Operations Research Letters 41 (2013) 570–575

Contents lists available at ScienceDirect

Operations Research Letters

journal homepage: www.elsevier.com/locate/orl

Analysis of Smith's rule in stochastic machine scheduling

Caroline Jagtenberg [a], Uwe Schwiegelshohn [b], Marc Uetz [c,*]

[a] CWI, Science Park 123, 1098 XG Amsterdam, The Netherlands
[b] TU Dortmund, Robotics Research Institute, 44227 Dortmund, Germany
[c] University of Twente, Applied Mathematics, P.O. Box 217, 7500 AE Enschede, The Netherlands

ARTICLE INFO

Article history:
Received 18 February 2013
Received in revised form
30 July 2013
Accepted 2 August 2013
Available online 9 August 2013

Keywords:
Stochastic scheduling
WSEPT
Exponential distribution

ABSTRACT

In a landmark paper from 1986, Kawaguchi and Kyan show that scheduling jobs according to ratios weight over processing time – also known as Smith's rule – has a tight performance guarantee of $(1+\sqrt{2})/2 \approx 1.207$ for minimizing the weighted sum of completion times in parallel machine scheduling. We prove the counterintuitive result that the performance guarantee of Smith's rule is not better than 1.243 when processing times are exponentially distributed.

1. Introduction

Minimizing the weighted sum of completion times on m parallel, identical machines is an archetypical problem in the theory of scheduling. In this problem, we are given n jobs which have to be processed non-preemptively on m machines. Each job j comes with a processing time p_j and a weight w_j, and when C_j denotes job j's completion time in a given schedule, the goal is to compute a schedule that minimizes the total weighted completion time $\sum_j w_j C_j$. In the classical 3-field notation for scheduling problems [5], the problem is denoted by $P||\sum w_j C_j$. For a single machine, a simple exchange argument shows that scheduling the jobs in order of non-increasing ratios w_j/p_j gives the optimal schedule [15]. Greedily scheduling the jobs in this order on parallel machines is known as WSPT rule, weighted shortest processing times first, or Smith's rule. On parallel identical machines, WSPT is known to be a $\frac{1}{2}(1+\sqrt{2})$–approximation, and this bound is tight [8]. The computational tractability of the problem was finally settled by showing the existence of a PTAS [14], given that the problem is strongly NP-complete if m is part of the input [3,4].

In this paper, we consider the stochastic variant of the problem. It is assumed that the processing time p_j of a job j is not known in advance. It becomes known upon completion of the job. Only the distribution of the corresponding random variable P_j, or at least its expectation $\mathbb{E}\left[P_j\right]$, is given beforehand. More specifically, we assume that the processing times of jobs are governed by independent, exponentially distributed random variables. That is to say, each job comes with a parameter $\lambda_j > 0$, and the probability that its processing time exceeds t equals

$$\mathbb{P}\left[P_j > t\right] = e^{-\lambda_j t}.$$

We denote this by writing $P_j \sim \exp(\lambda_j)$. Exponentially distributed processing times somehow represent the cream of stochastic scheduling, in particular when juxtaposing stochastic and deterministic scheduling: the exponential distribution is characterized by the memoryless property, that is,

$$\mathbb{P}\left[P_j > s+t | P_j > s\right] = \mathbb{P}\left[P_j > t\right].$$

So for any non-finished job it is irrelevant how much processing it has already received. This is obviously a decisive difference to deterministic scheduling models, and puts stochastic scheduling apart. Next to that, the model with exponentially distributed processing times is attractive because it makes the stochastic model analytically tractable.

In the stochastic setting with the objective to minimize $\mathbb{E}[\sum w_j C_j]$, the analogue of Smith's rule is greedily scheduling the jobs in order of non-increasing ratios $w_j/\mathbb{E}\left[P_j\right]$, also called WSEPT (weighted shortest expected processing time first) [12]. For a single machine, this is again optimal [13]. For parallel machines, it has been shown that the WSEPT rule achieves a performance bound of $(2 - 1/m)$ within the class of all non-anticipatory stochastic scheduling policies [11]. Here, the considered metric is the expected performance of WSEPT relative to that of an (unknown) optimal non-anticipatory scheduling policy. We refer to [10] for the

* Corresponding author.
E-mail addresses: c.j.jagtenberg@cwi.nl (C. Jagtenberg), uwe.schwiegelshohn@udo.edu (U. Schwiegelshohn), m.uetz@utwente.nl, marc.uetz@gmail.com (M. Uetz).

0167-6377/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.orl.2013.08.001

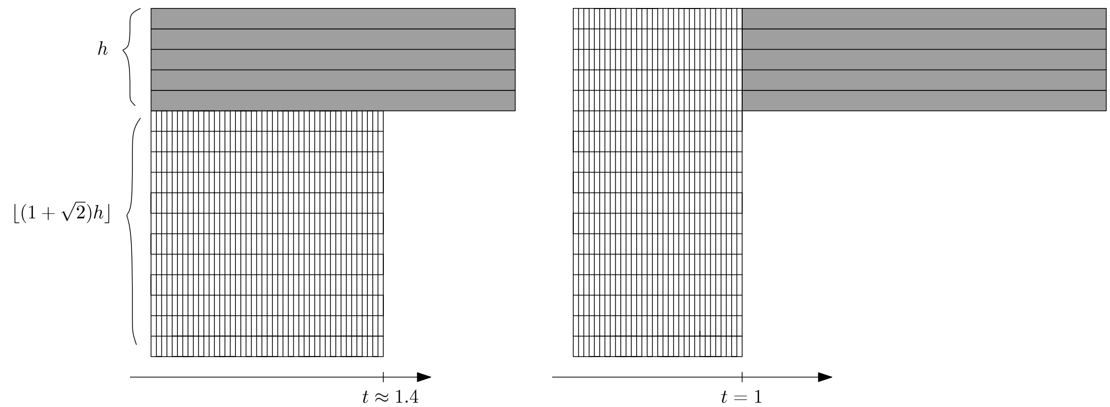

Fig. 1. Two different WSPT schedules, one with optimal objective value v^* on the left, and one with suboptimal value v on the right, respectively.

precise definition on non-anticipatory stochastic scheduling policies. For the purpose of this paper, it suffices to know that non-anticipatory stochastic scheduling policies are, at any given time t, only allowed to use information that is available at that time t. Obviously, this is also the case for WSEPT, as the distributions P_j, thus particularly expected processing times $\mathbb{E}\left[P_j\right]$ are even available beforehand.

The major purpose of this paper is to establish the first lower bound for the $(2-1/m)$ performance guarantee of [11] for exponentially distributed processing times. In fact, we are not aware of any result in this direction. The only result known to us is an instance showing that WSEPT can miss the optimum by a factor 3/2, but then for arbitrary processing time distributions [16, Ex. 3.5.12]. Our main result is the following.

Theorem 1. *When scheduling jobs with exponentially distributed processing times on parallel, identical machines in order to minimize $\mathbb{E}[\sum w_jC_j]$, the performance guarantee of Smith's rule is no better than α with $\alpha > 1.243$.*

To obtain our result, we carefully adapt and analyse the worst-case instance of [8]. Note that the originality of this result lies in the fact that $1.243 > \frac{1}{2}(1+\sqrt{2}) \approx 1.207$. Hence, stochastic scheduling with exponentially distributed processing times has worse worst-case instances than deterministic scheduling. This result may seem counterintuitive, as Pinedo correctly claims the following.

> "It is intuitively acceptable that a deterministic problem may be NP-hard while its counterpart with exponentially distributed processing times allows for a very simple policy to be optimal" [12].

An example for this intuition is given by the problem to minimize the makespan on parallel identical machines: while the problem is NP-hard in deterministic scheduling, the version with exponentially distributed processing times is solved optimally by the LEPT policy (longest expected processing times first) [17]. For the minsum objective considered in this paper, the picture is as follows. For unit weights where $w_j = 1$, the SPT rule is optimal for minimizing $\sum_j C_j$ in the deterministic setting [12], and also SEPT (shortest expected processing time first) is optimal for minimizing $\mathbb{E}[\sum_j C_j]$ when processing times are exponentially distributed [1]. For exponentially distributed processing times and weights that are agreeable in the sense that there exists an ordering such that $w_1 \geq \cdots \geq w_n$ and $w_1\lambda_1 \geq \cdots \geq w_n\lambda_n$, scheduling the jobs in order $1, 2, \ldots, n$ is optimal [7], while the corresponding deterministic problem is NP-hard, and in particular, WSPT is not optimal.

That is to say, there are examples where the stochastic version with exponentially distributed processing times is computationally easier than the deterministic version of the same problem, under the realm of minimizing expected performance. Our result shows that with arbitrary weights, the situation is different. Next to this qualitatively new insight, our analysis also sheds light on phenomena in stochastic scheduling which are interesting on their own.

The paper is organized as follows. In Section 2, we briefly review and visualize the worst-case instance presented in [8]. We explain the intuition behind the stochastified instance of [8] in Section 3. Then we derive four technical lemmas about scheduling jobs with exponentially distributed processing times, and finally prove the claimed lower bound for the performance of Smith's rule. Finally, Section 4 contains our conclusions.

2. Recap of the Kawaguchi and Kyan instance

We briefly summarize the instance from [8] that achieves the bound $(1+\sqrt{2})/2$ for deterministic scheduling, as the instance we propose is a stochastic variant thereof.

Let n be the number of jobs and m the number of machines. Denote the processing time of job j by p_j and its weight by w_j. The (deterministic) instance is then given by

$$m = h + \lfloor(1+\sqrt{2})h\rfloor$$
$$n = mk + h$$
$$p_j = w_j = 1/k \quad \text{for } 1 \leq j \leq mk$$
$$p_j = w_j = 1+\sqrt{2} \quad \text{for } mk+1 \leq j \leq mk+h.$$

Here, h denotes an integer, and k is an integer that can be divided by $\lfloor(1+\sqrt{2})h\rfloor$. Notice that $w_j/p_j = 1$ for all jobs j. This means that any list schedule is in fact a WSPT schedule. Let us refer to the first mk jobs as short jobs, and the remaining h jobs as long jobs.

Let v^* be the total weighted completion time of a schedule where the long jobs are processed first, and v be the total weighted completion time of a schedule in which all short jobs are processed first. Fig. 1 depicts these two schedules. The schedule on the left of Fig. 1 has objective value v^*. Here the last jobs of length $1/k$ finish at time $1 + h/\lfloor(1+\sqrt{2})h\rfloor \approx 1.4$ (for large values of h and k). The schedule on the right of Fig. 1 has value v, and it finishes the last jobs of length $1/k$ exactly at time 1. In Fig. 1 we used $h = 5$ and $k = 32$. It can be verified (see [8]) that $v = (1+\sqrt{2})(2+\sqrt{2})h + (m/2)(1+1/k)$ and $v^* = (1+\sqrt{2})^2h + (m/2)(m/\lfloor(1+\sqrt{2})h\rfloor + 1/k)$. The ratio v/v^* then tends to $(1+\sqrt{2})/2$ as $h \to \infty$ and $k \to \infty$.

3. The stochastic Kawaguchi and Kyan instance

We find it particularly instructive to consider the stochastic analogue of the instance presented by Kawaguchi and Kyan [8],