Advanced Reservation based SLA for Resource negotiations in Grid Scheduling

Animesh Kuity
Electronics & Computer Engineering Department
Indian Institute of Technology Roorkee
Roorkee-247667
animesh2kuity@gmail.com

Sateesh Kumar Peddoju
Electronics & Computer Engineering Department
Indian Institute of Technology Roorkee
Roorkee-247667
sateesh@ieee.org

***Abstract*— In grid environment, multiple resource providers work together in order to accomplish a complex job. The service level agreement is negotiated between client and a provider for executing the job on high performance computing resources. The performance of the negotiation process highly depends on number of resources selected for the negotiation and process of the negotiation.**

This paper proposes an advance reservation based resource negotiation using the deviation based resource scheduling algorithm to get commitment of the resources towards the job. The proposed model provides the commitment not only when resources are available but also when resource are not available for that time. The simulations of the proposed model are done on Gridsim simulator. The results are produced. The results indicate the improved performance in terms of number of successful SLA creations and throughput.

***Keywords*—Grid Computing, SLA, DRS, Advance reservation based resource negotiation, QoS.**

I. INTRODUCTION

Current internet technology mainly focuses on computation and information exchange among computers but not coordinated use of resources at multiple site for computation. Sharing not only confined to file exchange but direct access to computer, software, data and other resources as is required by a large range of collaborative problem solving and resource brokering strategies emerging in industry, science and engineering. Grid [1] is a dynamic collaborative computing environment which allows sharing and allocation of resources among multiple administrative domains.

Service level agreement [2] is a contract between user and resource providers which state QoS required by the job, restriction on utilization of resources and penalties during violation of objective. Users need some commitment and assurance on the top of allocated resources to accomplish a job and for dealing with erroneous condition. These terms need to be agreed upon before use and manifested in form of SLA.

When a job is submitted by a user to resource broker, a SLA is created between user and resource broker by describing all the terms and conditions. Resource broker works as middle man. Next, SLA(s) are created between resource broker and service providers [3, 4].

Most of the current research [5, 6] address that SLA is negotiated between resource broker and service provider but they do not address what happens if none of the service provider is available for that time. The main focus of this paper is to address these issues. A simple architecture of SLA based job scheduling is shown in Fig. 1.

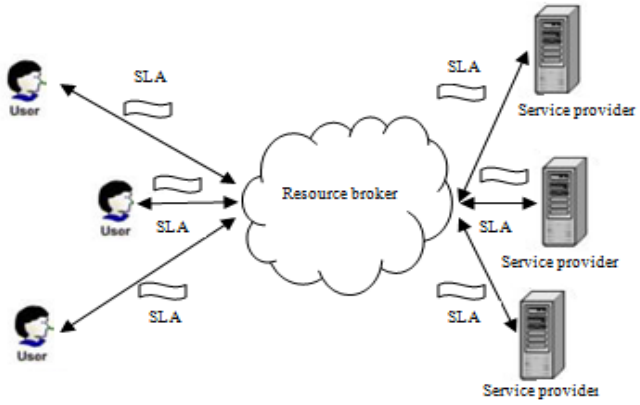

Figure 1. SLA based job scheduling.

II. RESOURCE SELECTION FOR EXECUTING THE JOB

Whenever a user submits a job, resource broker does the entire task on the behalf of user that is selection of suitable resources to execute the job and negotiation with the resource providers. The payment for the job will be the sum of cost incurred due to resource broker and service provider. According to the deviation value computed using deviation based resource scheduling algorithm (DRS) [7], jobs are scheduled to the resource provider.

A. Computation of deviation coefficient

This proposed model schedules the job according to the requirement of the job rather than resources' own scheduling metric. It calculates the deviation using the deviation based

resource scheduling algorithm against the requirement of the job and the amount of resources that resource provider have. Resources are ordered according to the deviation coefficient. Then, it is selected to fulfil the user requirement.

In deviation based resource scheduling algorithm (DRS), the percentage of deviation D_{ij} is calculated against ^{jth} parameter of the request and the corresponding parameter of the i^{th} available resources using (1).

For every available resources from i=1 to N

Percentage of deviation

$$D_{ij}=\begin{cases}\frac{A_v(t)-R_j(t)}{A_v(t)}\times 100 & if\ A_v(t)>R_j(t)\\ \frac{A_v(t)-R_j(t)}{R_j}\times 100 & if\ R_j(t)>A_v(t)\end{cases} \quad (1)$$

Where j=number of resource parameter from 1 to m

$A_v(t)$ is the j^{th} parameter of the i^{th} available resources at time t

$R_j(t)$ is the j^{th} parameter of the request at time t

After calculating percentage deviation D_{ij} for every available resource, in order to scale down percentage of deviation between -1 to +1, percentage of deviation is divided by maximum percentage deviation if it is positive otherwise it is divided by minimum percentage of deviation. DRS first selects the resource that has zero deviation value. If no resource is available with zero deviation value then it selects the resource of positive deviation value and at last resources of negative deviation value.

III. Advance Reservation Based Resource Negotiation

After selecting the resources using DRS algorithm, the advance reservation [8] based resource negotiation are used to get the commitment of the resources towards the completion of the job. If resource with zero or positive deviation value is available then negotiation will be done with the resource using advance reservation by setting the reservation time as current time. If current time slot is not available then resource provider will send their free time slot. Resource broker will then negotiate for one of those free time slots as per user requirement to execute the job. If there does not exist any zero or positive deviation value resource that means user requirement of the job is greater than availability of the resources. In this case two or more resource providers are needed to execute the job. For that the proposed model uses advance reservation based resource negotiation thread. The job of the advance reservation based resource negotiation thread is, it will first select the resources which are required to execute the job. Then it will create as many threads as the selected number of resources and request them simultaneously to send their free time slot. A common free time slot is chosen from those free time slot and resource broker negotiates with all the selected resources using advance reservation with time as the common free slot time.

If any of the resources disagree then resource broker selects another resource from the remaining resources. After successful completion of negotiation, the job is submitted to the selected resource(s). Then, the entire process is monitored by a monitoring engine that monitors the proper consumption of the resource(s). If any kind of violation occurs then an event will be generated and appropriate action is taken.

ALGORITHM -1

1. Identify the minimum number of resource(s) required to execute the job selected using DRS;
M=minimum number of resource(s)
2. If resource with zero or positive deviation value is available
//set reservation time to current time
Set time=current time;
CreateReservation with the resource
If createReservation is successful
Then commitReservation with that resource
Else
Send request with QueryFreeTime for free time slot
Receive free time slot and send commitment for one of the free time slot
3. Else
For 1 to M
Create resource negotiation thread
For each resource negotiation thread
Send request with QueryFreeTime for free time slot
Find a common free time slot that is suitable to execute the job
For 1 to M
Set time=common free time slot
CreateReservation with the resource
If createReservation is successful
Then commitReservation with that resource
Else
Choose another resource for negotiation
Repeat the process

IV. Simulation Results

In order to simulate advanced reservation based SLA for resource negotiation in grid scheduling on GridSim, we have created 10 resources, each having more than one machine and each machine have more than one processing element using a random function. 50 users are created, each having 30 to 50 jobs i.e gridlets with full specification of the user requirement

that is number of processing element, RAM in MB, CPU percentage and along with the other requirements, every job request has a set of restriction on parameters. Using deviation based resource scheduling algorithm, deviation value of all resources are computed against the requirement of job. The job is scheduled for execution according to the deviation value. As resource broker schedules the job according to the user requirement, so number of jobs submitted to each resource depends on the user requirement and availability of resources. From Fig. 2, it is shown that out of 70 request 31% of the requests (i.e 22 requests) are fulfilled by the advance reservation based resource negotiation. If advance reservation based resource negotiation is not there, then all the requests that cannot be fulfilled by a single resource would fail. The advance reservation based resource negotiation not only fulfils the user requirement when more than one resource is required to execute the job but also when no resource is available for the current time.

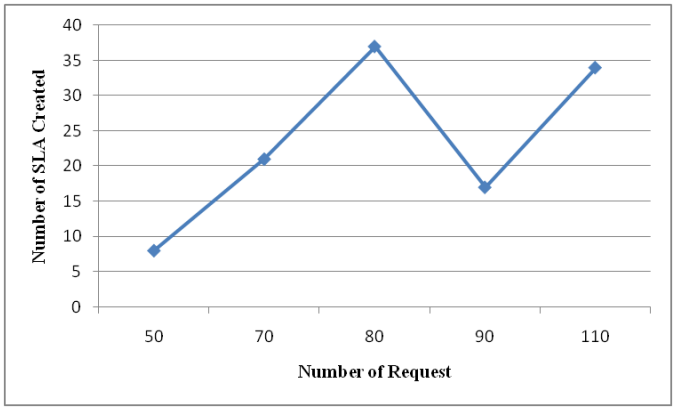

Figure 2. Number of Request vs. Number of SLA created with 10 resources.

So, throughput of the service level agreement based grid scheduling using advance reservation based resource negotiation is maximized due to the fact that resource broker not only negotiate for the job that requires more than one resource but also eliminate the fact, if resources are not available for that time period using common free slot searching. Throughput depends on the number of successful SLA creation, which in turn depends on the successful negotiation between resources. If numbers of requests are less, then resources can successfully negotiate with other resources for each job easily because each resource has less number of jobs so they can easily negotiate for another job execution. Fig. 3 shows the improve performance of the number of requests vs throughput for 10 resources.

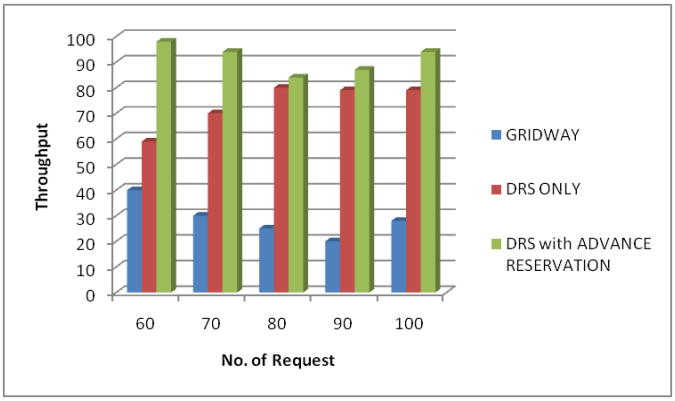

Figure 3. Number of request VS Throughput.

V. Conclusions

Grid computing environment can be highly heterogeneous. Different hardware characteristics such as CPU speed, cache and interconnect can impact the time that can take to execute a job. Different sites use different local scheduling policies. Some site might favor parallel job or may restrict on finish time of the job. So SLA can be used to remove all these difficulties. A SLA is a contract between participating parties which acts as a legal document for a set of guarantee and QoS metrics. This paper proposes service level agreement based grid scheduling using advance reservation based resource negotiation which improves not only throughput but also maximize the utilization of resources. In future, this work on service level agreement based grid scheduling could be upgraded to make it useful for workflow applications in which each job contains more than one coordinated task.

References

[1] I. Foster, C. Kesselman, S.Tuecke, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations", International Journal of Supercomputer Applications, 15(3), 2001.

[2] P.Balakrishnan, S. Thamarai Selvi and G.Rajesh Britto, "GSMA based Automated Negotiation Model for Grid Scheduling", publish in SCC 2008WIP, IEEE Congress on Services, SERVICES 2008, July 8-11,2008, Honolulu, Hawaii, USA, pp. 569-570.

[3] V. Stantchev and C. Schröpfer, " Negotiating and Enforcing QoS and SLAs in Grid and Cloud Computing," In Advances in Grid and Pervasive Computing , vol. 5529, pp. 25-35, 2009.

[4] A. Ismail, J. Yan and J. Shen, " An offer generation approach to SLA negotiation support in service oriented computing," Service Oriented Computing and Applications , vol.4 (4) , pp. 277-289, 2010.

[5] J. MacLaren, R. Sakellariou, K.T. Krishnakumar, J.Garibaldi, and D. Ouelhadj, "Towards Service Level Agreement Based Scheduling on the Grid", in Proceedings of the 2nd European Across Grids Conference, 2004, pp. 100-102.

[6] V. Yarmolenko, R. Sakellariou, D.Ouelhadj, J. M.Garibaldi, SLA Based Job Scheduling: A Case Study on Policies for Negotiation with Resources. Proceedings of the UK e-Science All Hands Meetings, Nottingham, 2005, pp. 20-25.

[7] P.Balakrishnan, S. Thamarai Selvi and G.Rajesh Britto, "Service Level Agreement based Grid Scheduling" International Conference on Web Service publish in SCC 2008WIP, Honolulu, Hawaii, USA, 2008, pp. 203-210.

[8] Anthony Sulistio , Rajkumar Buyya, Rao Kotagiri, "Advance Reservation and Revenue-based Resource Management for Grid System", Doctor of Philosophy thesis, University of Melbourn, Australia, may 2008.