Research on Single-Machine Scheduling With Past-Sequence-Dependent Setup Times and Effects of Deterioration and Learning

Gan-hua Yu (1325715538@qq.com)
Qufu Normal University School of Management

Research Article

Keywords: Scheduling, Single machine, Setup times, Deteriorating jobs, Learning effect

Posted Date: March 23rd, 2021

DOI: https://doi.org/10.21203/rs.3.rs-329280/v1

Research on single-machine scheduling with past-sequence-dependent setup times and effects of deterioration and learning

Gan-hua Yu

Abstract:Some result in a recent paper(Wang et al.,Int J Adv Manuf Technol,41:1221–1226,2009) are incorrect.In this note, we show by a counterexample that the published results are incorrect.
Keywords:Scheduling; Single machine;Setup times;Deteriorating jobs;Learning effect

1 Introduction

The recent paper"single-machine scheduling with past-sequence-dependent setup times and effects of deterioration and learning" [1] addresses the single machine scheduling problems with past-sequence-dependent setup times and effects of deterioration and learning.They showed that the makespan minimization problem, the total completion time minimization problem,the number of tardy jobs minimization problem and the total weighted completion time minimization problem remain polynomially solvable, respectively.As we observe, some results for the method in Wang et al.[1] are incorrect.In this note we will give a counterexamples to show the incorrectness of some results in Wang et al.[1].

We shall follow the notations and terminologies given in Wang et al.[1]. There are given a single machine and n independent and non-preemptive jobs that are available for processing at some time $t_0 \geq 0$. The machine can handle one job at a time and preemption is not allowed. Let α_j be the deterioration rate of job J_j in a sequence. In addition, let $p_{[k]}^A$ be the actual processing time of a job if it is scheduled in the kth position in a sequence. As in Wang and Cheng [2], we assume that the actual processing time of job J_j if it is started at time t and scheduled in position r is given by:

$$p_{jr}^A(t) = \alpha_j (b + ct) r^a, r, j = 1,2,..., n, \quad (1)$$

where $a \leq 0$ is a constant learning effect. Also, as in Koulamas and Kyparisis [3] and Kuo and Yang [4], we assume that the p-s-d setup time of job $J_{[r]}$ if it is scheduled in position r is given by:

Gan-hua Yu (*Corresponding author*)
School of Management,
Qufu Normal University,
Rizhao 276826, Shandong, China
e-mail address:yuganhua@qq.com
Gan-hua Yu
School of Engineering and Management,
Pingxiang University,
Pingxiang 337055, Jiangxi,China

$$s_{[1]} = 0 \text{ and } s_{[r]} = d\sum_{i=1}^{r-1} p_{[i]}^{A} \tag{2}$$

where $d \geq 0$ is a normalizing constant, $\sum_{i=1}^{0} p_{[i]} := 0$ .For convenience, we denote by s_{psd} the p-s-d setup given by Eq. 2 (see Koulamas and Kyparisis [3]).Let C_j be the completion time of job J_j . For a given schedule $\pi = [J_1, J_2, ..., J_n]$, $C_j = C_j(\pi)$ represents the completion time of job J_j .Let $\sum U_j$,where $U_j = 1$ if $C_j > d_j$ and $U_j = 0$ otherwise, $j = 1,2,...,n$, represent the number of tardy jobs of a given permutation, where d_j is the due date of job J_j .Using the conventional notation (Graham et al. [5]),the number of tardy jobs scheduling problem is denoted as $1\left|p_{jr}^{A}(t) = \alpha_j(b+ct)r^a, s_{psd}\right|\sum U_j$.

2 A counterexample

Let N denote the set of jobs already scheduled, N^e be the set of jobs already considered for scheduling but having been discarded because they will not meet their due dates in the optimal schedule, and N^f denote the set of jobs not yet considered for scheduling. The problem $1\left|\ \right|\sum U_j$ is known to be solved by Moore's algorithm [6] as follows:

Moore's algorithm

Step 1 Order the jobs in non-decreasing order of d_j (the earliest due date (EDD) rule).

Step 2 If no jobs in the sequence are late, stop. The schedule is optimal.

Step 3 Find the first late job in the schedule. Denote this job by J_μ .

Step 4 Find a job J_ν with $a_\nu = \max_{i=1,2,...,\mu} a_i$. Remove job J_ν from the schedule and process it after the completion of all the jobs that were processed. Go to Step 2.

Wang et al.[1] gave the following results.

Theorem 1' (Theorem 6, [1]).For the problem $1\left|p_{jr}^{A}(t) = \alpha_j(b+ct)r^a, s_{psd}\right|\sum U_j$,if the jobs have agreeable condition,i.e., $\alpha_j \leq \alpha_k$ implies $d_j \leq d_k$ for all the jobs J_j and J_k ,then an optimal schedule can be obtained by Moore's algorithm.

Theorem 2' (Corollary 5, [1]).For the problem $1\left|p_{jr}^{A}(t) = \alpha_j(b+ct)r^a, s_{psd}, d_j = \theta\alpha_j\right|\sum U_j$, an optimal schedule can be obtained by Moore's algorithm.

The following example shows that Theorem1' and Theorem2' are incorrect.

Counterexample1.Let $\theta = 0.5, \alpha_1 = 1, d_1 = 0.5, \alpha_2 = 3, d_2 = 1.5, \alpha_3 = 5, d_3 = 2.5, a = -3, b = 1, c = 1,$ $d = 0, t = 0$ .Consider a problem containing $n = 3$ jobs, as described in Table 1.

Table 1 Data of problem set.

Job J_j	J_1	J_2	J_3
α_j	1	3	5
d_j	0.5	1.5	2.5

Table 2 The processing information for the sequence $[J_1, J_2, J_3]$.

Job J_j	J_1	J_2	J_3
$p_{jr}^{A}(t)$	1	0.3750	0.1852
C_j	1	1.3750	1.5602
U_j	1	0	0

Table 3 The processing information for the sequence $[J_2, J_3]$.

Job J_j	J_2	J_3
$p_{jr}^{A}(t)$	3	0.6250
C_j	3	3.6250
U_j	1	1

Table 4 The processing information for the sequence $[J_3]$.

Job J_j	J_3
$p_{jr}^{A}(t)$	5
C_j	5
U_j	1

Table 5 The processing information for the sequence $[J_3, J_2, J_1]$.

Job J_j	J_3	J_2	J_1
$p_{jr}^{A}(t)$	5	0.3750	0.0370
C_j	5	5.3750	5.4120
U_j	1	1	1

In this counterexample, $\alpha_j \le \alpha_k$ implies $d_j \le d_k$ for all the jobs J_j and J_k , and the jobs are already indexed by EDD, as required in Step 1 of the Moore's Algorithm. In Step 3, job J_1 is found to be the fifirst late job (see Table 2). In Step 4, job J_1 is removed from N and placed in N^e .In the next pass at Steps 3 and 4, job J_2 is removed from N and placed in N^e (See Table 3). In the next pass at Steps 3 and 4, job J_3 is removed from N and placed in N^e (See Table 4).

Therefore, $N^e = \{J_1, J_2, J_3\}$ .According to Moore's algorithm, any processing sequence is an optimal schedule. However, it can be checked that the objective value of the processing sequence $[J_1, J_2, J_3]$ is 1 (see Table 2)and that the objective value of the processing sequence of $[J_3, J_2, J_1]$ is 3(see Table 5), a contradiction. It follows that Corollary 5 in Wang et al.[1] is incorrect. Since Corollary 5 is a special case of Theorem 6 in Wang et al.[1]. Therefore, Theorem 6 in Wang et al. [1]is also incorrect.

Acknowledgments: This research was supported by the Science and Technology Research Project of Jiangxi Provincial Education Department (Grant No.GJJ191156),the Youth Research Foundation of Pingxiang University (Grant No. 2019D0203).

Declarations

-Ethical Approval:Not applicable.
-Consent to Participate:Not applicable.
-Consent to Publish:Not applicable.
-Authors Contributions:Not applicable.
-Funding:Not applicable.
-Competing Interests:Not applicable.
-Availability of data and materials:Not applicable.

References

1.Ji-Bo Wang,Yong Jiang,Gang Wang(2009) Single-machine scheduling with past-sequence-dependent setup times and effects of deterioration and learning.Int J Adv Manuf Technol,41:1221–1226
2. Wang J-B, Cheng TCE (2007) Scheduling problems with the effects of deterioration and learning. Asia-Pac J Oper Res 24(2):245–261
3. Koulamas C, Kyparisis GJ (2008) Single-machine scheduling problems with past-sequence-dependent setup times. Eur J Oper Res 187:1045–1049
4. Kuo W-H, Yang D-L (2007) Single-machine scheduling with past-sequence-dependent setup and learning effects. Inf Process Lett 102:22–26
5. Graham RL, Lawler EL, Lenstra JK, Rinnooy Kan AHG (1979) Optimization and approximation in deterministic sequencing and scheduling: a survey. Annal Discrete Math 5:287–326
6. Moore J (1968) An n job, one machine sequencing algorithm for minimizing the number of late jobs. Manage Sci 15:102– 109