J. Opl Res. Soc. Vol. 45, No. 5, pp. 589–594
Printed in Great Britain. All rights reserved

0160-5682/94 $9.00+0.00

Single Machine Earliness-Tardiness Scheduling Problems Using the Equal–Slack Rule

CEYDA OĞUZ and CEMAL DINCER
Bilkent University, Turkey

The purpose of this paper is to analyse a special case of the non-pre-emptive single machine scheduling problem where the distinct due dates for each job are related to processing times according to the Equal–Slack rule. The scheduling objective is to minimize the sum of earliness and tardiness penalties. After determining some properties of the problem, the unrestricted case is shown to be equivalent to a polynomial time solvable problem, whereas the restricted case is shown to be NP-hard, and suggestions are made for further research.

Key words: machine scheduling

INTRODUCTION

Recently, the single-machine scheduling problem with earliness and tardiness penalties has attracted enormous attention from researchers. Surveys can be found in Baker and Scudder[1], Cheng and Gupta[2] and Sen and Gupta[3]. The objective function of a scheduling problem with earliness and tardiness penalties is consistent with the just-in-time concept in which both early and late completion of jobs from their due dates are prohibited. Very briefly, while early jobs result in inventory holding costs, late jobs result in penalties, such as loss of customer goodwill and loss of the orders. Therefore, minimizing the earliness and tardiness penalties has important practical applications. So far, most of the literature concentrates on problems with a common due date for all jobs. But, from a practical point of view, it is meaningful to have distinct due dates for every job.

Unfortunately, the general problem with distinct due dates is one of the NP-complete scheduling problems. This paper will present a special case of the general problem. The next section introduces the notation used in the paper and some properties of the special case are determined in the section after, where the problem is split into two cases—unrestricted and restricted. After showing the equivalence of the unrestricted case to a polynomially solvable problem, the NP-hardness for the restricted case is presented.

NOTATION

The machine scheduling problem studied in this paper requires n independent jobs J_j ($j = 1, \ldots, n$) to be processed on a single machine with the following assumptions: (1) all jobs are available for processing at time zero; (2) the single machine can process at most one job J_j at a time; and (3) no pre-emption is allowed.

Throughout the paper, it is assumed that a processing time p_j for each J_j ($\mathcal{P} = \{p_1, p_2, \ldots, p_n\}$), a target starting time a_j by which J_j should ideally be started, a due date d_j by which J_j should ideally be completed ($d_j = a_j + p_j$), and a cost function $f_j : \mathcal{N} \to \mathcal{R}$, indicating the costs incurred as a function of the completion time of J_j, can be specified for each J_j. It is assumed that all data, except f_j, are non-negative integers. Given a processing order, the starting time S_j ($\mathcal{S} = \{S_1, S_2, \ldots, S_n\}$), the completion time $C_j = S_j + p_j$, the tardiness $T_j = \max\{0, C_j - d_j\}$, and the earliness $E_j = \max\{0, d_j - C_j\}$ can be computed for each J_j.

Correspondence: C. Oğuz, Department of Management, Hong Kong Polytechnic, Hung Hom, Kowloon, Hong Kong

In the analysis of the given scheduling problem, the following additional notation is used:
π and σ denote sequences of jobs;
$\pi(i)$ and $\sigma(i)$ denote the ith job in the sequence;
$z(\sigma)$ denotes the value of optimality criterion for the schedule of σ and $z(\sigma) = \sum_{j=1}^{n}(E_j + T_j)$;
$\mathcal{E}$ represents the set of jobs that complete before the target starting time;
$\mathcal{T}$ represents the set of jobs that start exactly on or after the target starting time.

In this paper, we follow the terminology used in Lawler *et al.*[4] in order to identify the scheduling problems defined by the above formulation.

ANALYSIS OF $1|d_j = p_j + q|\sum_{j=1}^{n}(E_j + T_j)$

Garey *et al.*[5] have shown that $1|d_j|\sum_{j=1}^{n}(E_j + T_j)$ is NP-hard by a reduction from the Even–Odd Partition problem. They presented an efficient algorithm to find the optimal schedule for fixed job sequencing. This algorithm inserts idle times between jobs in the given sequence. They also developed a polynomial algorithm for the problem when all jobs have equal processing times.

Since $1|d_j|\sum_{j=1}^{n}(E_j + T_j)$ is NP-hard in its general form, a special case, which is one of the models given by Baker and Scudder[1], is analysed. Namely, problems in which distinct due dates are related to processing times according to the Equal–Slack rule are considered. This means that distinct due dates are given as $d_j = p_j + q$, $q > 0\ \forall j$. Hence, the problem can be stated as $1|d_j = p_j + q|\sum_{j=1}^{n}(E_j + T_j)$. Since this problem permits distinct due dates that relate to processing times, it allows a nice structure for the optimal solutions in a special case.

For further analysis, the notation $\sum_{j=1}^{n}|C_j - d_j|$ will be used instead of $\sum_{j=1}^{n}(E_j + T_j)$ since minimizing the sum of unweighted earliness and tardiness penalties is equivalent to minimizing the sum of absolute deviation of completion times from respective distinct due dates. Furthermore, it is easy to observe the following.

Observation 1

Minimizing $|C_j - d_j|$ is equivalent to minimizing $|S_j - a_j|$.

Proof

By definition, $d_j = a_j + p_j$. Substituting this into $|C_j - d_j|$ yields the following:

$$|C_j - d_j| = |C_j - (a_j + p_j)| = |C_j - p_j - a_j|.$$

The definition $S_j = C_j - p_j$, gives $|C_j - d_j| = |S_j - a_j|$. Hence, minimizing $|C_j - d_j|$ is equivalent to minimizing $|S_j - a_j|$.

Observation 2

If $d_j = p_j + q$ then $a_j = q\ \forall j$.

Proof

This result is obtained easily after substituting $d_j = p_j + q$ into $a_j = d_j - p_j$ as follows:

$$a_j = d_j - p_j = p_j + q - p_j = q.$$

Observation 2 means that each job with $d_j = p_j + q$ has a common target starting time, namely q.

Observation 3

$1|d_j = p_j + q|\sum_{j=1}^{n}(E_j + T_j)$ is equivalent to $1|a_j = q|\sum_{j=1}^{n}|S_j - a_j|$.