Information Processing Letters 112 (2012) 738–742

Contents lists available at SciVerse ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

A note on minimizing the sum of quadratic completion times on two identical parallel machines

F. Della Croce [a,*], C. Koulamas [b]

[a] DAI – Politecnico di Torino, Italy
[b] Florida International University, United States

ARTICLE INFO

Article history:
Received 8 November 2011
Accepted 19 June 2012
Available online 26 June 2012
Communicated by Jinhui Xu

Keywords:
Parallel machines
Scheduling
Quadratic completion times

ABSTRACT

We consider the problem of minimizing the sum of quadratic completion times on two parallel machines and we discuss the approximation ratio of the generalized shortest processing time (*GSPT*) priority rule according to which the jobs are sorted in non-decreasing processing time order and the next job on the list is assigned to the earliest available machine. We show that the approximation ratio of the *GSPT* rule is bounded above by $\frac{\sqrt{5}+2}{\sqrt{5}+1} \approx 1.309$ and below by $\frac{\sqrt{6}+2}{\sqrt{6}+1} \approx 1.290$. Extensions to the parallel m-machine problem are also discussed.

1. Introduction

One of the earliest observations in single-machine scheduling is that the scheduling problem is easy when the optimal solution can be obtained by implementing a simple index priority rule. In that case, a single priority index is computed for each job using its own characteristics and an optimal sequence is obtained by ranking the jobs according to the values of their indices.

In the case of non-preemptive deterministic single-machine scheduling problems with simultaneous job arrivals, the earliest application of an index priority rule dates back to Smith [8] who showed that the shortest processing time (*SPT*) priority rule can be used to minimize the summation of the job completion times $TC = \sum_{j=1}^{n} C_j$ where C_j denotes the completion time of job j, $j = 1, \dots, n$. It should be pointed out that Smith's [8] result was actually derived for the more general total weighted completion time objective function $\sum_{j=1}^{n} w_j C_j$ where w_j denotes the weight of job j. Smith's [8] results motivated the consideration of other scheduling functions that can be optimized by the use of a simple index priority rule.

The issue of investigating the existence of additional scheduling objective functions that can be optimized by an index priority rule was settled by Rothkopf and Smith [7] who showed that the $\sum_{j=1}^{n} w_j C_j$ and the $\sum_{j=1}^{n} w_j(1 - e^{-rC_j})$ can be optimized by the *SPT* index priority rule. Similarly, Townsend [9] showed that the quadratic function $\sum_{j=1}^{n} C_j^2$ is also minimized by the *SPT* index priority rule but this result does not extend to the weighted case.

Additional research focused on extending the above results to an identical parallel machine setting. It was observed that the problem of minimizing the summation of the job completion times on m identical parallel machines (the $Pm \parallel \sum_{j=1}^{n} C_j$ problem) can be solved by the generalized *SPT* rule (*GSPT*) of Conway et al. [3] according to which the jobs are listed in the *SPT* order and the next job on the list is assigned to the next available machine (with ties broken in favor of the lowest numbered machine). The proof of this result was based on the application of the weight matching approach of Hardy et al. [5] to a parallel machine environment. It was also observed that this approach does not extend to the weighted case since the $P2 \parallel \sum_{j=1}^{n} w_j C_j$ was shown to be NP-hard by Bruno et al. [1].

* Corresponding author.
E-mail addresses: federico.dellacroce@polito.it (F. Della Croce), koulamas@fiu.edu (C. Koulamas).

0020-0190/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.ipl.2012.06.011

Kawaguchi and Kyan [6] implemented the *WSPT* heuristic for the $Pm \parallel \sum_{j=1}^{n} w_j C_j$ problem (in which the jobs are sequenced in the non-decreasing order of their ratios with the next job on the list being assigned to the earliest available machine) and obtained the $\frac{\sqrt{2}+1}{2}$ ratio bound for it. With respect to the other two objective functions mentioned above, it can be shown that the *GSPT* rule is optimal for the unweighted $Pm \parallel \sum_{j=1}^{n}(1-e^{-rC_j})$ problem. In fact a stronger result holds: *GSPT* minimizes $\sum j = 1^n f(C_j)$ for any concave increasing function f. Furthermore, a similar result is true also for stochastic processing times: the shortest expected processing time first policy is optimal if the processing times are stochastically comparable. This has been shown in [10].

On the other hand, the complexity status of the $Pm \parallel \sum_{j=1}^{n} C_j^2$ problem remained open until recently when the problem was proved to be strongly NP-hard by Cheng and Liu [2]. Here, we consider the special case problem of minimizing the sum of quadratic completion times on two parallel machines. In [2], it is shown by probabilistic analysis that the approximation ratio of the *GSPT* rule asymptotically solves the problem. Here, we show that, on 2 machines, the approximation ratio of the *GSPT* rule is bounded above by $\frac{\sqrt{5}+2}{\sqrt{5}+1} \approx 1.309$ and below by $\frac{\sqrt{6}+2}{\sqrt{6}+1} \approx 1.290$. Also, we show that for the more general m-machine case such ratio is not less than $\frac{\sqrt{m+4}+2}{\sqrt{m+4}+1}$. A preliminary version of this work was presented in [4].

2. Main result

The $P2 \parallel \sum_{j=1}^{n} C_j^2$ problem can be stated as follows. A set of n jobs must be processed on a set of two identical parallel machines. Each job i has a processing time p_i which is identical on all machines. Each machine can process at most one job at a time. Preemption is not allowed. The objective is to minimize the sum of quadratic completion times. The problem is usually referred to as $P2 \parallel \sum C_j^2$. W.l.o.g. we can assume that the jobs are indexed in the shortest processing time order, that is $p_1 \leqslant p_2 \leqslant \cdots \leqslant p_n$ with ties broken arbitrarily. Given any schedule S, we denote by $C_j(S)$ the completion time of job j and by $C_{[j]}(S)$ the j-th completion time in the schedule S. Let S_{OPT} and S_{GSPT} be the optimal schedule and the schedule obtained by applying the *GSPT* rule respectively. The following lemma holds.

Lemma 1. *For any sequence S, we have*

$$C_{[j]}(S) \geqslant \max\left\{p_j, \frac{\sum_{i=1}^{j} p_i}{2}\right\}.$$

Proof. $C_{[j]}(S) \geqslant p_j$ trivially holds as j is the j-th jobs in the *GSPT* order. Besides, $C_{[j]}(S) \geqslant \frac{\sum_{i=1}^{j} p_i}{2}$ holds as $\sum_{i=1}^{j} p_i$ is a lower bound on the sum of the processing times of the first j jobs in S and the best we can do is to equally partition these processing times on the two machines. □

The following lemma also holds.

Lemma 2. *For any sequence S, we have $C_{[j]}(S) + C_{[j-1]}(S) \geqslant \sum_{i=1}^{j} p_i$.*

Proof. If $[j]$ and $[j-1]$ are sequenced on different machines,

$$C_{[j]}(S) + C_{[j-1]}(S) = \sum_{i=1}^{j} p_{[i]}(S) \geqslant \sum_{i=1}^{j} p_i.$$

Alternatively, $[j]$ and $[j-1]$ are sequenced on the same machine. Let $[h]$ be the largest completion time job on the other machine such that $C_{[h]}(S) \leqslant C_{[j-1]}(S)$. Then,

$$C_{[j]}(S) + C_{[h]}(S) = \sum_{i=1}^{j} p_{[i]}(S) \geqslant \sum_{i=1}^{j} p_i$$

and, correspondingly,

$$C_{[j]}(S) + C_{[j-1]}(S) \geqslant C_{[j]}(S) + C_{[h]}(S) \geqslant \sum_{i=1}^{j} p_i. \qquad \square$$

We focus now on the completion times of the $(j-1)$-th job and the j-th job in the *GSPT* schedule. The following lemma holds.

Lemma 3. *In the GSPT schedule, the following conditions hold*:

(a) $C_{[j]}(S_{GSPT}) + C_{[j-1]}(S_{GSPT}) = \sum_{i=1}^{j} p_i$,
(b) $C_{[j]}(S_{GSPT}) \leqslant p_j + C_{[j-1]}(S_{GSPT})$,
(c) $C_{[j]}(S_{GSPT}) \leqslant p_j + \frac{\sum_{i=1}^{j-1} p_i}{2}$.

Proof. Given the *GSPT* rule, the first condition has been shown in the proof of Lemma 2. For the second condition, given that $C_{[j]}(S_{GSPT}) + C_{[j-1]}(S_{GSPT}) = \sum_{i=1}^{j} p_i$ and therefore $C_{[j-1]}(S_{GSPT}) + C_{[j-2]}(S_{GSPT}) = \sum_{i=1}^{j-1} p_i$, we derive $C_{[j]}(S_{GSPT}) = p_j + C_{[j-2]}(S_{GSPT})$ with $C_{[j-2]}(S_{GSPT}) \leqslant C_{[j-1]}(S_{GSPT})$. Hence, $C_{[j]}(S_{GSPT}) \leqslant p_j + C_{[j-1]}(S_{GSPT})$. For the third condition, as

$$C_{[j-1]}(S_{GSPT}) + C_{[j-2]}(S_{GSPT}) = \sum_{i=1}^{j-1} p_i$$

and

$$C_{[j-2]}(S_{GSPT}) \leqslant C_{[j-1]}(S_{GSPT}),$$

we have $C_{[j-2]}(S_{GSPT}) \leqslant \frac{\sum_{i=1}^{j-1} p_i}{2}$. Hence,

$$C_{[j]}(S_{GSPT}) = p_j + C_{[j-2]}(S_{GSPT}) \leqslant p_j + \frac{\sum_{i=1}^{j-1} p_i}{2}. \qquad \square$$

The following properties hold.

Property 1. *If $\sum_{i=1}^{j-1} p_i \geqslant p_j$, then $C_{[j]}^2(S_{OPT}) + C_{[j-1]}^2(S_{OPT}) \geqslant 2(\frac{\sum_{i=1}^{j} p_i}{2})^2$ else, $\sum_{i=1}^{j-1} p_i < p_j$ and $C_{[j]}^2(S_{OPT}) + C_{[j-1]}^2(S_{OPT}) \geqslant p_j^2 + (\sum_{i=1}^{j-1} p_i)^2$.*