SCHEDULING IN GRID TO MINIMIZE THE IMPOSED OVERHEAD ON THE SYSTEM AND TO INCREASE THE RESOURCE UTILIZATION

Preetha Evangeline.D

Post Graduate Scholar, Dept Of computer science& engineering, Karunya University, Coimbatore-641114

Preethadavid4@gmail.com

ABSTRACT

Grid computing is one of the area which is having outstanding advancement nowadays. Grid computing have emerged to harness distributed computing and network resources to provide powerful services. The non-deterministic characteristic of the resource availability in these new computing platforms raises an outstanding challenge. Since grid computing is rapidly growing in heterogeneous environment it is used for utilizing and sharing large scale resources to solve complex problems. One of the most challenging task in the area of grid is resource provisioning. As it is very well known that there are enormous amount of resources present in the grid environment, it is very hard to schedule them. There are two types users namely the local users and the external users. Resources has to be provided to both the users request. When the user submits the job, the request has to be satisfied by providing the resources. Here the resources are provided in the form of virtual machines (VM).The problem arises when there is insufficiency of resources to provide both the users. The problem becomes more hectic when the external requests have more QoS requirements. The local users can be provided with the resources by preempting the VMs from the external requests which causes imposed overhead on the system. Now the idea is how to decrease the preemption of VMs and how to utilize the resources efficiently. Here we propose a policy in Intergrid which reduces the no of VM preemption and which will also distribute the external requests accordingly so that preemption of valuable requests is reduces.

KEYWORDS

QoS, Preemption, PAP, Intergrid, Lease abstraction, Analytic queing model

1. INTRODUCTION

Providing the resources for user applications is one of the main challenges in grid. Intergrid supports sharing, selection and aggregation of resources across several grid which are connected through high bandwidth. Job abstraction is widely used in resource management of Grid environments. However, due to advantages of Virtual Machine (VM) technology, recently, many resource management systems have emerged to enable another style of resource management based on lease abstraction. Intergrid also interconnects islands of virtualized grids. It provides resources to the users in the form of virtual machines (VM) and it also allows users to create environment for executing their applications on the VMs. In each constituent Grid, the provisioning rights over several clusters inside the Grid are delegated to the InterGrid Gateway (IGG).The intergrid gateway coordinates the resource allocation for the requests that is coming from outer grids(external requests) through predefined contracts between the grids. Hence, resource provisioning is done for two different types of users, namely: local users and external users. Typically, local requests have priority over external requests in each cluster . As illustrated in Fig. 1, local users (hereafter termed as local requests), refer to users who ask their local cluster resource manager (LRM) for resources. External users (hereafter termed as external requests) are those users who send their requests to a gateway (IGG) to get access to a larger amount of shared resources. In our previous research we demonstrated how preemption of external requests in favor of local requests can help serving more local requests. However, the side-effects of preemption are twofold:

• From the system owner perspective, preempting VMs imposes a notable overhead to the underlying system and degrades resource utilization.
• From the external user perspective, preemption increases the response time of the external requests. As a result, both the resource owner and external users benefit from fewer VM preemptions in the system.

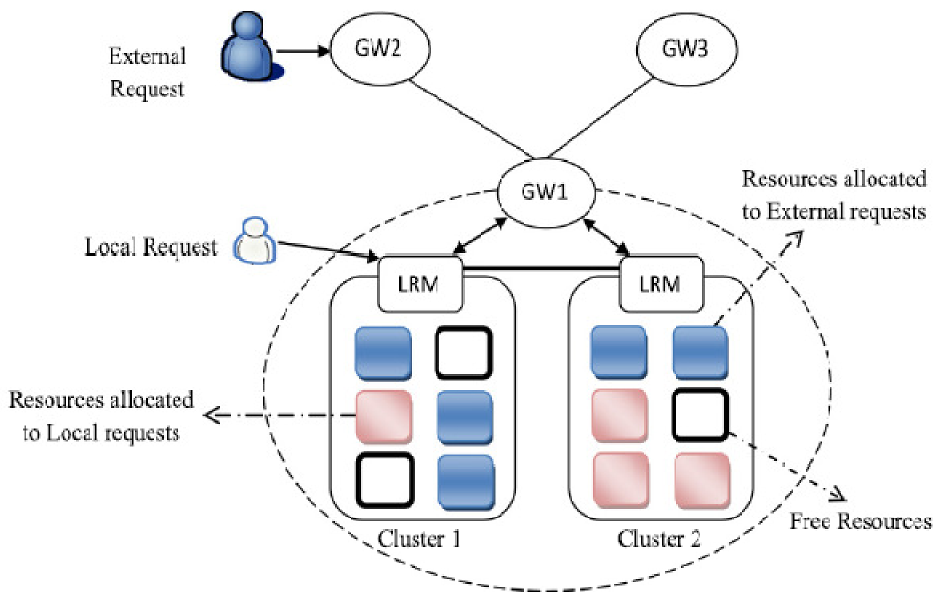

Fig1: Contention between Local Request and External Request

We believe that with the extensive current trend in applying VMs in distributed systems, and considering preemption as an outstanding feature of VMs, it is crucial to investigate policies that minimize these side-effects. Therefore, one problem we are dealing with in this research is how to decrease the number of VM preemptions that take place in a virtualized Grid environment. The problem gets complicated further when external requests have different levels of Quality of Service (QoS) requirements For instance, some external requests can have deadlines whereas others do not. Preemption affects the QoS constraints of such requests. This implies that some external requests are more valuable than others and, therefore, more precedence should be given to valuable requests by reducing the chance of preemption of these requests. To address these problems, in this paper, we propose a QoS and preemption-aware scheduling policy for a virtualized Grid which contributes resources to a federated Grid. This scheduling policy comprises of two parts. The first part, called workload allocation policy, determines the fraction of external requests that should be allocated to each cluster in a way that minimizes the number of VM preemptions. The proposed policy is based on the stochastic analysis of routing in parallel, non-observable queues. Moreover, this policy is knowledge-free (i.e. it is not dependent on the availability information of the clusters). Thus, this policy does not impose any overhead on the system. However, it does not decide the cluster that each single external request should be dispatched upon arrival. In other words, dispatching of the external requests to clusters is random. Therefore, in the second part, called dispatch policy, we propose a policy to find out the cluster to which each request should be allocated to. The dispatch policy has the awareness of request types and aims to minimize the likelihood of preempting valuable requests. This is performed by working out a deterministic sequence for dispatching external requests. In summary, our paper makes the following contributions:

- Providing an analytical queuing model for a Grid, based on the routing in parallel non-observable queues.
- Adapting the proposed analytical model to a preemption-aware workload allocation policy.
- Proposing a deterministic dispatch policy to give more priority to more valuable users and meet their QoS requirements.

Evaluating the proposed policies under realistic workload models and considering performance metrics such as number of VM preemptions, utilization, and average weighted response time.

2. INTERGRID

INTERGRID is inspired by the peering agreements established between Internet Service Providers (ISPs) in the Internet, through which ISPs agree to allow traffic into one another's networks. The architecture of InterGrid relies on InterGrid Gateways (IGGs) that mediate access to resources of participating Grids. The InterGrid also aims at tackling the heterogeneity of hardware and software within Grids. The use of virtualization technology can ease the deployment of applications spanning multiple Grids as it allows for resource control in a contained manner. In this way, resources allocated by one Grid to another are used to deploy virtual machines. Virtual machines also allow the use of resources from Cloud computing providers. The InterGrid aims to provide a software system that allows the creation of execution environments for various applications (a) on top of the physical infrastructure provided by the participating Grids (c). The allocation of resources from multiple Grids to fulfil the requirements of the execution environments is enabled by peering arrangements established between gateways (b). A Grid has pre-defined peering arrangements with other Grids, managed by IGGs and, through which they co-ordinate the use of resources of the InterGrid. An IGG is aware of the terms of the peering with other Grids; selects suitable Grids able to provide the required resources; and replies to requests from other IGGs.

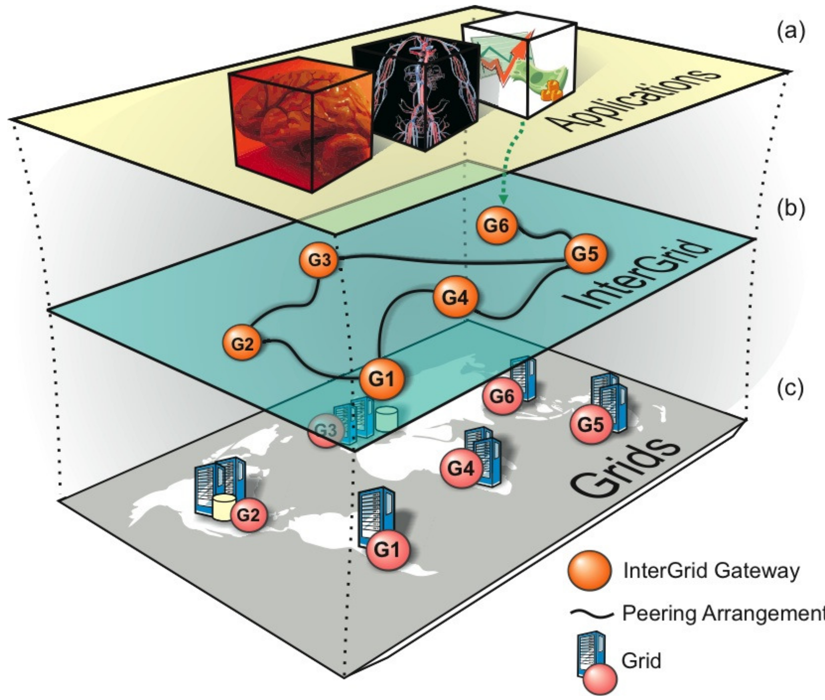

Fig 2: Layers of intergrid

The Local Resource Manager (LRM)1 is the resource manager in each cluster which provisions resources for local and external (Grid) requests. Resource provisioning in clusters of InterGrid is based on the lease abstraction. A lease is an agreement between resource provider and resource consumer whereby the provider agrees to allocate resources to the consumer according to the lease terms presented by the consumer .Virtual Machine (VM) technology is used in InterGrid to implement lease-based resource provisioning InterGrid creates one lease for each user request (in this paper we apply these two terms interchangeably).

3. ANALYTICAL QUEING MODEL:

In this section, we describe the analytical modeling of preemption in a virtualized Grid environment based on routing in parallel queues. This section is followed by our proposed scheduling policy in IGG built upon the analytical model provided in this part. The queuing model that represents a gateway along with several non-dedicated clusters (i.e. clusters with shared resources between local and external requests) is depicted in Fig. 3. There are N clusters where cluster j receives requests from two independent sources. One source is a stream of local requests with arrival rate λj and the other source is a stream of external requests which are sent by the gateway with arrival rate $\hat{\Lambda} j$. The gateway receives external requests from other peer gateways [12] (G1, . . . , Gpeer in Fig. 3). Therefore, external request arrival rate to the gateway is $\Lambda = \overline{\Lambda}1 + \overline{\Lambda}2 + \cdot \cdot \cdot + \overline{\Lambda}peer$ where peer indicates the number of gateways that can potentially send external requests to the gateway. Submitted local requests to cluster j must be executed on cluster j unless the requested resources are occupied by another local request or a Non-preemptive external request. The first and second moment of service time of local requests in cluster j are τj and μj, respectively. On the other hand, an external request can be allocated to any cluster but it might get preempted later on. We consider θj and ωj as the first and second moment of service time of external requests on cluster j, respectively. For the sake of clarity, Table 1 gives the list of symbols we use in this paper along with their meaning.

Table1: List of Symbols

Table 1
Description of symbols used in the queuing model.

Symbol	Description
N	Number of clusters
M_j	Number of computing elements in cluster j where $1 \leq j \leq N$
$\bar{\Lambda}_j$	Original arrival rate of external requests to cluster j
$\hat{\Lambda}_j$	Arrival rate of external requests to cluster j after load distribution
Λ	$= \sum_{i=1}^{peer} \bar{\Lambda}_i = \sum_{j=1}^{N} \hat{\Lambda}_j$
θ_j	Average service time of a external request on cluster j
ω_j	Second moment of external requests service time on cluster j
γ_j	$= \theta_j \cdot \hat{\Lambda}_j$
λ_j	Arrival rate of local requests to cluster j
κ_j	Arrival rate of local requests plus external requests to cluster j
τ_j	Average service time of local requests on cluster j
μ_j	Second moment of local requests service time on cluster j
ρ_j	$= \tau_j \cdot \lambda_j$
m_j	$= \frac{\hat{\Lambda}_j}{\kappa_j}\omega_j + \frac{\lambda_j}{\kappa_j}\mu_j$
u_j	Utilization of cluster j $(=\gamma_j + \rho_j)$
r_j	Average response time of local requests on cluster j
η_j	Number of VM preemptions that happen in cluster j
T	Average response time of all external requests
T_j	Average response time of external requests on cluster j
$\bar{v}_j$	Average number of VMs required by external requests
d_j	Average duration of external requests
s_{ij}	Processing speed (MIPS) of processing element i in cluster j

Indeed, the analytical model aims at distributing the total original arrival rate of external requests (Λ) amongst the clusters. In this situation if we consider each cluster as a single queue and the gateway as a meta-scheduler that redirects each incoming external request to one of the clusters, then the problem of scheduling external requests in the gateway (IGG) can be considered as a routing problem in distributed parallel queues. Considering these situation, the goal of the scheduling in the IGG is to schedule the external requests amongst the clusters in a way that minimizes the overall number of VM preemptions in a Grid. To the best of our knowledge, there is no scheduling policy for such an environment with the goal of minimizing the number of VM preemptions. However, several research works have been undertaken in similar circumstances to minimize the average response time of external requests. Some initial experiments as well as results of our previous research intuitively imply that there is an association between response time and number of VM preemptions in the Grid. The regression analysis with least squares method (depicted in Fig. 4).

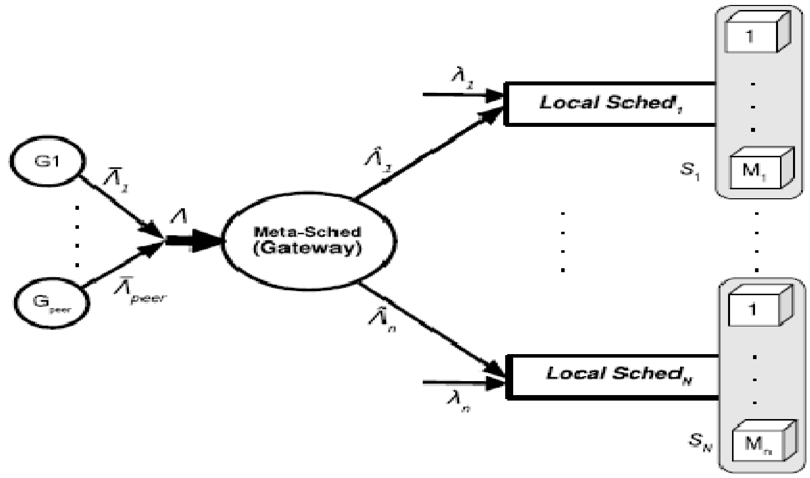

Fig 3: Analytical Queing Model

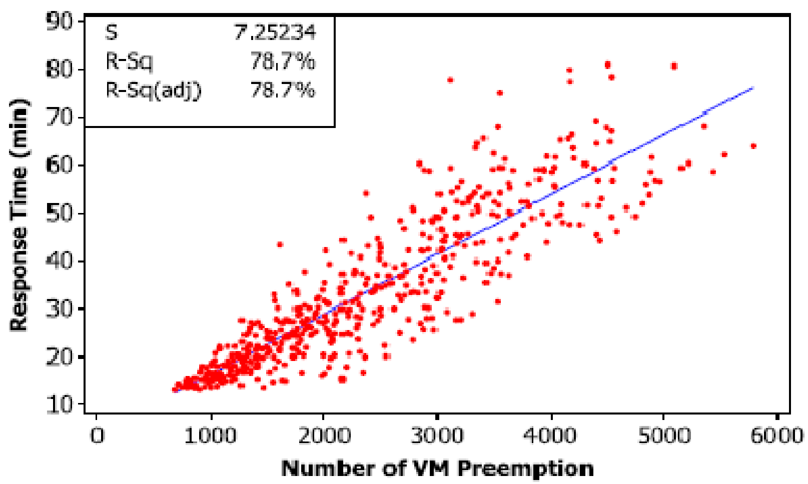

Fig 4: Regression between the number of VMs preempted and response time of external requests.

4.QoS And Preemption Aware Scheduling Policy

In this section, we propose a workload allocation policy and a dispatch policy. The positioning of this scheduling policy in IGG is demonstrated in Fig. 2. The proposed scheduling policy comprises of two parts. The first part, discusses how the analysis mentioned in the previous section can be adapted as the workload allocation policy for external requests in IGG. The second part, is a dispatch policy which determines the sequence of dispatching external requests to different clusters considering the type of external requests. On the other hand, in our scenario we encounter parallel requests (requests require more than one VM) that follow a general distribution. Additionally, we apply a conservative backfilling policy as the local scheduler of each cluster. The reason of using conservative backfilling is that it increases the number of requests getting served at each moment with respect to the SJF policy. Moreover, it is proved that conservative backfilling performs better in multi-cluster environments compared with other scheduling policies.Considering the above differences, we do not expect that the preemption-aware workload allocation policy still performs optimally. In fact,we examined how efficient the proposed analysis would be in a virtualized Grid environment by relaxing these assumptions. To adapt the analysis in a way that covers requests that need several VMs, we modify the service time of external requests on cluster j (θj) and local requests on cluster j (τj).

4.1 Dispatch policy

The algorithm proposed in the previous subsection determines the routing probability to each cluster (i.e. ˆΛj/Λ). However, it does not offer any deterministic sequence for dispatching each external request to the clusters. More importantly, as mentioned earlier, external requests are in different levels of QoS which implies that some external requests are more valuable. Hence, we would like to decrease the chance of preemption for more valuable requests to the minimum possible. We plan to put this precedence in place through a dispatch policy. In this part,wepropose a policy that, firstly, reduces the number of VM preemptions for more valuable external requests. Secondly, this policy makes a deterministic sequence for dispatching external requests. It is worth noting that the dispatch policy uses the same routing probabilities that worked out for each cluster using the workload allocation policy. The only difference is in the sequence of requests dispatched to each cluster. For this purpose, we adapt a Billiard strategy as the dispatching policy.

5. PERFORMANCE EVOLUTION

In this section, we discuss different performance metrics considered, the scenario in which the experiments are carried out; finally, experimental results obtained from the simulations are discussed.

5.1 User satisfaction

As mentioned earlier, both resource owners and users benefit from fewer VM preemptions. From the resource owner perspective, less VM preemption leads to less overhead on the underlying system and improves the utilization of resources. However, from the external user perspective, preemption has different impacts based on the lease types. One of the comtribution of this paper is to give more precedence to more valuable external users.

5.2 Resource utilization

Time overhead due to VM preemptions leads to resource under-utilization. Thus, from the system owner perspective, we are interested to see how different scheduling policies affect the resource utilization. Resource utilization for the Grid system is defined as follows:

$$\text{Utilization} = \left(1 - \frac{\sum_{j-1}^{N} \text{overhead}_j}{\sum_{i=1}^{N} \text{computationTime}_j}\right) \cdot 100$$

6. EXPERIMENTAL SETUP

We use GridSim as simulator, to evaluate the performance of the scheduling policies. We consider a Grid with 3 clusters with 64, 128, and 256 processing elements with different computing speeds (s1 = 2000, s2 = 3000, s3 = 2100 MIPS). This means that in the experiments we assume computing speed homogeneity within each cluster. This assumption helps us to concentrate more on the preemption aspect of resource provisioning. Moreover, considering that the resources are provisioned in the form of VMs, the assumption of homogeneous resources within the clusters is not far from reality. Each cluster is managed by an LRM and a conservative backfilling scheduler. Clusters are interconnected using a 100 Mbps network bandwidth. We assume all processing elements of each cluster as a single core CPU with one VM. The maximum number of VMs in the generated requests of each cluster does not exceed the number of processing elements in that cluster. The overhead time imposed by preempting VMs varies based on the type of external leases involved in preemption [For Cancelable leases the overhead is the time needed to terminate the lease and shutdown its VMs. This time is usually much lower than the time required for suspending or migrating leases and can be ignored.

7. EXPERIMENT RESULTS

7.1 Number of VM preemptions

The primary goal of this paper is to express the impact of scheduling policies on the number of VMs preempted in a Grid environment. Therefore, in this experiment we report the number of VMs getting preempted by applying different scheduling policies. As we can see in all sub-figures of Fig. 5, the number of VMs preempted increases by increasing the average number of VMs (Fig. 5(a)), duration (Fig. 5(b)), arrival rate of external requests (Fig. 5(c)), and arrival rate of local requests (Fig. 5(d)). In all of them PAP-RTDP statistically and practically significantly outperforms other policies.

7.2 Resource utilization

In this experiment we explore the impact of preempting VMs on the resource utilization as a system centric metric.In general, resource utilization resulted from applying PAP RTDP is drastically better than other policies as depicted in Fig.6.

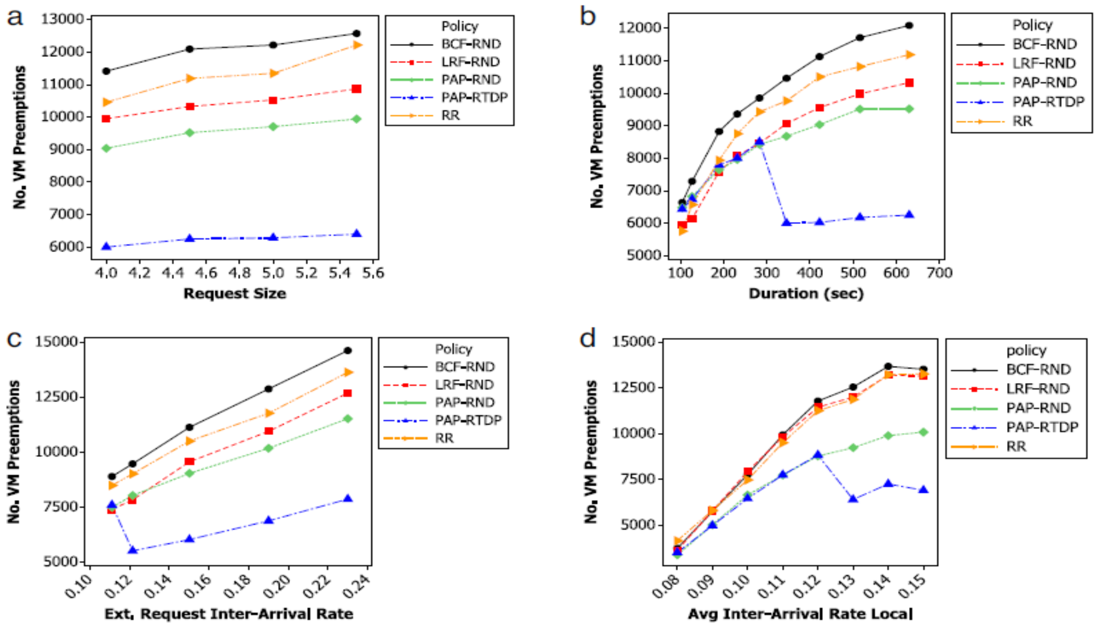

Fig 5: The no of VM preempted by applying different policies. a) the average number of VM's b) the average duration , c) arrival rate of external request, d) the arrival rate of local request.

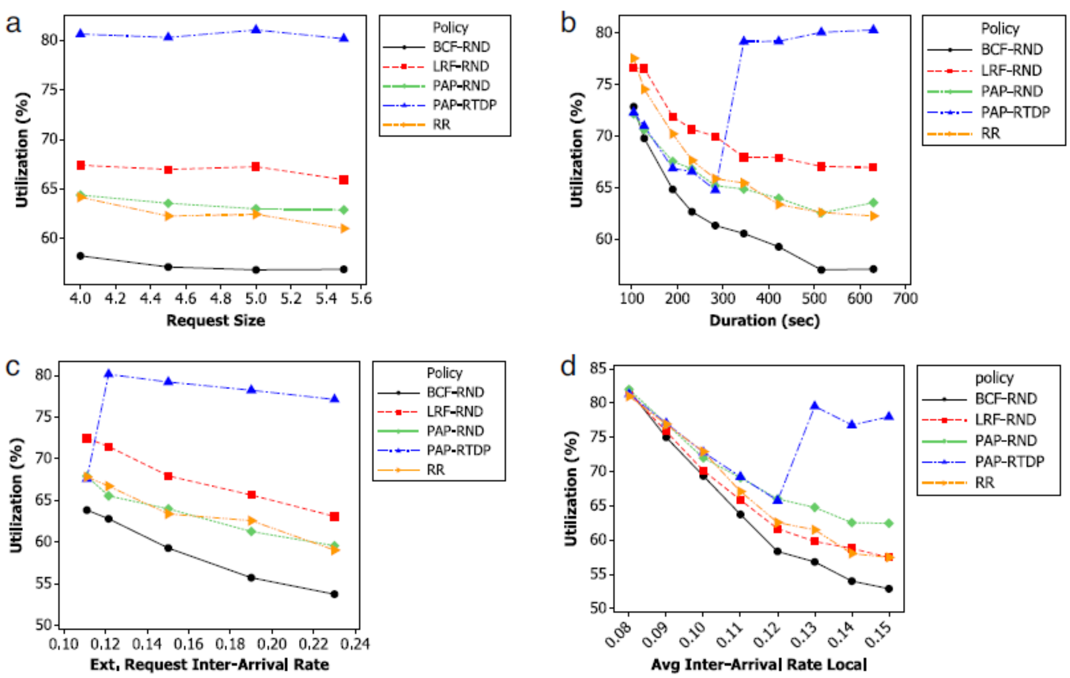

Fig 6. Resource utilization resulted from different policies.

8. RELATED WORKS

He et al. have aimed at minimizing response time and miss rate for non-real-time and soft real-time jobs separately in the global scheduler of a multi-cluster by applying a non-observable approach. They have also recognized workload allocation policy from job dispatching policy. Nonetheless, for job dispatching they have applied policies such as weighted round robin policies. By contrast, we consider a mixture of such jobs in our scenario which are affected by local requests in each cluster.

Assuncao and Buyya have proposed policies in IGG based on adaptive partitioning of the availability times between local and external requests in each cluster. Consequently, there is a communication overhead between IGG and clusters for submitting availability information.

Bucur and Epema have evaluated the average response time of jobs in DAS-2 [25] by applying local, global (external), and combination of these schedulers in the system. They have concluded that the combination of local and global schedulers is the best choice in DAS-2.They have also observed that it is better to give more priority to local jobs while the global jobs should also have chance to run.

He et al. have proposed a multi-cluster scheduling policy where the global scheduler batches incoming jobs and assigns each batch tuning to run the assigned jobs with minimized make span and idle times.

Amar et al. have added preemption to cope with the non optimality of the on-line scheduling policies. In the preemption policy jobs are prioritized based on their remaining time as well as the job's waiting cost. Schwiegelshohn and Yahyapour have investigated different variations of preemptive first-come first- serve policy for an on-line scheduler that schedules parallel jobs where the jobs' run times and end times are unknown.

Margo et al. have leveraged a priority scheduling based on preempting jobs for Catalina to increase the utilization of the resources. They determine the priority of each job based on the expansion factor and number of processing elements each job needs.

9. CONCLUSIONS AND FUTURE WORK

In this research we explored how we can minimize the side effects of VM preemptions in a federation of virtualized Grids such as InterGrid. We consider circumstances that local requests in each cluster of a Grid coexist with external requests. Particularly, we consider situations that external requests have different levels of QoS. For this purpose, we proposed a preemption-aware workload allocation policy (PAP) in IGG to distribute external requests amongst different clusters in a way that minimizes the overall number of VM preemptions that take place in a Grid. Additionally, we investigated on a dispatch policy that regulates dispatching of different external requests in a way that external requests with higher QoS requirements have less chance of getting preempted. The proposed policies are knowledge-free and do not impose any communication overhead to the underlying system.

References

[1] L. Amar, A. Mu'Alem, J. Stober, The power of preemption in economic online markets, in: Proceedings of the 5th International Workshop on Grid Economics and Business Models, GECON'08, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 41–57.

[2] J. Anselmi, B. Gaujal, Optimal routing in parallel, non-observable queues and the price of anarchy revisited, in: 22nd International Teletraffic Congress, ITC, Amsterdam, The Netherlands, 2010, pp. 1–8.

[3] N.T. Argon, L. Ding, K.D. Glazebrook, S. Ziya, Dynamic routing of customers with general delay costs in a multiserver queuing system, Probability in theEngineering and Informational Sciences 23 (2) (2009) 175–203.

[4] F. Bonomi, A. Kumar, Adaptive optimal load balancing in a non homogeneous multiserver system with a central job scheduler, IEEE Transactions on Computers 39 (1990) 1232–1250.

[5] A. Bucur, D. Epema, Priorities among multiple queues for processor co allocation in multicluster systems, in: Proceedings of the 36th Annual Symposium on Simulation, IEEE Computer Society, 2003, pp. 15–22.

[6] J.S. Chase, D.E. Irwin, L.E. Grit, J.D. Moore, S.E. Sprenkle, Dynamic virtual clusters in a Grid site manager, in: Proceedings of the 12th IEEE International Symposium on High Performance Distributed Computing, Washington, DC, USA, 2003, pp. 90–98.

[7] P. Chen, J. Chang, T. Liang, C. Shieh, A progressive multi-layer resource reconfiguration framework for time-shared Grid systems, Future Generation Computer Systems 25 (6) (2009) 662–673.

[8] B. Chun, D. Culler, T. Roscoe, A. Bavier, L. Peterson, M. Wawrzoniak, M. Bowman, PlanetLab: an overlay testbed for broad-coverage services, ACM SIGCOMM Computer Communication Review 33 (3) (2003) 3–12.

[9] M. Colajanni, P. Yu, V. Cardellini, Dynamic load balancing in geographically distributed heterogeneous web servers, in: Proceedings 18th International Conference on Distributed Computing Systems, 1998, IEEE, 2002, pp. 295–302.

[10] M.D. De Assunção, R. Buyya, Performance analysis of multiple site resource provisioning: effects of the precision of availability information, in: Proceedings of the 15th International Conference on High Performance Computing, HiPC'08, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 157–168.

[11] M. De Assunção, R. Buyya, S. Venugopal, InterGrid: a case for internetworking islands of Grids, Concurrency and Computation: Practice and Experience 20 (8) (2008) 997–1024.

[12] A. di Costanzo, M.D. de Assuncao, R. Buyya, Harnessing cloud technologies for a virtualized distributed computing infrastructure, IEEE Internet Computing 13 (2009) 24–33. doi:10.1109/MIC.2009.108.

[13] J. Fontán, T. Vázquez, L. Gonzalez, R.S. Montero, I.M. Llorente, OpenNebula: the open source virtual machine manager for cluster computing, in: Open Source Grid and Cluster Software Conference—Book of Abstracts, San Francisco, USA, 2008.

[14] L. Gong, X.-H. Sun, E.F. Watson, Performance modeling and prediction of nondedicated network computing, IEEE Transactions on Computers 51 (2002) 1041–1055.

[15] C. Grimme, J. Lepping, A. Papaspyrou, Prospects of collaboration between compute providers by means of job interchange, in: Job Scheduling Strategiesfor Parallel Processing, Springer, 2008, pp. 132–151.

[16] L. He, S. Jarvis, D. Spooner, X. Chen, G. Nudd, Dynamic scheduling of parallel jobs with QoS demands in multiclusters and Grids, in: Proceedings of the 5th IEEE/ACM International Workshop on Grid Computing, IEEE Computer Society, 2004, pp. 402–409.