



## Exploring decentralized dynamic scheduling for grids and clouds using the community-aware scheduling algorithm

Ye Huang<sup>a,d,\*</sup>, Nik Bessis<sup>b,c</sup>, Peter Norrington<sup>b</sup>, Pierre Kuonen<sup>d</sup>, Beat Hirsbrunner<sup>a</sup>

<sup>a</sup> Department of Informatics, University of Fribourg, Switzerland

<sup>b</sup> Department of Computer Science and Technology, University of Bedfordshire, UK

<sup>c</sup> School of Computing and Mathematics, University of Derby, UK

<sup>d</sup> Department of Information and Communication Technologies, University of Applied Sciences of Western Switzerland (Fribourg), Switzerland

### ARTICLE INFO

#### Article history:

Received 12 November 2010

Received in revised form

14 March 2011

Accepted 7 May 2011

Available online 13 May 2011

#### Keywords:

Grid

Cloud

Scheduling

Meta-scheduling

Community-aware scheduling algorithm (CASA)

SmartGRID

### ABSTRACT

Job scheduling strategies have been studied for decades in a variety of scenarios. Due to the new characteristics of the emerging computational systems, such as the grid and cloud, metascheduling turns out to be an important scheduling pattern because it is responsible for orchestrating resources managed by independent local schedulers and bridges the gap between participating nodes. Equally, to overcome issues such as bottleneck, single point failure, and impractical unique administrative management, which are normally led by conventional centralized or hierarchical schemes, the decentralized scheduling scheme is emerging as a promising approach because of its capability with regards to scalability and flexibility.

In this work, we introduce a decentralized dynamic scheduling approach entitled the community-aware scheduling algorithm (CASA). The CASA is a two-phase scheduling solution comprised of a set of heuristic sub-algorithms to achieve optimized scheduling performance over the scope of overall grid or cloud, instead of individual participating nodes. The extensive experimental evaluation with a real grid workload trace dataset shows that, when compared to the centralized scheduling scheme with BestFit as the metascheduling policy, the use of CASA can lead to a 30%–61% better average job slowdown, and a 68%–86% shorter average job waiting time in a decentralized scheduling manner without requiring detailed real-time processing information from participating nodes.

© 2011 Elsevier B.V. All rights reserved.

### 1. Introduction

Job scheduling strategies [1] have been extensively studied in the last few decades within a variety of scenarios, such as manufacturing systems and distributed computation environments. The increasing demand of computation resources has led to new types of cooperative distributed systems, such as the grid [2] and cloud computing [3]. Due to the new characteristics of emerging computational systems, conventional scheduling techniques need to evolve into more complex and sophisticated solutions in order to cover new scheduling constraints, such as heterogeneous resources, variety of job requirement, and dynamic and volatile networks. Furthermore, scheduling techniques allowing jobs to be shared between decentralized sites, virtual organizations (VOs), or

even different grids/cloud providers, have appeared to be a promising approach because of their capability with regards to scalability and flexibility.

Metascheduling, also known as grid scheduling within the context of grid computing, turns out to be an important scheduling scheme because it is responsible for orchestrating resources managed by independent local schedulers and bridges the gap between isolated local computation resource pools. However, current research and implementation work have several crucial constraints and limitations, including: (a) scheduling for serving the hosting node, instead of the entire grid; and (b) assuming the detailed processing information of each participating node, such as the status of local job queue and real-time resource utilization, is known. In order to conquer such issues, a novel scheduling approach is desired to serve the overall grid, instead of each individual node, with a variety of preferred optimization objectives. Furthermore, such an approach is also supposed to work in a decentralized manner and be able to dynamically adapt to the changes in the grid through time.

Since the mid-1990s, the vision of a grid as a computation infrastructure has been widely accepted; numerous grid based

\* Corresponding author at: Department of Informatics, University of Fribourg, Switzerland.

E-mail addresses: [ye.huang@unifr.ch](mailto:ye.huang@unifr.ch), [huangye177@gmail.com](mailto:huangye177@gmail.com) (Y. Huang), [nik.bessis@beds.ac.uk](mailto:nik.bessis@beds.ac.uk) (N. Bessis), [peter.norrington@beds.ac.uk](mailto:peter.norrington@beds.ac.uk) (P. Norrington), [pierre.kuonen@hefr.ch](mailto:pierre.kuonen@hefr.ch) (P. Kuonen), [beat.hirsbrunner@unifr.ch](mailto:beat.hirsbrunner@unifr.ch) (B. Hirsbrunner).

resource sharing infrastructures such as Grid5000 [4], TeraGrid [5], D-Grid [6], EGEE [7], PlanetLab [8], and NorduGrid [9] have been established in different countries and continents serving both for production work and scientific research. Meanwhile, an obstacle has emerged since these grids are established for different purposes and work in isolation from each other. Some pilot work [10,11] has already observed that the next natural step is to enable interoperation between multiple grids, in order to serve much larger scientific communities and enhance the overall performance of the joint grids. Regarding nontrivial issues such as bottleneck, single point of failure, and impractical administrative management, which are normally led by conventional centralized or hierarchical design, these could get worse in an inter-operational grid-based infrastructure; the desire for a complete decentralized design has increased dramatically.

Meanwhile, research on the characteristics and performance of groups of jobs in grids [12] has shown that 70% of jobs, which consumed 80% of resource processing time, are submitted in a batch pattern. In other words, most jobs are submitted by specific batch engines with a very short interval time between each other. In this case, users of grids normally submit many jobs as a single batch with a single runtime estimation, instead of specifying estimate processing time for each individual job. Consequently, scheduling algorithms which rely on job estimate processing time, such as shortest job first (SJF), longest job first (LJF), and backfilling variants, are severely affected. Even concerning a centralized grid wherein each node maintains complete global information and is interconnected upon a static and stable network, the widely adopted backfilling series algorithms [13,14] can still lead to complex dynamic problems. Related research [15] has shown jobs requesting short processing time and few processors are likely to find “holes” easier in heavily loaded systems, which makes the prediction of future system performance as static even harder. In this case, a novel scheduling heuristic supporting dynamic rescheduling through time has shown its great importance, which is also covered in this paper.

As the extension to some pilot work [16], we propose a novel decentralized dynamic scheduling approach named the community-aware scheduling algorithm (CASA). In this work, based on the previously proposed two-phase scheduling protocol, a set of heuristic algorithms are designed to efficiently distribute jobs amongst participating nodes without asking for detailed node real-time processing information nor control authorities of remote nodes. The remainder of the paper is organized as follows: the related work in terms of decentralized scheduling is given in Section 2. The problem statement and algorithm principle is presented in Section 3, followed by the detailed discussion of each heuristic algorithm in Section 4. Section 5 introduces the experiment configuration for the algorithm evaluation, whilst Section 6 discusses the results observed in this experiment. Finally, Section 7 presents the conclusion and some insights to future work.

## 2. Related work

Metacomputing, the term firstly introduced by Smarr and Catlett [17], is widely accepted in the field of grid computing [2] to describe the computational pool formed from resources of different participating nodes. In general, metascheduling solutions are classified into three categories, namely the centralized, hierarchy, and decentralized schemes [18,10,15]. Specifically, the decentralized metascheduling scheme allows each node to own a metascheduler to receive job submissions originated by local users, and to assign such jobs to the local resource management system, i.e., local scheduler, of the node. Meanwhile, metaschedulers of different nodes are capable of exchanging information and sharing jobs between each other in order to balance the resource load

amongst participating nodes. The nature of the decentralized scheme brings better scalability compared to other scheduling schemes, but leads to the issue of efficiency and overhead on the other hand.

NWIRE (Net-Wide-Resources) [19] is a brokerage and trading based metacomputing scheduling architecture. On the top logical level, NWIRE consists of a set of MetaDomains, wherein each MetaDomain is controlled by one MetaManager. By taking into consideration several scheduling properties, the NWIRE relies on a market mechanism [20] to trade resources between domains via the MetaManager, wherein a description of the requests and objectives appears to be a key element for the job allocation. In addition, NWIRE provides high flexibility in terms of fault tolerance because the failure of a single trader will not affect the whole metascheduling system.

The K-Distributed and K-Dual Queue Models [15,21] propose a distributed scheduling algorithm which redundantly distributes jobs to different sites simultaneously, instead of only sending jobs to the most lightly loaded sites. The K-Distributed model enables each metascheduler of each site to distribute their jobs to the K least loaded sites, whereby such jobs will be scheduled by all K sites respectively. The K-Dual model works on the K-Distributed model and gives priority to jobs originated by the local sites. Jobs transferred from remote nodes will be executed only if they do not adversely affect the start time of queued jobs which are originated from the local sites.

InterGrid [22] is a cross-grid cooperation architecture composed of a set of InterGrid Gateways (IGGs) responsible for managing peering arrangements between grids. InterGrid promotes interlinking of islands grids through peering arrangements to enable inter-grid resource sharing, and provides a scalable structure allowing grids to interconnect with each other and to grow in a sustainable way. Although the structure of the overall InterGrid ecosystem is hierarchical, the InterGrid Gateways employed upon the top of each participating grid are distributed in a decentralized manner. Each IGG is aware of the agreements with other IGGs, and is capable of enabling resource allocation across multiple grids with pluggable policies. The InterGrid/IGG relies on external decentralized approaches, such as a self-organizing economic model [23], to meet its design objectives, including incentive-oriented peering arrangements, decentralized resource management, and reservation and brokering across grids.

Delegated Matchmaking (DMM) [10] is a decentralized approach for grid inter-operation by temporarily binding resources from remote sites to the local environment. First, the DMM leverages a hierarchical architecture in which nodes represent computing sites, and nodes of the same hierarchical level are allowed to inter-operate to form a completely decentralized network. Second, the DMM employs two independent policies named the *delegation algorithm* and *local requests dispatching policy*, to disseminate resource requests within the established network. By delegating resources instead of the traditional job delegation way, the DMM aims to lower the administrative overhead of managing user/group accounts on each site during the inter-grid operations.

Grid-Federation [24,25] is a metascheduling framework which highlights a bid-based SLA contract negotiation model. Benefitting from the market-based SLA coordination mechanism, the Grid-Federation framework allows resource owners to have finer control over the resource allocation. An SLA is the agreement negotiated between a metascheduler, entitled the Grid Federation Agent (GFA), and the LRMSs of the local sites in terms of acceptable job QoS constraints, such as job response time and budget spent. Furthermore, the contract net protocol [26] based SLA bids are restricted with a certain expiration time, and a variety of economic parameters such as setting price, user budget and deadline. A greedy backfilling heuristic is also proposed for application

متن کامل مقاله

دریافت فوری ←

**ISI**Articles

مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات