Information Processing Letters 115 (2015) 969–974

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

A note on unbounded parallel-batch scheduling

Zhichao Geng, Jinjiang Yuan *

School of Mathematics and Statistics, Zhengzhou University, Zhengzhou, Henan 450001, People's Republic of China

ARTICLE INFO

Article history:
Received 14 November 2014
Received in revised form 2 July 2015
Accepted 2 July 2015
Available online 7 July 2015
Communicated by B. Doerr

Keywords:
Analysis of algorithm
Parallel-batch scheduling
Maximum cost
Strongly polynomial time
Pareto optimization

ABSTRACT

This paper revisits the scheduling problem on an unbounded parallel batch machine for minimizing a maximum cost $f_{\max}$. It was reported in the literature that the decision version of the problem is solvable in $O(n^2 + n \log P)$ time, where n is the number of jobs and P is the total processing time of jobs. This implies that the optimization version for minimizing $f_{\max}$ can be solved in weakly polynomial time. But a strongly polynomial-time algorithm has not been provided for this problem. In this paper, we present an $O(n^4)$-time algorithm for the Pareto optimization problem for minimizing $C_{\max}$ and $f_{\max}$, where $C_{\max}$ is the maximum completion time of jobs. Consequently, the problem for minimizing $f_{\max}$ can also be solved in $O(n^4)$ time.

1. Introduction

Suppose that we are given a set of n jobs $J_1, J_2, \cdots, J_n$ to be processed on a parallel-batch (shortly, p-batch) machine. All jobs and the machine are available from time zero onwards. Each job J_j has an integer processing time $p_j > 0$. A p-batch machine can process jobs in batches, where a batch B is a subset of jobs, the processing time of batch B is the largest processing time of the jobs in B, i.e., $p(B) = \max\{p_j : J_j \in B\}$, and the completion times of all jobs in a batch are defined to be the completion time of the batch. When the objective function to be minimized is regular (nondecreasing in the completion times of the jobs), we only need to consider the feasible schedules in which the batches are scheduled consecutively with no idle times. Then a schedule is given by a batch sequence $\sigma = (B_1, B_2, \cdots, B_k)$ in which the completion time of a job $J_j \in B_i$ in σ is given by $C_j(\sigma) = p(B_1) + p(B_2) + \cdots + p(B_i)$. As to the batch capacity, denoted by b, there are two versions in the literature: bounded version ($b < n$) and unbounded version ($b \geq n$). In this paper, we only consider the unbounded version. Moreover, we assume in this paper that all objective values are integral.

For a given schedule σ, let $f_j(\cdot)$ be the cost function for job J_j, $1 \leq j \leq n$. In this paper we assume that each $f_j(\cdot)$ is a regular (i.e., $f_j(t)$ is nondecreasing in t) function and for each $t \geq 0$, $f_j(t)$ can be calculated in constant time. Let $f_{\max}(\sigma) = \max_{1 \leq j \leq n} f_j(C_j(\sigma))$ be the maximum cost of σ. Using the standard three-field $\alpha|\beta|\gamma$ notation introduced by Graham et al. [2], we use $1|\text{p-batch}, b \geq n|f_{\max}$ to denote the unbounded p-batch scheduling problem to minimize the maximum cost $f_{\max}$.

P-batch scheduling was first introduced in Lee et al. [7] with bounded capacity. Later, Brucker et al. [1] extended the research of p-batch scheduling to the unbounded version. Up to now, p-batch scheduling has been extensively studied. For our purpose, we only review some related work.

Brucker et al. [1] presented an $O(n^2)$-time algorithm for problem $1|\text{p-batch}, b \geq n|L_{\max}$, where $L_{\max}$ is the maximum lateness of jobs. As a byproduct, Brucker et al. [1] also pointed that the feasibility (decision) problem $1|\text{p-batch}, b \geq n, f_{\max} \leq U|$ is solvable in $O(n^2 + n \log P)$ time. Then problem $1|\text{p-batch}, b \geq n|f_{\max}$ can be solved in polynomial time by binary search for the feasibility problem. However, this is not a strongly polynomial-time

* Corresponding author.
E-mail address: yuanjj@zzu.edu.cn (J. Yuan).

http://dx.doi.org/10.1016/j.ipl.2015.07.002
0020-0190/

algorithm. To the best of our knowledge, a strongly polynomial-time algorithm for the problem has not been provided in the literature. In this paper, we present an $O(n^4)$-time algorithm for problem $1|\text{p-batch}, b \geq n|f_{\max}$. Our research is closely related to the Pareto optimization scheduling for minimizing $C_{\max}$ and $f_{\max}$, where $C_{\max}$ is the maximum completion time of jobs.

Following Hoogeveen [5], the Pareto optimization scheduling problem on a single machine to minimize two regular objective functions f and g can be formulated as follows.

Pareto optimization: For a given schedule π, we denote by $(f(\pi), g(\pi))$ the objective vector of π. If there exists no schedule σ such that $(f(\sigma), g(\sigma)) \leq (f(\pi), g(\pi))$ and at least one of the two strict inequalities $f(\sigma) < f(\pi)$ and $g(\sigma) < g(\pi)$ holds, we call π a *Pareto optimal schedule* and $(f(\pi), g(\pi))$ a *Pareto optimal point* corresponding to π. The goal of the Pareto optimization scheduling is to find all Pareto optimal points and, for each Pareto optimal point, provide a corresponding Pareto optimal schedule. Following the notation of T'kindt and Billaut [8], the Pareto optimization scheduling problem on a single machine to minimize two objective functions f and g can be denoted by $1|\beta|\#(f, g)$, where β denotes the restricted constraints of the feasible schedules. Related to problem $1|\beta|\#(f, g)$, there are two constrained optimization scheduling problems $1|\beta|f : g \leq V$ and $1|\beta|g : f \leq U$, where $g \leq V$ ($f \leq U$) means the restriction that, for each feasible schedule, the value of objective function g (f) is no more than the given upper bound V (U).

If π is an optimal schedule of problem $1|\beta|f : g \leq V$ and also a Pareto optimal schedule of problem $1|\beta|\#(f, g)$, we say that π is *optimal and Pareto optimal* for problem $1|\beta|f : g \leq V$. If an algorithm returns an optimal and Pareto optimal schedule for each feasible instance of problem $1|\beta|f : g \leq V$, we also say that the algorithm is an *optimal and Pareto optimal algorithm* for problem $1|\beta|f : g \leq V$.

Suppose that we have an optimal and Pareto optimal algorithm $\mathcal{A}(V)$ for problem $1|\beta|f : g \leq V$ in hand. Then all Pareto optimal points for problem $1|\beta|\#(f, g)$ can be generated iteratively by the following algorithm Pareto-Generating($\mathcal{A}(V)$).

Pareto-Generating($\mathcal{A}(V)$): Set $V := +\infty$ and do $\mathcal{A}(V)$ on problem $1|\beta|f : g \leq V$ to obtain the first Pareto optimal point $(U^{(1)}, V^{(1)})$ and the corresponding Pareto optimal schedule π_1. Generally, if $(U^{(i)}, V^{(i)})$ and π_i have been generated, we set $V := V^{(i)} - 1$ and do $\mathcal{A}(V)$ on problem $1|\beta|f : g \leq V$ to obtain the next Pareto optimal point $(U^{(i+1)}, V^{(i+1)})$ and the corresponding Pareto optimal schedule π_{i+1}. This procedure terminates when we meet a number N so that $(U^{(N)}, V^{(N)})$ and π_N have been generated but the problem $1|\beta|f : g \leq V$ is infeasible for $V = V^{(N)} - 1$. Then $(U^{(i)}, V^{(i)})$, $1 \leq i \leq N$, are all Pareto optimal points of problem $1|\beta|\#(f, g)$, and their corresponding Pareto optimal schedules are given by π_i, $1 \leq i \leq N$.

The following lemma, which can be easily observed, was widely used in the literature, for example, in Hoogeveen and Van de Velde [6], He et al. [3], and He et al. [4].

Lemma 1.1. *Suppose that problem $1|\beta|\#(f, g)$ has at most $v(n)$ Pareto optimal points and the optimal and Pareto optimal algorithm $\mathcal{A}(V)$ used in Pareto-Generating($\mathcal{A}(V)$) has a time complexity $O(q(n))$ which is independent of the choice of V. Then algorithm Pareto-Generating($\mathcal{A}(V)$) solves problem $1|\beta|\#(f, g)$ in $O(q(n)v(n))$ time.*

Now the Pareto optimization scheduling problem on an unbounded parallel batch machine for minimizing $C_{\max}$ and $f_{\max}$ can be denoted by $1|\text{p-batch}, b \geq n|\#(C_{\max}, f_{\max})$, and the two related constrained optimization scheduling problems are denoted by $1|\text{p-batch}, b \geq n|C_{\max} : f_{\max} \leq F$ and $1|\text{p-batch}, b \geq n|f_{\max} : C_{\max} \leq C$, respectively.

Related work: He et al. [3] first studied the Pareto optimization problem $1|\text{p-batch}, b \geq n|\#(C_{\max}, L_{\max})$. Suppose that the jobs are sorted in SPT (shortest processing time first) order in the preprocessing procedure. They showed that an optimal and Pareto optimal schedule of problem $1|\text{p-batch}, b \geq n|C_{\max} : L_{\max} \leq L$ can be found in linear time, provided that L is a given upper bound. They also showed that problem $1|\text{p-batch}, b \geq n|\#(C_{\max}, L_{\max})$ has at most $\frac{n(n-1)}{2} + 1$ Pareto optimal points. As a consequence, problem $1|\text{p-batch}, b \geq n|\#(C_{\max}, L_{\max})$ can be solved in $O(n^3)$ time.

Recently, He et al. [4] studied problem $1|\text{p-batch}, b \geq n|\#(C_{\max}, f_{\max})$. In their research, in $O(n \log P)$ time, the constrained condition $f_{\max} \leq F$ is reduced to deadlines d_j for all jobs J_j, $1 \leq j \leq n$, given by the form $d_j = \max\{t : f_j(t) \leq F\}$. Then problem $1|\text{p-batch}, b \geq n|C_{\max} : f_{\max} \leq F$ is reduced to problem $1|\text{p-batch}, b \geq n|C_{\max} : L_{\max} \leq 0$ under the deadlines. By similar technique for solving problem $1|\text{p-batch}, b \geq n|C_{\max} : L_{\max} \leq L$ in [3], they showed that an optimal and Pareto optimal schedule of problem $1|\text{p-batch}, b \geq n|C_{\max} : f_{\max} \leq F$ can be obtained in $O(n \log P)$ time. They further showed that problem $1|\text{p-batch}, b \geq n|\#(C_{\max}, f_{\max})$ has at most $\frac{n(n-1)}{2} + 1$ Pareto optimal points. Consequently, problem $1|\text{p-batch}, b \geq n|\#(C_{\max}, f_{\max})$ can be solved in weakly polynomial $O(n^3 \log P)$ time.

Our contribution: In this paper, we directly devise an optimal and Pareto optimal algorithm for problem $1|\text{p-batch}, b \geq n|C_{\max} : f_{\max} \leq F$ with running time $O(n^2)$. Since there are at most $\frac{n(n-1)}{2} + 1$ Pareto optimal points from [4], by Lemma 1.1, problem $1|\text{p-batch}, b \geq n|\#(C_{\max}, f_{\max})$ can be solved in $O(n^4)$ time. We also construct an instance of problem $1|\text{p-batch}, b \geq n|\#(C_{\max}, f_{\max})$ which has exactly $\frac{n(n-1)}{2} + 1$ Pareto optimal points. This shows that the upper bound $\frac{n(n-1)}{2} + 1$ of Pareto optimal points established in He et al. [4] is tight. This leaves the following unaddressed issue.

Open problem: Is the upper bound $\frac{n(n-1)}{2} + 1$ of Pareto optimal points established in [3] tight for problem $1|\text{p-batch}, b \geq n|\#(C_{\max}, L_{\max})$?

2. Algorithm and analysis

2.1. A strongly polynomial-time algorithm

Following Brucker et al. [1], a schedule $\sigma = (B_1, B_2, \cdots, B_k)$ is called an *SPT-batch schedule*, if for every two