Information Processing Letters 113 (2013) 251–254

Contents lists available at SciVerse ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Coordination mechanism for selfish scheduling under a grade of service provision ☆

Li Guan, Jianping Li *

Department of Mathematics, Yunnan University, Kunming 650091, PR China

ARTICLE INFO

Article history:
Received 11 November 2011
Received in revised form 17 January 2013
Accepted 18 January 2013
Available online 23 January 2013
Communicated by J. Xu

Keywords:
Scheduling
Selfish scheduling
Grade of service
Coordination mechanism
Makespan
Price of anarchy

ABSTRACT

In this paper, we study the problem of selfish scheduling game under a grade of service provision, where all machines and all jobs are labeled with the different grade of service (GoS) levels such that a job J can be assigned to execute on machine M only when the GoS level of machine M is not higher than the GoS level of job J. We consider two coordination mechanisms for this selfish scheduling game: the makespan policy and the LG-LPT policy. For the first mechanism, we show that the price of anarchy is exactly $\frac{3}{2}$ for two machines and $\Theta(\frac{\log m}{\log \log m})$ for m ($\geqslant 3$) machines, respectively. For the second mechanism, we point out that the price of anarchy is $\frac{5}{4}$ for two machines and $2 - \frac{1}{m-1}$ for m ($\geqslant 3$) machines, respectively, and we finally analyze the convergence to a Nash equilibrium of the induced game.

1. Introduction

The game theory is to study some situations concerning selfish agents who are interested in achieving their individual goals, as well as in opposing to obtain a global optimum. The agents act selfishly until reaching some equilibria. In general, the social optimum is not typically obtained. Quantifying the efficiency loss due to selfish behavior is an important research interest in such settings. The price of anarchy (POA, for short), which was first proposed by Koutsoupias and Papadimitriou [8], is the most popular measure to quantify the inefficiency of equilibrium. Precisely, the price of anarchy of a game is defined as the ratio between the worst objective function value of an equilibrium of the game and the objective function value of an optimal outcome. We only consider the pure Nash equilibrium in this paper.

How can we reduce the inefficiency of equilibrium in a game? An important approach is defined as a coordination mechanism, which is a local policy that assigns an outcome to each strategy s, where the outcome of such a strategy s is a function of the agents who have chosen the strategy s. The purpose is to lead the independent and selfish choices of the agents to obtain a better result in a socially desired outcome. Obviously, the primary goal of a coordination mechanism for the designer is to guarantee the existence of pure Nash equilibria for the induced game.

The selfish scheduling game problem has been studied extensively in the literatures [1–4,7], which is defined as follows. There are n jobs owned by some independent agents, say $J_1, J_2, \ldots, J_n$, and m machines, say $M_1, M_2, \ldots, M_m$, and some processing times p_{ij}, where p_{ij} indicates the processing time for the job J_i to be executed on the machine M_j, each agent may select a machine to minimize his own completion time. The social objective is to minimize the makespan, *i.e.*, the maximum completion time of m machines. A pure Nash equilibrium is an assignment of these n jobs to be executed on m machines such that no job has an unilateral incentive to switch to

☆ Supported by the National Natural Science Foundation of China [Nos. 10861012, 61063011], the Project of the First 100 High-level Overseas Talents of Yunnan Province IRTSTYN.

* Corresponding author. Tel.: +86 871 65033701; fax: +86 871 65033700.

E-mail addresses: guanli@ynu.edu.cn (L. Guan), jianping@ynu.edu.cn (J. Li).

0020-0190/$ – see front matter
http://dx.doi.org/10.1016/j.ipl.2013.01.014

another machine. A coordination mechanism for this game is a set of scheduling policies, one for each machine, that determines the way to schedule the jobs on that machine. The price of anarchy of a selfish scheduling game is the ratio between the maximum makespan in all pure Nash equilibria and the minimum makespan in all available assignments.

In service industry, the service providers often provide differentiated services to the special customers who are more valued than the regular customers. One simple scheme for providing differentiated services is to label all machines and all jobs with the different grade of service (GoS, for short) levels. A job J can be assigned to execute on a machine M only when the GoS level of machine M is not higher than the GoS level of job J. Thus, when we label relatively higher GoS levels on the jobs of more valued customers, we can provide a better service to them.

The model of a selfish scheduling game under a GoS provision is defined as follows. There are a set of m machines, say $\mathcal{M} = \{M_1, M_2, \ldots, M_m\}$, and a set of n jobs, say $\mathcal{J} = \{J_1, J_2, \ldots, J_n\}$. Each job J_j owned by a selfish agent of the game has a processing time $p(J_j)$, and it is labeled by a GoS level $g(J_j)$. Each machine M_i is also labeled by a GoS level $g(M_i)$. The job J_j is allowed to be executed on the machine M_i only when $g(J_j) \geqslant g(M_i)$. We may assume that all jobs can be scheduled on these m machines.

In this paper, we study the selfish scheduling game under a GoS provision, and we obtain main two results: (1) For the makespan policy, we prove that the price of anarchy is exactly $\frac{3}{2}$ for two machines and $\Theta(\frac{\log m}{\log\log m})$ for m ($\geqslant 3$) machines, respectively; (2) For the LG-LPT policy, we point out that the price of anarchy is $\frac{5}{4}$ for two machines and $2 - \frac{1}{m-1}$ for m ($\geqslant 3$) machines, respectively, and we finally analyze the convergence to a Nash equilibrium of the induced game.

This paper is organized as follows. In Section 2, we consider the makespan policy and then obtain the results in (1); In Section 3, we consider the LG-LPT policy and then obtain the results in (2).

2. Makespan policy

In the makespan policy, each machine processes the jobs assigned in parallel. If a job J_j is assigned to a machine M_i, the completion time of job J_j is equal to the completion time of machine M_i. For any makespan policy, the pure Nash equilibrium of the induced selfish scheduling game always exists [7]. This fact is also true for the selfish scheduling game under a GoS provision since the latter is a special version of selfish scheduling game. We first consider POA for two machines, and we obtain the first result as follows.

Theorem 1. *For two machines, the price of anarchy of the makespan policy is exactly $\frac{3}{2}$ for the selfish scheduling game under a grade of service provision.*

Proof. If the GoS levels of two machines are same or the GoS levels of all jobs are same, our scheduling problem becomes the ordinary identical parallel machine scheduling problem $P\|C_{max}$ [5]. From the earlier work [7], we know the fact that the price of anarchy of the makespan policy for $P\|C_{max}$ is $\frac{4}{3}$ for these two machines, which is less than $\frac{3}{2}$.

Now we consider the state that the two machines are labeled with two different GoS levels meanwhile all jobs are labeled with two different GoS levels, respectively. The set of two machines is $\mathcal{M} = \{M_1, M_2\}$, and for convenience, we may assume that $g(M_1) = 1$, $g(M_2) = 2$. The set of n jobs is $\mathcal{J} = \{J_1, J_2, \ldots, J_n\}$. Let $\mathcal{J}_i = \{J_j \mid g(J_j) = i,\ 1 \leqslant j \leqslant n\}$ and $P_i = \sum_{J_j \in \mathcal{J}_i} p(J_j)$, where $i = 1, 2$. By distinguishing the following two cases, we shall prove that POA is at most $\frac{3}{2}$.

Case 1. $P_1 > P_2$

In this case, the induced game has only one Nash equilibrium, where all jobs in $\mathcal{J}_1$ are assigned to execute on M_1 and all jobs in $\mathcal{J}_2$ are assigned to execute on M_2. This equilibrium schedule is also an optimum schedule. Thus, POA is equal to 1.

Case 2. $P_1 \leqslant P_2$

Let *OPT* denote the optimal value, *i.e.*, the minimum makespan. We have

$$OPT \geqslant \max\left\{ \max_{1\leqslant j\leqslant n} p(J_j), P_1, \frac{P_1 + P_2}{2} \right\}.$$

Let μ be any Nash equilibrium and l_i the completion time of the machine M_i in μ, where $i = 1, 2$. Let $cost(\mu)$ be the makespan of μ, *i.e.*, $cost(\mu) = \max\{l_1, l_2\}$. We consider the two subcases.

(1) $cost(\mu) = l_1$

In this subcase, without loss of generality, we may assume that there is at least one job in $\mathcal{J}_2$ assigned to execute on machine M_1, otherwise we obtain $cost(\mu) = OPT$ and the conclusion follows trivially. Let J^* be any job in $\mathcal{J}_2$ assigned to execute on machine M_1. Suppose that if the job J^* changes its strategy, by moving from machine M_1 to machine M_2, then the completion time of J^* should not be decreased, by the fact that μ is a Nash equilibrium. It follows $l_1 \leqslant l_2 + p(J^*)$. Thus, we obtain

$$2\,cost(\mu) \leqslant l_1 + l_2 + p\big(J^*\big) = P_1 + P_2 + p\big(J^*\big) \leqslant 3OPT$$

which implies $cost(\mu) \leqslant \frac{3}{2}OPT$.

(2) $cost(\mu) = l_2$

In this subcase, let J^* be any job assigned to M_2. Obviously, we have $g(J^*) = 2$. Since μ is a Nash equilibrium, it follows $l_2 \leqslant l_1 + p(J^*)$. Thus, we get

$$2\,cost(\mu) \leqslant l_2 + l_1 + p\big(J^*\big) = P_1 + P_2 + p\big(J^*\big) \leqslant 3OPT$$

which also implies $cost(\mu) \leqslant \frac{3}{2}OPT$.

To sum up, we prove that POA is at most $\frac{3}{2}$ for two machines.

The following example shows that this upper bound $\frac{3}{2}$ on POA is tight.

Suppose that we have $\mathcal{M} = \{M_1, M_2\}$ and $\mathcal{J} = \{J_1, J_2, J_3\}$, where $g(M_1) = 1$, $g(M_2) = 2$, $g(J_1) = 1$, $g(J_2) = g(J_3) = 2$, $p(J_1) = p(J_2) = 1$ and $p(J_3) = 2$. An optimal schedule assigns two jobs J_1 and J_2 to be executed