Available online at www.sciencedirect.com

SCIENCE DIRECT®

European Journal of Operational Research 161 (2005) 3–10

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

www.elsevier.com/locate/dsw

An improved approximation algorithm for the single machine total completion time scheduling problem with availability constraints

Cherif Sadfi [a,*], Bernard Penz [a], Christophe Rapine [a], Jacek Błażewicz [b], Piotr Formanowicz [b]

[a] *Laboratoire GILCO, 46 avenue Félix Viallet, 38031 Grenoble Cedex 1, France*

[b] *Institute of Computing Science, Poznań University of Technology, Piotrowo 3a, 60-965 Poznań, Poland*

Available online 13 November 2003

Abstract

In this paper, we study the single machine total completion scheduling problem subject to a period of maintenance. We propose an approximation algorithm to solve the problem with a worst case error bound of 3/17. Furthermore, an example is provided to show that the bound is tight. Computational experiments and an analysis are given afterwards.

Keywords: Single machine scheduling; Total completion time; Availability constraints; Approximation algorithm; Worst case analysis

1. Introduction

We consider the problem of scheduling jobs on a single machine in order to minimize the total completion time. The machine is subject to a period of maintenance during the processing of jobs (for a survey of scheduling problems with limited machine availability see [4,5]). The date of the maintenance and its duration are known. Preemptions are not allowed, i.e., a job that is preempted due to the maintenance must be restarted after the machine is repaired. The problem is denoted by $1, h_1 || \sum C_i$. Adiri et al. [1] and Lee and Liman [2] show that the problem is *NP*-hard. They also study the shortest processing time (SPT) algorithm as a heuristic solving this problem. Lee and Liman [2] show that the error bound of the SPT method is 2/7.

In this paper we propose an improved heuristic method for solving the problem with a performance guarantee of 20/17. The heuristic is based on a post-optimization of the SPT solution using a 2-OPT procedure. We call this heuristic MSPT in reference to Modified SPT.

2. Notations and description of the MSPT heuristic

Let $J = \{J_i / i = 1, \ldots, n\}$ be the set of n jobs to be scheduled. The following notations are used to formulate the problem.

* Corresponding author.

E-mail addresses: sadfi@gilco.inpg.fr (C. Sadfi), penz@gilco.inpg.fr (B. Penz), rapine@gilco.inpg.fr (C. Rapine), blazewic@cs.put.poznan.pl (J. Błażewicz), piotr@cs.put.poznan.pl (P. Formanowicz).

0377-2217/$ - see front matter
doi:10.1016/j.ejor.2003.08.026

Notations

$J_{[i]}$	job scheduled at position i
p_i	processing time of job J_i
$p_{[i]}$	processing time of the job scheduled at position i
C_i	completion time of job J_i
$C_{[i]}$	completion time of the job scheduled at position i
R	date of the maintenance
L	duration of the maintenance. We denote $R+L=D$
$B\max$	maximum number of jobs scheduled before the period of the maintenance in any feasible solution. This quantity is given by the SPT schedule.

The MSPT heuristic tries to improve the result given by the SPT algorithm by exchanging one job scheduled before the maintenance with another job scheduled after the period of maintenance in order to minimize the objective function. In each exchange, the jobs are reordered before and after the maintenance in non-decreasing order of their processing times.

The MSPT heuristic

1. Schedule the jobs according to the SPT rule.
2. Try all possible exchanges of one job scheduled before the period of maintenance in SPT, with one job scheduled after the period of maintenance. Choose the best exchange.

3. Analysis of the MSPT heuristic

In this section we first establish some of the properties of the solution obtained using SPT. These properties are inspired from Lee and Liman [2], and will be useful to analyse the MSPT heuristic.

We consider throughout this paper the schedule S generated by the SPT algorithm, together with an optimal schedule S^*. The schedule S' will denote the MSPT solution computed from S, see Figs. 1–3. We can look at any feasible solution as a partition of the jobs into two sets: the jobs scheduled before and the jobs scheduled after the period of maintenance. Clearly sequencing both sets by non-decreasing processing time of the jobs is dominant.

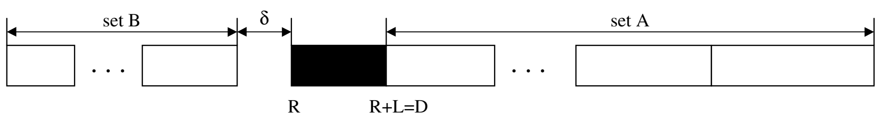

Fig. 1. Schedule S (SPT).

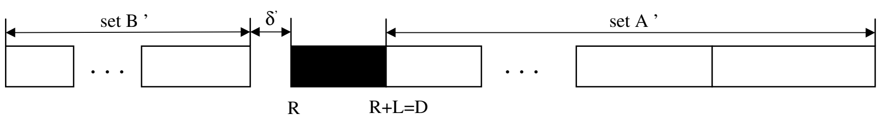

Fig. 2. Schedule S' (MSPT).

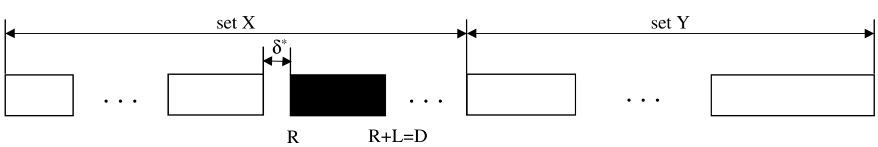

Fig. 3. Schedule S^* (OPT).