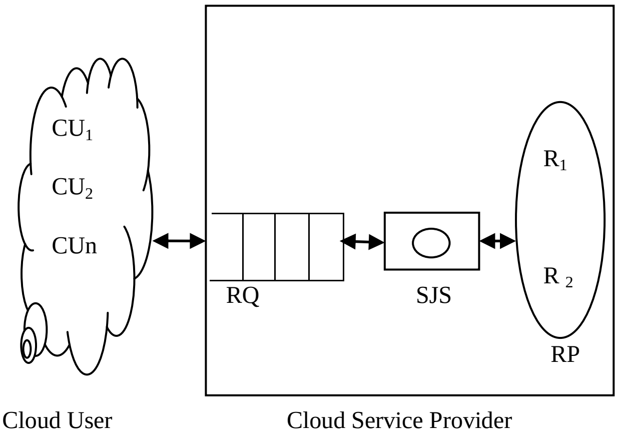

Figure 3: Cloud queuing model

In the above CSP model n number of cloud user send request to the cloud service provider (CSP). CSP stored the request at the RQ and then it directly communicates with SJS inside CSP. SJS then apply any one Task Assignment Approach algorithms (STTA, OTTA) [13] and tries to assign each of these tasks or jobs to the resources as per the request comes from the CU.

Now we briefly describe about the Task Assignment Approach (TAA) [13]. In this approach a process is consider to compose of multiple tasks and goal is to find optimal assignment policy for the tasks of individual process. Some preliminary assumptions we must consider in task assignment work as follows-

A) Here considered each request as a task and it is also very much important to inter task communication in some cases, so the task will have integrity itself and data transfer among the tasks will be minimized.

B) The amount of computation required by each task and speed of each processor or resource are known.

C) The cost of processing each task on every resource of the system is known. The cost usually derived based on the information about the speed of each processor and amount of computation required by each task.

D) The Intercrosses Communication (IPC) costs between every pair of tasks are known. The IPC cost is negligible or zero for task assigned to the same node. They are usually estimated by an analysis static program of a process. Suppose two task communicate n times and if the average time for each inter-task communication is t, the inter-task communication cost for the two tasks is n*t.

E) Resource requirement of the tasks; the availability of resources and precedence relationship among the task also need to be known.

F) The main constrain of the model is that reassignment of the task is generally not possible.

With the above assumptions, the task assignment algorithms seek to assign the tasks of a process to particular resource as per as requirement of CU.

Table 1: Inter-task Communication Costs

	T$_1$	**T$_2$**	**T$_3$**	**T4**	**T$_5$**	**T$_6$**
T$_1$	0	6	4	0	0	12
T$_2$	6	0	8	12	3	0
T$_3$	4	8	0	0	11	0
T$_4$	0	12	0	0	5	0
T$_5$	0	3	11	5	0	0
T$_6$	12	0	0	0	0	0

Table 1 represent Inter-task communication costs (here cost means how much time require of communication between two tasks); where the numerical value represents the time require communicating between two tasks. If communication between two tasks is not possible then the value must consider as zero. Ex: Time requires communicating between tasks T$_1$ and T$_2$ is 6.

Table 2 represent the Execution Costs (here cost means the amount of time require to execute the tasks) into of the tasks. In the above table the numerical value represents amount of time require to execute the tasks into every resources individually. An infinity cost for a particular task against a particular resource indicates that the task cannot be executed on that resource due to the task's requirement of specific resources that are not available on that resource. Ex: task T$_2$ cannot be executed on resource S$_2$.

In the next section we describe two different algorithms of Task Assignment Approach and describe their performance and key benefits of TAA algorithms.

Table 2: Execution Costs

Tasks	Resource	Resource
	R_1	R_2
T_1	5	10
T_2	2	∞
T_3	4	4
T_4	6	3
T_5	5	2
T_6	∞	4

.4. PERFORMANCE ANALYSIS

In this section describe two different TAA algorithms and describe the procedure of the assignment of task into the resources. If there are m number of tasks and q number of resources, then there are m^q possible assignments of tasks to resources. However, the actual number of possible assignments of tasks to resources may be less than m^q due to the restriction that certain tasks cannot be assigned to certain resources due to their specific resource requirements. TAA algorithms must recover those assignment problems.
Before start to describe the algorithms we consider that there are six task { T_1,T_2,T_3,$T_{4,}$$T_5$,$T_6$} and two resources {$R_1$,$R_2$} are available. Using the TAA algorithms we assign the six tasks into two resources such a way that total assignment cost must be optimum.

4.1. Serial Task Assignment Approach:-

In STAA approach tasks must be assigned to the available resources serially. Means first some number tasks assign to resource 1, next to resource 2 and so on. In this way all the tasks are assigned to all the resources serially.
For example if consider table 1 and table 2 then get another table 3, where the procedure of STTA algorithm is discussed.
Table 3 shows a serial assignment of the tasks to the two resources in which first three tasks are assigned to resource R_1 and remaining three are assigned to resource R_2.This assignment is aimed at minimizing the total execution costs. But we must to consider the execution cost as well as inter-process communication cost which must be shown below-

Table 3: Serial Assignment

Task	Resource
T_1	R_1
T_2	R_1
T_3	R_1
T_4	R_2
T_5	R_2
T_6	R_2

Serial Assignment Execution Cost = $X_{11}+X_{21}+X_{31}+X_{42}+X_{52}+X_{62}$ = 5+2+4+3+2+4 = 20

Serial Assignment Communication Cost = $C_{14}+C_{15}+C_{16}+C_{24}+C_{25}+C_{26}+C_{34}+C_{35}+C_{36}$ = 0+0+12+12+3+0+0+11+0 = 38

Serial Assignment Total Cost = X+C = 20+38 = 58

So total serial assignment cost is 58 in case of STTA algorithms. To reduce the total assignment cost next section we describe another TAA algorithm.

4.2. Optimal Task Assignment Approach:-

In OTTA the problem of finding an assignment of task to resources that minimize the total execution and communication cost elegantly analyzed using a network flow model and network flow algorithm. In this approach, an optimal assignment is found by creating static assignment graph. Using the above procedure every task must be assign to every other resource; those resources are available to reduce the overall cost of the system.
For example if consider the table 1 and table 2 then get another table 4, where produce OTTA algorithm is discussed.

Table 4 shows a optimal assignment of the task to the two resources using the procedure of network flow graph and network flow algorithm. So first five tasks T_1 to T_5 must be assigned to resource $R_1$and task T_6 only assign to resource R_6. This assignment is aimed at optimizing the total execution costs. But here also we must need to consider the execution cost as well as inter-process communication cost which must be shown below-

Table 4: Optimal Assignment

Task	**Resource**
T_1	R_1
T_2	R_1
T_3	R_1
T_4	R_1
T_5	R_1
T_6	R_2

Optimal Assignment Execution Cost = $X_{11}+X_{21}+X_{31}+X_{41}+X_{51}+X_{62}$ = 5+2+4+6+5+4 = 26

Optimal Assignment Communication Cost = $C_{16}+C_{26}+C_{36}+C_{46}+C_{56}$ = 12+0+0+0+0 = 12

Optimal Assignment Total Cost = X+C = 26+12 = 38

So in case of OTAA overall assignment cost must be reduced to 38; which is too much smaller compare to STAA where total serial assignment costs are 58.

Key benefits of Task Assignment Approach:-

a) Minimizing Inter-process communication cost.
b) Quick turnaround time of complete process.
c) A high degree of parallelism.
d) Efficient utilization of system resources.

5. CONCLUSION AND FUTURE WORK

Last two sections have been describing two different types of Task Assignment Approach Models. In our model, the Smart Job Scheduler mainly responsible for assigning the tasks to the pertinent resources. If there are m number of tasks and p number of resources then there are m^p numbers of possibility to assign the task to the resources. To remove this problem mainly the Task Assignment Approach algorithms are introduced and removing the problem of task assignment to the resources. The above section describe the Cloud Service Provider model where the TAA algorithms are applied in the Smart Job Scheduler which is in side CSP and recover the problem of assignment of the requested tasks to the available resources in minimum execution time as well as minimum inter-process communication time between the tasks.

For future, next generation Task Assignment Algorithms have been developing to reduce the overall assignment costs of CSP in Cloud environment and to reduce the complexity of CSP model when the number of resources would be maximum.

6. REFERENCES

[1] "Service Performance and Analysis in Cloud Computing" by Kaiqi Xiong, Harry Perros 978-0-7695- 3708-5/09 $25.00 © 2009 IEEE page- 693-700

[2] "Virtual Infrastructure Management in Private and Hybrid Clouds" by Borja Sotomayor, Rubén S. Montero and Ignacio M. Llorente, Ian Foster 1089-7801/09/$26.00 © 2009 IEEE

[3] "Research on Distributed Architecture Based on SOA" by Hongqi Li, Zhuang Wu 978-0-7695-3522-7/09 $25.00 © 2009 IEEE 670-674

[4] "A Berkeley View of Cloud computing". M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, M. Zaharia. Above the Clouds: Technical Report No. UCB/EECS-2009-28, University of California at Berkley, USA, Feb. 10, 2009

[5] Hock, N.C., Queueing Modelling Fundamentals. JOHN WILEY&SONS, 1997

[6] "An Approach to a Cloud Computing Network" by Francesco Maria Aymerich, Gianni Fenu1, Simone Surcis 978-1-4244-2624-9/08/$25.00 ©2008 IEEE 113 page 113-118

[7] "Cloud Computing and Services Platform Construction of Telecom Operator" by Xu Lei, Xin Zhe, Ma Shaowu, Tang Xiongyan. Broadband Network & Multimedia Technology, 2009. IC-BNMT '09. 2nd IEEE International Conference on Digital Object Identifier, pp. 864 – 867.

[8] "Service Performance and Analysis in Cloud Computing" , Kaiqi Xiong and Harry Perros 2009 Congress on Services –I

[9]" An Optimistic Differentiated Service Job Scheduling System for Cloud Computing Service Users and Providers" by *Luqun Li* 2009 Third International Conference on Multimedia and Ubiquitous Engineering page-295-299

[10] "Probability, Markov chains, queues, and simulation: the mathematical basis of performance modeling". Stewart, William J. (2009) Princeton University Press. p. 409. ISBN 0-691-14062-6.

[11] "Stochastic Processes Occurring in the Theory of Queues and their Analysis by the Method of the Imbedded Markov Chain". Kendall, David G. (September 1953). Annals of Mathematical Statistics 24 (3): 338–354. doi:10.1214/aoms/1177728975. JSTOR 2236285

[12] "A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing" by Martin Randles, David Lamb, A. Taleb-Bendiab 2010 IEEE 24th International Conference on Advanced Information Networking and Applications Workshops page 551-556

[13] "Distributed Operating System Concept and Design - Pradeep K. Sinha; PHI publication.