Computers & Operations Research 51 (2014) 200–207

Contents lists available at ScienceDirect

Computers & Operations Research

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

Single machine scheduling problem with interval processing times to minimize mean weighted completion time

Ali Allahverdi [a,*], Harun Aydilek [b], Asiye Aydilek [c]

[a] Department of Industrial and Management Systems Engineering, Kuwait University, P.O. Box 5969, Safat, Kuwait
[b] Department of Mathematics and Natural Sciences, Gulf University for Science and Technology, P.O. Box 7207, Hawally 32093, Kuwait
[c] Department of Economics and Finance, Gulf University for Science and Technology, P.O. Box 7207, Hawally 32093, Kuwait

ARTICLE INFO

Available online 14 June 2014

Keywords:
Scheduling
Single machine
Mean completion time
Uncertainty
Heuristics

ABSTRACT

The single resource scheduling problem is commonly applicable in practice not only when there is a single resource but also in some multiple-resource production systems where only one of the resources is bottle neck. Thus, the single resource (machine) scheduling problem has been widely addressed in the scheduling literature. In this paper, the single machine scheduling problem with uncertain and interval processing times is addressed. The objective is to minimize mean weighted completion time. The problem has been addressed in the literature and efficient heuristics have been presented. In this paper, some new polynomial time heuristics, utilizing the bounds of processing times, are proposed. The proposed and existing heuristics are compared by extensive computational experiments. The conducted experiments include a generalized simulation environment and several additional representative distributions in addition to the restricted experiments used in the literature. The results indicate that the proposed heuristics perform significantly better than the existing heuristics. Specifically, the best performing proposed heuristic reduces the error of the best existing heuristic in the literature by more than 75% while the computational time of the best performing proposed heuristic is less than that of the best existing heuristic. Moreover, the absolute error of the best performing heuristic is only about 1% of the optimal solution. Having a very small absolute error along with a negligible computational time indicates the superiority of the proposed heuristics.

1. Introduction

Scheduling decisions directly affect production costs and customer satisfaction. This is because the right scheduling decisions help reduce production costs as a result of better resource utilization. This leads to shorter delivery time to customers, and hence, increased customer satisfaction.

Scheduling jobs (tasks) on a single machine (resource) is widely applicable in real life. Moreover, in many applications of multiple-machine production systems, one machine is bottle neck, and hence, the right scheduling decision on that particular machine greatly affects the performance of the production system. Therefore, the problem of scheduling on a single machine is important, and hence, numerous researchers addressed this problem.

There are many applications of the single machine scheduling problem where job processing times are known with certainty, e.g., Vilà and Pereira [14], Valente and Schaller [13], Kianfar and Moslehi [4]. Therefore, the vast majority of research on the single machine scheduling problem has been devoted to the case of deterministic problem where job processing times are treated as known and fixed values. Some researchers addressed the problem where job processing times are modeled as stochastic random variables with certain mean and variance, e.g., Iranpoor et al. [3].

For some scheduling environments, the exact probability distributions for processing times may not be known. A solution obtained by assuming a certain probability distribution may not be even close to the optimal solution for the realized processing times. It has been observed that although it is hard to obtain the exact probability distributions of processing times before scheduling, it is relatively easier to obtain the upper and lower bounds of processing times in many practical cases. Therefore, the bounds of processing times can be utilized in finding a solution for the scheduling problem. This problem is known as uncertain scheduling problem with bounded or interval processing times, Sotskov et al. [10].

Scheduling problems with uncertain and bounded processing times have also been addressed in the literature for other scheduling environments such as flowshops. For example, Allahverdi and Aydilek [1] addressed the two-machine flowshop scheduling problem with interval processing times with the objective of

* Corresponding author.
E-mail addresses: ali.allahverdi@ku.edu.kw (A. Allahverdi), aydilek.h@gust.edu.kw (H. Aydilek), aydilek.a@gust.edu.kw (A. Aydilek).

http://dx.doi.org/10.1016/j.cor.2014.06.003

minimizing makespan. They provided several polynomial time heuristic algorithms and showed that one of the algorithms yields close to the optimal solution. Some other researchers that addressed the scheduling problems with interval job processing times include Lai et al. [6], Lai & Sotskov [5], Sotskov et al. [9,11], Sotskov and Lai [12], and Aydilek et al. [2]. It should be noted that Sotskov et al. [11] and Sotskov and Lai [12] used the stability method, by introducing a stability box which is a subset of stability region, in identifying solutions. The objective of this paper, however, is to construct heuristic algorithms.

The performance measure considered in this paper is mean completion time which is related to the work in process (*WIP*) inventory. The *WIP* inventory includes the set of partially completed products either ongoing or awaiting processing. Holding *WIP* inventory is costly due to inventory storage and handling costs, taxes and insurance costs, damage, loss and spoilage. Therefore, holding *WIP* inventory can be considerably costly and this cost may even exceed the cost of holding finished goods inventory. Consequently, one of the main goals of the manager becomes sustaining the production with the minimum level of *WIP* inventory. Therefore, many researchers worked on developing new methods for handling *WIP*, e.g., Yang [15] and Massim et al. [7]. The objective function of minimizing the mean completion time minimizes the average *WIP* inventory during the entire production process of jobs. Since the *WIP* inventory cost is an important component of production cost, the considered objective helps reduce production cost, and hence, it increases the profit.

We consider the single machine scheduling problem with mean weighted completion time performance measure where job processing times are random and bounded. This problem was recently addressed by Sotskov et al. [10] where they presented some dominance relations, and developed two efficient heuristics. By computational experiments, they indicated that the errors of both heuristics were close to the optimal solution. In this paper, we address the same problem and propose several new heuristics. We show that our newly proposed heuristics perform considerably better than those of Sotskov et al. [10] while computational times of our proposed heuristics are less than those of their heuristics.

The rest of this paper is organized as follows. The next section briefly describes the problem. The proposed heuristics are presented in Section 3, and an illustrative example is provided in the subsequent section. Computational experiments are explained in Section 5, comparison of the heuristics is performed in Section 6, and finally Section 7 concludes the paper.

2. Problem definition

It should be noted that minimizing total weighed completion time and minimizing mean weighed completion time are equivalent performance measures. We use the second measure in this paper. Let *MWCT* denote the mean weighted completion time. Besides, let *MWCT* (π) represent the mean weighted completion time of a given sequence π.

The problem is to minimize the mean weighted completion time in a single-machine scheduling environment. There is a set of n available jobs waiting for processing and setup times are included in the processing times. There are no precedence relationships between the jobs.

We assume that the job processing times are uncertain variables with unknown probability distributions where only a lower bound $t_j^L \geq 0$ and an upper bound $t_j^U \geq t_j^L$ of the processing time t_j of job j ($j \in J = \{1, 2, \ldots, n\}$) are known before scheduling. Let C_j stand for the completion time of job j. Let the bracket $[j]$ denote the job in position j in a given sequence. Then, the completion time of the job in position j can be computed as

$$C_{[j]} = \sum_{i=1}^{j} t_{[i]}$$

The mean weighted completion time of a given sequence π can be computed by taking the average of the weighted job completion times in each position of the sequence π and given as follows:

$$MWCT(\pi) = \left(\sum_{j=1}^{n} w_{[j]} C_{[j]} \right) / n$$

where $w_{[j]}$ denotes the weight of the job in position j.

Such a single machine problem can be denoted as $1|t_j^L \leq t_j \leq t_j^U|\sum w_i C_i$ where the first term denotes that the problem involves a single machine. The second term indicates that processing times are uncertain variables with a value between some lower and upper bounds. The last term specifies that the performance measure is to minimize weighted completion time which is equivalent to minimizing mean weighted completion time. Notice that the problem $1|t_j^L \leq t_j \leq t_j^U|\sum w_i C_i$ can be considered as an uncertain single machine problem without any prior information about the probability distribution of the processing times. In this case, it is only known that the processing times of each job falls between some given lower and upper bounds with probability one.

3. Heuristics

When $t_j^L = t_j^U$ for all $j = 1, 2, \ldots, n$, the problem reduces to the deterministic single machine scheduling problem for which an optimal solution can be obtained by the weighted shortest processing time (*WSPT*) rule, Pinedo [8]. However, for at least some jobs, the lower bound is different from the upper bound. Moreover, it is not possible to know the exact value of the processing time t_j before the processing of job j has been completed. Yet, a decision on when to process job j has to be made before the observation of t_j. Hence, a decision on the timing of the jobs can be made only using the lower and upper bounds, t_j^L and t_j^U, which are the only available data on the processing time of job j. Therefore, several heuristics are generated using the lower and upper bounds, t_j^L and t_j^U, and these heuristics are described below.

For heuristics *AA1*–*AA5*, *WSPT* rule is applied to the problem by using at_j in place of job processing times where $at_j = [t_j^L + \beta(t_j^U - t_j^L)]/w_j$ for each job $j \in \{1, \ldots, n\}$ and for a given value of β which indicates the weight assigned to the lower and upper bounds. The sequence obtained becomes one of our proposed heuristics. The heuristic sequence *AA1* is obtained when $\beta = 0$. Similarly, the heuristic sequences *AA2*–*AA5* are obtained when $\beta = 0.25$, $\beta = 0.50$, $\beta = 0.75$ and $\beta = 1$. It should be noted that while *AA1* uses only the information of lower bounds, *AA5* uses only the information about upper bounds. *AA3* gives equal weights to the lower and upper bounds. On the other hand, *AA2* gives higher weight to lower bounds and *AA4* gives higher weight to upper bounds.

Additional heuristics are obtained by using the bounds. Heuristic *GA* is obtained by assigning the geometric average of t_j^U/w_j and t_j^L/w_j to at_j. More specifically, the heuristic sequence of *GA* is obtained when $at_j = (t_j^L t_j^U)^{1/2}/w_j$. Another heuristic, *HA*, is obtained by using the harmonic average of the bounds such that the $at_j = [2(t_j^L t_j^U)/(t_j^L + t_j^U)]/w_j$. In addition to *GA* and *HA*, *GC* and *HC* are obtained by using the complement such that the at_j for *GC* is $at_j = [+ t_j^L + t_j^U - (t_j^L t_j^U)^{1/2}]/w_j$ and the at_j for *HC* is $at_j = [t_j^L + t_j^U - 2(t_j^L t_j^U)/(t_j^L + t_j^U)]/w_j$.