Theoretical Computer Science 607 (2015) 181–192

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Penalty cost constrained identical parallel machine scheduling problem

Weidong Li [a], Jianping Li [a], Xuejie Zhang [a], Zhibin Chen [b,*]

[a] *Yunnan University, Kunming, 650091, PR China*
[b] *Kunming University of Science and Technology, Kunming, 650500, PR China*

ARTICLE INFO

Article history:
Received 11 November 2014
Received in revised form 30 September 2015
Accepted 2 October 2015
Available online 8 October 2015

Keywords:
Scheduling
Rejection penalty
Approximation algorithms
Polynomial time approximation scheme
Fully polynomial time approximation scheme

ABSTRACT

We consider a version of parallel machine scheduling with rejection. An instance of the problem is given by m identical parallel machines and a set of n independent jobs, with each job having a processing time and a penalty. A job may be accepted to be processed or be rejected at its penalty. The objective of the problem is to partition the set of jobs into two subsets, the subset of accepted and the subset of rejected jobs, and to schedule the set of accepted jobs such that the makespan is minimized under the constraint that the total penalty of the rejected jobs is no more than a given bound. In this paper, we present a 2-approximation algorithm within strongly polynomial time for the problem. We also present a polynomial time approximation scheme whose running time is $O(nm^{O(\frac{1}{\epsilon^2})} + mn^2)$ for the problem. Moreover, for the case where the number of machines is a fixed constant m, our results lead to a fully polynomial time approximation scheme for the problem. Our result is fairly good in the sense that in a reasonable size of jobs, our FPTAS improves previous best running time from $O(n^{m+2}/\epsilon^m)$ to $O(1/\epsilon^{2m+3} + mn^2)$.

1. Introduction

Given a set of machines and a set of jobs such that each job has to be processed on one of the machines, the classical scheduling problem $P \parallel C_{\max}$ is to minimize the makespan. Since Graham [8] designed a classical *list scheduling* (LS for short) algorithm, which is to assign the next job in an arbitrary list to the first available machine, for the problem, this strongly *NP*-hard problem has been widely studied for more than four decades. However, as surveyed in Shabtay et al. [20], in many cases, processing all jobs may not be a good strategy. In some systems, a wise decision may be required in advance to partition the set of jobs into a set of accepted and a set of rejected jobs so that some limited production capacities could be used more efficiently. In this paper we consider a generalized version of classical scheduling problem in which a job may be rejected at a certain penalty, and we call this problem *Penalty cost constrained identical parallel machine scheduling problem*, which was first introduced by Zhang et al. [24]. Given are a set of identical machines $\mathcal{M} = \{M_1, M_2, \ldots, M_m\}$, a set of jobs $\mathcal{J} = \{J_1, J_2, \ldots, J_n\}$ with a processing time (or size) $p_j > 0$ and a penalty $e_j \geq 0$ for each J_j of $\mathcal{J}$, and a bound B. We wish to partition the set of jobs into two subsets, the subset of accepted and the subset of rejected jobs, and to schedule the set of accepted jobs on the m machines such that the makespan is minimized under the constraint that the total penalty of the rejected jobs is no more than the bound B. Following the convention of Graham et al. [9] as well as Lawler et al.

* Corresponding author.
E-mail addresses: weidong@ynu.edu.cn (W. Li), jianping@ynu.edu.cn (J. Li), xjzhang@ynu.edu.cn (X. Zhang), chenzhibin11@gmail.com (Z. Chen).

http://dx.doi.org/10.1016/j.tcs.2015.10.007
0304-3975/

[13], we denote this problem and a special case of this problem where the number of machines is a fixed constant m by $P|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$ and $P_m|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$, respectively.

The problem $P|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$ is a generalized version of the classical scheduling problem $P \parallel C_{\max}$ as we just mentioned, but on the other hand, the problem $P|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$ can be viewed as a special case of the bicriteria problem on unrelated parallel machines as defined in Angel et al. [2], where we are given n independent jobs and m unrelated parallel machines with the requirement that when job J_j is processed on machine M_i, it requires $p_{ij} \ge 0$ time units and incurs a cost c_{ij}. The objective of the general bicriteria problem on unrelated parallel machines is to find a schedule obtaining a trade-off between the makespan and the total cost.

The general bicriteria problem on unrelated parallel machines as well as its many variants has received a great deal of attention from researchers over the last two decades. For instance, readers are referred to Lin and Vitter [14], Shmoys and Tardos [21], Jansen and Porkolab [12], Angel et al. [2], and so on. Given T, C and ϵ, assume that there exists a schedule with a makespan value of T and a cost of C. Lin and Vitter [14] presented a polynomial time algorithm, which finds a schedule with a makespan of at most $(2+\frac{1}{\epsilon})T$ and a cost of at most $(1+\epsilon)C$. This result was improved by Shmoys and Tardos [21], who proposed a polynomial time algorithm, which finds a solution with a makespan value of at most $2T$ and with a cost of at most C. It is worthwhile pointing out that the algorithm in Shmoys and Tardos [21] could imply a 2-approximation algorithm for the problem $P|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$. However, their algorithm may not be strongly polynomial, as it requires the solution of a linear programming formulation. One of objectives of this paper is devoted to providing a strongly polynomial 2-approximation algorithm for the problem $P|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$ in Section 2. Based on our 2-approximation algorithm, we also design a polynomial time approximation scheme (PTAS), which finds a solution of makespan at most $(1+\epsilon)OPT$ for $P|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$ in Section 3.

When the number of machines m is a constant, we present a fully polynomial time approximation scheme (FPTAS) with running time of $O(1/\epsilon^{2m+3}+mn^2)$ for $P_m|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$ in Section 3. As aforementioned, the problem $P_m|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$ has a strong connection with some bicriteria problems on unrelated parallel machines. Some results from these bicriteria problems may lead to FPTAS for the problem $P_m|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$ too. For instance, Jansen and Porkolab [12] designed a FPTAS for the scheduling on unrelated parallel machines. The FPTAS computes a schedule in time $O(n(m/\epsilon)^{O(m)})$ with makespan of at most $(1+\epsilon)T$ and cost of at most $(1+\epsilon)C$, provided there exists a schedule with makespan of T and cost of C. This approximation ratio was improved by Angel et al. [2], who proposed a FPTAS, that finds a schedule with makespan of at most $(1+\epsilon)T$ and cost of at most C, if there exists a schedule with makespan of T and cost of C, for the unrelated parallel machines scheduling problem with costs. However, the running time of the FPTAS in Angel et al. [2] is $O(n(n/\epsilon)^m)$, which is higher than that in Jansen and Porkolab [12]. As a consequence, the algorithm in Angel et al. [2] could imply a FPTAS for the problem $P_m|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$. Recently, Zhang et al. [24] has presented a FPTAS with running time of $O(n^{m+2}/\epsilon^m)$ for the problem $P_m|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$. In the case where the number of jobs is relatively large, say $n > \frac{1}{\epsilon}$, which is a reasonable size of jobs in most of the practical cases, our FPTAS has a better running time of $O(1/\epsilon^{2m+3}+mn^2)$.

Recently, some variants of the parallel machine scheduling problem with rejection have received considerable attention. Engels et al. [7] studied the objective of minimizing the sum of the weighted completion times of the accepted jobs plus the sum of the penalties of the rejected jobs. The preemptive cases are considered by Hoogeveen et al. [10] and Seiden [18]. The batch cases are studied by Cao and Yang [4] and Lu et al. [15–17]. Cao and Zhang [5] presented a PTAS for scheduling with rejection and job release times, which generalized the result in Bartal et al. [3]. For the single-machine scheduling with rejection, Cheng and Sun [6] designed some FPTAS for the case where the size of a job is a linear function of its starting time under different objectives. Zhang et al. [22] considered the single machine scheduling problem with release dates and rejection. The same authors [23] also considered the problems of minimizing some classic objectives under the job rejection constraint. Shabtay et al. [19] presented a bicriteria approach to scheduling on a single machine with job rejection. More related results can be found in the recent survey in Shabtay et al. [20].

We would like to make a remark that the problem $P|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$ we considered in this paper is closely related to the parallel machine scheduling problem with rejection introduced by Bartal et al. [3], in which the given are m identical parallel machines and a set of n jobs with each job J_j characterized by a size p_j and a penalty e_j, and the objective is to minimize the makespan of the schedule for accepted jobs plus the sum of the penalties of the rejected jobs. Bartal et al. [3] presented a PTAS for the general case and a FPTAS for the case where the number of machines is fixed. Unfortunately, their method can not be generalized to our problem directly, as it may violate the job rejection constraint.

The rest of the paper is organized as follows. In Section 2 we design a strongly polynomial 2-approximation algorithm for the problem $P|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$ and we show that the factor analysis of our algorithm is best possible by providing tight examples. Section 3 is devoted to presenting a polynomial time approximation scheme (PTAS) for $P|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$. As a consequence, Section 3 contains a fully polynomial time approximation scheme (FPTAS) for $P_m|\sum_{J_j\in\mathcal{R}} e_j \le B|C_{\max}$. Finally, Section 4 contains the concluding remarks.