Future Generation Computer Systems 29 (2013) 488–496

Contents lists available at SciVerse ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Self-adjusting resource sharing policies in Federated Grids

Katia Leal *

Universidad Rey Juan Carlos, Departamento de Sistemas Telemáticos y Computación, Escuela Superior de Ingeniería de Telecomunicación, Camino del Molino SN, 28943 Fuenlabrada, Madrid, Spain

ARTICLE INFO

Article history:
Received 14 March 2012
Received in revised form
6 July 2012
Accepted 14 July 2012
Available online 20 July 2012

Keywords:
Self-adjusting
Resource sharing
Performance model
Decentralized
Non-coordinated
Federated Grids

ABSTRACT

The majority of non-coordinated decentralized meta-schedulers in a Federated Grid perform scheduling strategies without taking into account resources' current load or specific resource owners' internal demands, leading to suboptimal schedules. Clearly, these policies increase the number of job migrations, the number of messages generated per re-scheduled job, and also the application makespan. The main purpose of the present study is to analyze the effect of applying self-adjusting resource sharing policies to previously proposed performance based scheduling strategies. For example, when a resource is near saturation or has an internal peak demand, it can decide not to accept new external jobs. On the other hand, when a job owner receives the previous action, it can decide not to submit temporally more jobs to that resource. In this way, the proposed self-adjusting resource sharing policies save time and communication bandwidth by reducing the number of jobs migrations, and thus, avoiding the generation of the corresponding messages per re-scheduled job. At the same time, the new resource sharing strategies improve application makespan and resource performance objective functions while maintaining infrastructure owners complete autonomy.

1. Introduction

A Federated Grid is characterized by allowing resource sharing among several grid infrastructures of different types, that belong to different administrative domains and are controlled by domain specific resource management policies. Thus, we can obtain a powerful resource by summing up the different participating infrastructures. However, resource management and application scheduling over a federated grid is quite complicated due to large system size, infrastructures heterogeneity, domain specific policies and dynamic environment. To solve the scheduling problem, we have previously presented a generic decentralized model that situates a meta-scheduler on the top level of the system architecture. In contrast to local schedulers and workload managers, which possess complete knowledge of system state and user requests, meta-schedulers do have general information about the entire Federated Grid. This is why we cannot apply fine-grained techniques that are more suitable to local schedulers or workload managers that completely control the resources. Instead, this obstacle is overcome with light, decoupled, and coarse-grained techniques. In Section 2 we briefly review our proposed decentralized model [1] and a performance based algorithm [2] for scheduling independent tasks in such a system.

In the same way, the majority of brokering approaches solve the Federated Grid scheduling problem based on a decentralized model. As mentioned in [3], these systems can be further classified into two categories: coordinated and non-coordinated. These solutions and their limitations are discussed in Section 3.

Section 4 investigates the effects of applying self-adjusting resource sharing policies to the previously proposed performance based mapping strategy. In fact, we want to demonstrate that our sharing policies can contribute to reduce the number of jobs migrations, and thus saving time and improving application makespan while keeping Federated Grid participating infrastructures owners autonomy.

In Section 5 we present the simulation model for evaluating the performance of our resource sharing policies. This simulation model, *FederatedGridSim*, represents a whole Federated Grid consisting in several grid infrastructures. FederatedGridSim allows us to simulate a common scenario on which the use of resources from the federation by a small organization when internal resources are not sufficient is totally transparent.

Simulations and the experimental setup are described in Sections 5 and 6. Three different experiments are performed to capture the behavior of the proposed resource sharing strategies. In all cases, the environmental setup is exactly the same, the experiments only differ in the resource sharing policies implemented at the meta-scheduler level. In addition, applications are submitted in specific simulation time instants that correspond with different infrastructure saturation levels, or what is the same, with different numbers of free computing elements. Thus, the three experiments

* Tel.: +34 91 488 87 65; fax: +34 91 488 70 49.
E-mail address: katia.leal@gmail.com.

0167-739X/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.future.2012.07.005

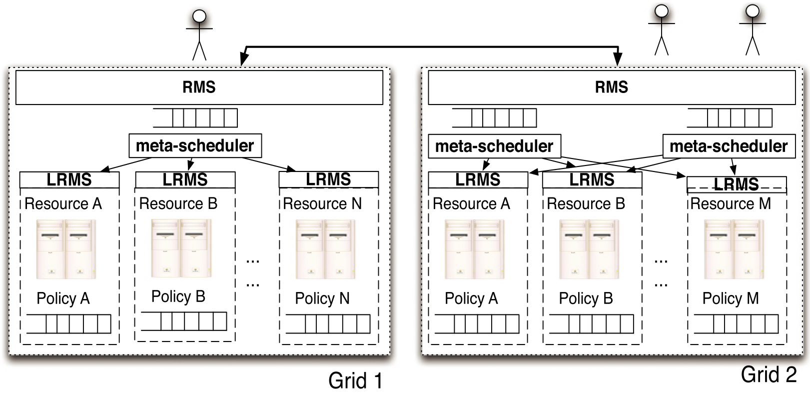

Fig. 1. Decentralized model.

coexist in a *low saturation*, *medium saturation* and *high saturation* level scenario.

Section 7 provides simulation results of the three experiments. We analyze the makespan achieved (completion time), the objective function of each algorithm (number of jobs submitted to each participating infrastructure), number of jobs migrated and the behavior of the different algorithm through graphics.

Finally, Section 8 summarizes the benefits of applying self-adjusting resource sharing policies to performance based scheduling strategies in Federated Grids. Guidelines for our current research activities are also provided.

2. Performance based scheduling strategies

In our previous work [1] we proposed the decentralized model depicted in Fig. 1 as an alternative to centralized, application-centric or ad-hoc solutions to the scheduling problem in Federated Grids. The model puts a meta-scheduler at the top level of each grid infrastructure, over the workload managers. This new layer presents a queue on which jobs have to wait to be scheduled. However, the experimental results demonstrated that, even with the delays introduced by the different queues jobs have to pass through, this model is still very efficient. Instead of having a unique centralized global scheduler to map the jobs of the distinct Grids, each one has its own meta-scheduler. The aim of the mapping strategy implemented on the meta-scheduler at each Grid infrastructure of the Federated Grid was to reduce the makespan of its applications and to increase the performance of its own Grid infrastructure.

Subsequent investigations [2] considered three algorithms for this model: ARAE (All Resources Are Equal), PT-AR (Per Type-All Resources), and PT-RR (Per Type-Resources with Results). These algorithms are based on a performance model [4] that allows to parameterize and compare different grids.

The performance model is based on the parameters proposed by Hockney and Jesshope [5]: r_∞ (asymptotic performance), which is the maximum rate of performance in tasks executed per second, and $n_{1/2}$ (half-performance length), which is the number of tasks required to obtain the half of the asymptotic performance. These parameters, which characterize the average behavior of the system, are obtained by means of a first order adjustment to the function representing the number of tasks completed as a function of time.

With every completed job, the corresponding algorithm recalculates the performance of the individual Grid infrastructure where it was executed, by updating its r_∞ and $n_{1/2}$ values [4]. Finally, to characterize the performance of the whole Federated Grid, we use an *aggregation* or *federation* model [6]. This allows us to calculate the *objective* or number of jobs to submit to each of the participating Grid infrastructures using integer linear programming.

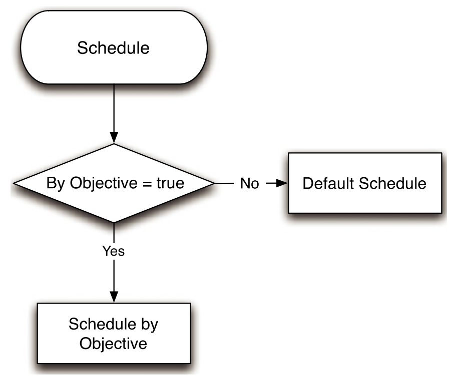

Fig. 2. The scheduling process.

All the strategies follow the same scheduling process, as depicted in Fig. 2. If "By Objective" is true, this means that the algorithm has enough information to determine the performance of the Federated Grid, so it can "Schedule by Objective". By using the performance of the Federated Grid, the "Schedule by Objective" process calculates the objective or number of jobs to submit to each participating Grid resource. In contrast, if "By Objective" is false, the algorithm does not have enough information to estimate the performance of the Federated Grid, so it has to apply a "Default Schedule". The "Default Schedule" process will first map as many jobs as possible to internal resources, and then it will apply a fair division among the external resources.

However, the proposed strategies present small variations in the implementation of the "Schedule by Objective" process. In fact, each policy imposes their own criteria to determine when there is enough information to determine the performance of the Federated Grid. In other words, each algorithm determines the number of *results* (jobs already completed by a particular Grid resource) needed to characterize a particular Grid infrastructure.

In particular, the PT-RR algorithm schedules as follows:

- *"By Objective"*: PT-RR groups results *Per Type* (PT) of resource. Thus, instead of waiting for two results of every resource forming the Federated Grid, PT-RR waits, at least, for two results of any internal and any external resource.
- *"Default Schedule"*: PT-RR applies fair division.
- *"Schedule by Objective"*: the number of jobs to submit to each type of resource (internal or external) is proportional to its performance. The internal and external performance are calculated respectively based on the internal and external resources that can provide results (jobs already completed). Thus, the algorithm knows that N jobs will be submitted to