Information Processing Letters 115 (2015) 838–844

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Inefficiency of the Nash equilibrium for selfish machine covering on two hierarchical uniform machines

Yong Wu [a,b,*], T.C.E. Cheng [b], Min Ji [c]

[a] Department of Fundamental Education, Ningbo Institute of Technology, Zhejiang University, Ningbo 315100, PR China
[b] Department of Logistics and Maritime Studies, The Hong Kong Polytechnic University, Kowloon, Hong Kong
[c] School of Computer Science and Information Engineering, Contemporary Business and Trade Research Center, Zhejiang Gongshang University, Hangzhou 310018, PR China

ARTICLE INFO

Article history:
Received 4 March 2014
Received in revised form 17 July 2014
Accepted 10 June 2015
Available online 18 June 2015
Communicated by S.M. Yiu

Keywords:
Scheduling
Hierarchy
Nash equilibrium
Price of anarchy
Price of stability

ABSTRACT

This paper studies the selfish scheduling game on two hierarchical uniform machines where the jobs are correspondingly classified into two hierarchical classes. The cost of a job is defined as the completion time of the machine to which it is assigned. Each selfish job is interested in minimizing its own cost, while the game seeks to meet the social objective of maximizing the machine cover. We obtain the (strong) price of anarchy and the (strong) price of stability as functions of the ratio between the speeds of the two machines s. We show that all the derived bounds are tight for any value of s, thus completely solving the problem of measuring the inefficiency of the Nash equilibrium on two hierarchical uniform machines.

1. Introduction

In this paper we study the selfish machine covering game on two hierarchical uniform machines. Bar-Noy et al. [3] first study hierarchical scheduling. It is a common practice in the service industry that differentiated services are provided to customers based on their entitled privileges, which are assigned according to their classes in the service hierarchy. While hierarchy is a subjective concept, it is often put into practice in terms of different levels of access privilege to service capacity. Hierarchical scheduling has many applications, e.g., in the service industry, computer systems, hierarchical databases and hierarchical network routing etc. In traditional scheduling research, it is usually assumed that there is a centralized controller that can gather all the information from users and coordinate the behaviors of all the users for the sharing of common resources. However, in many real-world distributed scheduling settings like (hierarchical) network routing, such centralized coordination is unlikely to exist. Therefore, in such a distributed scheduling environment (i.e., a scheduling game), each job is a player that is only interested in achieving its own goal without any concern for the global optimal outcome for the game. Recently, researchers have begun to address the issue of performance degradation due to a lack of coordination in such selfish scheduling games. Many solution concepts are used to study the behavior of selfish jobs in scheduling games. The best known concepts are probably the *Nash equilibrium* (NE) and the *strong Nash equilibrium* (SE). Following recent research on games relating to scheduling and routing problems [16,17], we study the NE and SE for the machine covering game on two hierarchical uniform machines. Compared with other studies in the related literature, our study has the novelty that both the machines and jobs have hierarchies.

* Corresponding author at: Department of Fundamental Education, Ningbo Institute of Technology, Zhejiang University, Ningbo 315100, PR China.

E-mail addresses: wuyong@nit.zju.edu.cn (Y. Wu), edwin.cheng@polyu.edu.hk (T.C.E. Cheng), jimkeen@163.com (M. Ji).

http://dx.doi.org/10.1016/j.ipl.2015.06.005

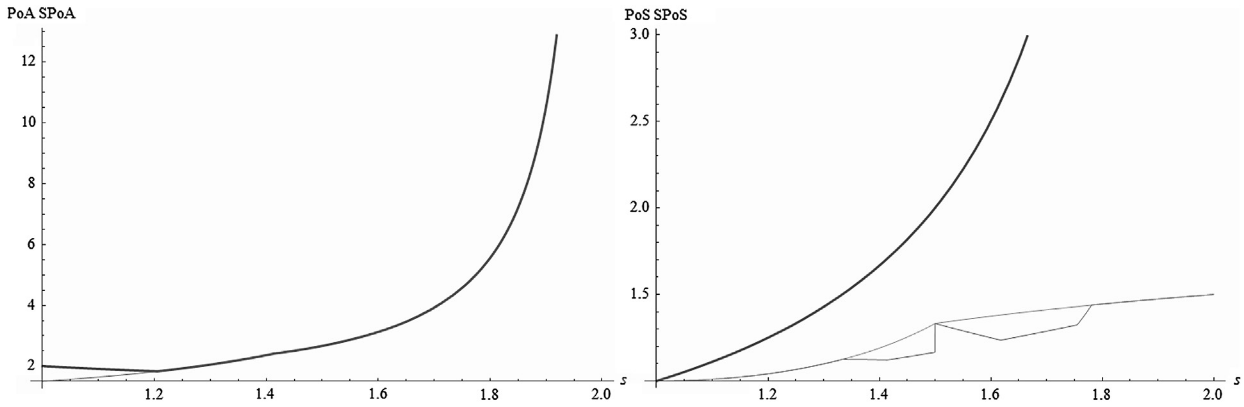

Fig. 1. *Left:* The PoA and SPoA on hierarchy machines (*thick line*), and the PoA and SPoA on non-hierarchy machines (*thin line*). *Right:* The PoS and SPoS on hierarchy machines (*thick line*), and the PoS and SPoS on non-hierarchy machines (*thin line*).

We are given two uniform machines M_1 and M_2, and a set $\mathcal{J}$ of n selfish independent jobs $J_1, J_2, \ldots, J_n$. The speed of M_i is s_i. W.l.o.g., we assume that $s_1 = 1$ and $s_2 = s > 0$. Each selfish job $J_i = (p_i, g_i)$ is a player of the game, where p_i is the size of J_i and $g_i \in \{1, 2\}$ is the class of J_i. Machine M_k has a certificate $g(M_k) = k$, $k = 1, 2$, associated with it. The job J_i is allowed to be executed on machine M_k only when $g_i \geq g(M_k)$. Thus, in this selfish scheduling game, the jobs in G_1 have no choice except to choose machine M_1. Each job may select a machine for processing to minimize its own completion time, which is equal to the completion time of the selected machine. The choices of all the jobs determine a schedule. An NE is a schedule under which no job has a unilateral incentive to switch to another machine. An SE is a schedule under which there exists no coalition of the jobs such that the completion time of each job in the coalition is reduced by its migration to a different machine. By definition, an SE is also an NE, but the reverse is not necessarily true. It has been proved that an NE and an SE always exist for the above scheduling game [1,11].

In a selfish scheduling game, each job is interested in achieving its own goal only without any concern for the global optimal solution for the game. However, the game has a social objective to fulfill, which is normally motivated by concerns for quality of service and fair resource allocation. We study the game with the social objective of maximizing the minimum completion time of all the machines, which is usually referred to as the machine covering problem by scheduling researchers [6,8,20]. An NE or SE is not always a social optimum due to a lack of central coordination. Thus, it is important to quantify the efficiency loss due to the selfish behaviors of the players. There are two popular measures of inefficiency: (i) the *price of anarchy* (PoA) of a game, which is defined as the performance ratio between the social objective of an optimal solution and that of the worst NE (an NE with the worst overall social cost Nash equilibria), and (ii) the *price of stability* (PoS) of a game, which is defined as the performance ratio between the social objective of an optimal solution and that of the best NE (an NE with the best overall social cost Nash equilibria) [2,16]. The *strong price of anarchy* (SPoA) and the *strong price of stability* (SPoS) are defined similarly, except that the NE is replaced by the SE [1]. By definition, it is evident that PoA $\geq$ SPoA $\geq$ SPoS $\geq$ PoS.

We consider the selfish scheduling game on hierarchical machines with the social objective of maximizing the machine cover. This goal is different from the regular makespan minimization goal (i.e., minimizing the maximum completion time of all the machines), which has been extensively studied in the game theoretic context. For the selfish scheduling game on hierarchical machines with the social objective of minimizing the makespan, the exact PoA is $\frac{3}{2}$ for the case with two hierarchical identical machines [15]. Epstein [7] derive the exact values of the PoA and SPoA for the case with two hierarchical uniform machines. If the classes of all the jobs are the same or the number of hierarchical classes is one, then the scheduling game on hierarchical uniform machines becomes the scheduling game on uniform machines. For the scheduling game with the social objective of minimizing the makespan, the PoA and SPoA are $\Theta(\frac{\log m}{\log\log m})$ and $\Theta(\frac{\log m}{(\log\log m)^2})$ for the case with m uniform machines, respectively [5,13], while the PoA is equal to $\frac{1+\sqrt{5}}{2}$ and 2 when m is 2 or 3 [12]. Epstein and Stee [10] analyze the PoA and SPoA for a special case of m uniform machines. The PoA and SPoA are both $\frac{2m}{m+1}$ for the case with m identical machines [1,14,18], and the PoS and SPoS are both 1 [1]. Epstein et al. [8] first study the inefficiency of equilibria of machine covering games on uniform and identical machines. Chen et al. [4] improve the results for the case with identical machines, where the PoS is exactly 1 and the overall PoA is exactly 1.7 for the case with m identical machines. Epstein et al. [9] analyze the PoA and PoS for a special case of m uniform machines. Epstein et al. [8] and Tan et al. [19] completely solve the problem of measuring the inefficiency of equilibria on two uniform machines (see Fig. 1). They obtain the exact PoA, SPoA, PoS, and SPoS as functions of the ratio between the speeds of the two machines s. Now we turn to the case where there are two machines each having a different certificate and where the jobs belong to different classes. We obtain the exact PoA, SPoA, PoS, and SPoS as functions of the ratio between the speeds of the two machines s. For the machine covering game on two hierarchical uniform ma-