Sustainable Computing: Informatics and Systems 9 (2016) 33–41

Contents lists available at ScienceDirect

Sustainable Computing: Informatics and Systems

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

Energy efficient scheduling strategies in Federated Grids[☆]

Katia Leal*

Universidad Rey Juan Carlos, Departamento de Tecnologías de la Información y las Comunicaciones, Escuela Superior de Ingeniería de Telecomunicación, Camino del Molino SN, 28943 Fuenlabrada, Madrid, Spain

ARTICLE INFO

Article history:
Received 2 March 2015
Received in revised form 22 July 2015
Accepted 27 August 2015
Available online 3 September 2015

Keywords:
Scheduling
Resource sharing
Energy efficient
Power consumption
Communication cost
Federated Grid

ABSTRACT

The impact of energy consumption and CO_2 emissions in high-performance grid computing is no longer ignored. So far, most efforts have focused on energy efficiency measures of grid infrastructures's underlying hardware and on direct reduction of CO_2 emissions. Thus, in the field of scheduling and resource allocation in computational grids some algorithms have been proposed for selecting grid infrastructures in energy- and CO_2-aware manner or based on hardware power consumption. However, these hardware energy efficiency measures are not suitable at meta-scheduler level. Instead, this paper presents *software energy efficiency measures* that reduce computing power and communication bandwidth, and hence reduce algorithm's total communication cost and power consumption. The main idea is to restrict to the maximum the number of needed messages per job scheduled. Thus, we proposed performance based scheduling policies and self-adjusting resource sharing strategies that accomplish with applications makespan and resources performance while reducing the total amount of computation power and communication bandwidth required by the algorithm itself.

1. Introduction

Although in 2007 the United States Environmental Protection Agency [1] predicted that worldwide energy use for servers and data centers would double from 2005 to 2010, an independent report [2] issued in 2011, said the actual increase was only about 56%. In the United States, the increase was even smaller, coming in at 36%. All in all, data center power consumption continues to rise, so now more than ever we must take steps to reduce this increase. Thus, the impact of the increase in energy consumption is clear. On one hand, in 2006, U.S. servers and data centers consumed around 61 billion kilowatt hours (kWh) at a cost of about 4.5 billion U.S. Dollars [3]. On the other hand, the 2008 "Impacts of Information and Communication Technologies on Energy Efficiency" report [4] for the European Commission says Information and Communication Technologies (ICT) accounted for 2% of Europe's greenhouse gas (GHG) emissions in 2005. One of its conclusions is that by 2020 the figure could reach nearly 4% with a "Business As Usual" scenario and nearly 3% with an energy-saving scenario (transition to energy-efficient solutions). Thus, one of the five ambitious objectives set by the European Union to be reached by 2020 is climate/energy [5]. In a changing world, they want the EU to become smart and sustainable.

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. As a result, 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. Typically, the main objective of scheduling and resource sharing strategies in Federated Grids was to reduce the overall applications's makespan and to enhance infrastructures's performance. Now, these policies have also to accomplish a new requirement: reduce energy consumption. In doing so, these algorithms will be able to decrease financial and environmental cost. Our approach 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, our meta-scheduler will have general information about the entire Federated Grid. This is why we cannot try to 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. Various algorithms that follow this idea and

---

[☆] This paper has been partially funded by Spanish Ministerio de Economía y Competitividad under project TIN2013-41819-R, by the Regional Government of Madrid (CM) under project Cloud4BigData (S2013/ICE-2894) cofunded by FSE & FEDER and by the European Commission under projects NUBOMEDIA FP7-ICT-2013-1.6 (GA: 610576) and FI-CORE FP7-2013-ICT-FI (GA-632893).

* Tel.: +34 914888765.
E-mail address: katia.leal@gmail.com

http://dx.doi.org/10.1016/j.suscom.2015.08.002
2210-5379/

that could be hosted in this meta-scheduler have been thought to restrict to the maximum the number of submitted messages per job scheduled.

As we will see in Section 2, in the field of scheduling and resource allocation in computational grids some algorithms have been proposed for selecting grid infrastructures in energy- and CO_2-aware manner or based on hardware power consumption. That is, those approaches have been designed to schedule based on grid infrastructures hardware characteristics and so, meta-schedulers need to collect information from both grid resource sites and users, thus increasing the amount of messages required to schedule a job. Also, some of these approaches claim that reducing energy consumption necessarily causes low performance. For instance, the ATLAS experiment at the LHC at CERN developed distributed analysis service that was used for more than 1400 users that submitted jobs and run a total of more 1 million jobs every week in the year 2011 [6]. Thus, it is extremely important to develop smart algorithms that require less computational power to make scheduling decisions per job submitted to the Federated Grid, like the decentralized model for performance-based scheduling proposed in Section 3. Also, we need resource sharing policies to save communication bandwidth by reducing the number of jobs migrations, and thus, avoiding the generation of the corresponding new messages per re-scheduled job. Section 4 explains how previously proposed self-adjusting resource sharing policies can achieve this goal. Simulations and results are described in Section 5. This section analyzes results of four experiments, including a naive scheduler based on the GridWay [7] meta-scheduler as a benchmark. Besides checking the makespan achieved, we mainly take notice of the amount of messages required to schedule a job and of the number of jobs that has to be re-scheduled due to the lack of energy efficient resource sharing strategies. Also, to better understand the implications of the reduction in the number of submitted messages per job scheduled, we estimate the amount of grams of CO_2 created by the proposed algorithms. Finally, Section 6 summarizes how the use of these *software energy efficiency measures* for scheduling in Federated Grids can contribute to reduce the amount of energy required while finding an optimal makespan.

2. Related work

The scheduling problem has been around over the past few decades [8], focusing on homogeneous as well as on heterogeneous systems. Thus, in the field of planning and scheduling there have been movements from isolated multi-host scenarios to open large scale infrastructures with several computational resources. However, in all these cases the main objective was to reduce the overall execution time and to increase the performance of the resources. Now, we ask for those algorithms not only to be energy aware, but also smart and sustainable.

So far, there have been efforts in two directions, namely, *hardware energy efficiency measures* and direct reduction of CO_2 emissions. This article proposes a new way to reduce computational consumption by means of energy efficient scheduling and resource sharing policies that require less computing power and less communication bandwidth.

The majority of brokering approaches solve Federated Grid scheduling and sharing problems based on a decentralized model. As mentioned in [9], these systems can be further classified into coordinated decentralized and non-coordinated decentralized. Thus, *decentralized non-coordinated schemes* perform scheduling related activities independent of the other schedulers in the system [10,11]. In fact, they directly submit their jobs without taking into account the current load or resources owner's internal necessities. Thus, these approaches produce sub-optimal schedules that can lead to coexistence problems in the Federated Grid. Although, our solution can also be classified as non-coordinated decentralized, the main axis of our proposed algorithms is that they consider the *performance* of the infrastructures forming the Federated Grid, not only their *state*. So, any changes in infrastructures behavior will be reflected in their performance and will be considered in order to determine the number of jobs to submit to them. At the same time, our solution does not strongly depend on resource discovery services. This is important since, as pointed out by Ranjan et al. [12], centralized and hierarchical information services (such as R-GMA [13] and MDS-2,3,4 [14]) have several design limitations including not only single point of failure and lack of scalability, but what is more important, high network communication cost and computational power to serve queries. Furthermore, the studies of scalability conducted by Zhang et al. [15] suggest that the information service has to be duplicated on dedicated machines as the number of users grows, which also means an increase in energy consumption. On the other hand, *decentralized coordinated scheduling schemes* negotiate resource conditions with the other application level schedulers in the system. In this case, the negotiation may lead to a large number of messages generated per job before being scheduled. Negotiation among all the participants can be based on the well known coordination protocol Contract Net [16]. This protocol have been widely used for Grid scheduling, as shown by several works [17–19]. However, negative effects of Contract Net communication protocol on job execution time in a multi-agent Grid computing system also have been studied [20]. In their work, authors claim that the main overhead in the system is due to the execution of the Contract Net protocol. They also describe the exchange of messages required for each request. Thus, the client first sends out call for-proposal requests to all agents in the system, then waits for bids in response until a deadline. After the deadline, it evaluates the proposals received from responding agents, selects the winner and then sends an accept message to the selected agent and reject messages to the others. Finally, the client waits for the result. It is assumed that if no bid arrives by the deadline, the process is repeated from the start, which means an uncertain number of messages per request. In contrast, since our sharing strategies are self-adjusting, they do not need to negotiate with the rest of participating infrastructures, and thus, they do not flood the network with negotiation messages. Also, we do not need to deploy and execute agents or specialized software across the different infrastructures. Therefore, our policies save both computation time and communication bandwidth.

Among hardware energy efficiency measures, *cooling* means 45% of the energy used in data centers worldwide [21] and can cause different Power Usage Effectiveness (PUE) [22] values between centers. Thus, do not cool or at least, choose a data center geographical location that reduces cool necessities. By *renting* the excess of computational resources provider companies can justify the hardware installed and consumer companies can reduce hardware investment costs. One example of this "Cloud Computing" scheme is Amazon's Elastic Computing Cloud (EC2) [23]. There are other ways to benefit the computational power of others, for example, by *sharing* the otherwise wasted idle time of our computers to perform useful calculations for some of several "Volunteer Computing" projects. You can choose one of the projects under BOINC [24], an open source middleware system for volunteer and grid computing. BOINC was originally developed to support the SETI@home project before it became useful as a platform for other distributed applications in areas as diverse as mathematics, medicine, molecular biology, climatology, and astrophysics. Although, some of these measures do not reduce CO_2 emissions, they try to maximize hardware resources utilization. Also, it is important to notice that all these techniques can be used in conjunction with other policies, like the ones proposed in this paper.