Applied and Computational Mathematics
2014; 3(3): 85-89
Published online June 10, 2014 (http://www.sciencepublishinggroup.com/j/acm)
doi: 10.11648/j.acm.20140303.13

Single machine scheduling problems with delivery times under simple linear deterioration

Juan Zou[1, 2]

[1]School of Management, Qufu Normal University, Rizhao, China
[2]School of Mathematical Sciences, Qufu Normal University, Qufu, China

Email address:
zoujuanin@163.com

To cite this article:
Juan Zou. Single Machine Scheduling Problems with Delivery Times under Simple Linear Deterioration. *Applied and Computational Mathematics.* Vol. 3, No. 3, 2014, pp. 85-89. doi: 10.11648/j.acm.20140303.13

Abstract: We consider several single machine scheduling problems in which the processing time of a job is a simple linear increasing function of its starting time and each job has a delivery time. The objectives are to minimize the functions about delivery completion times. For the former three problems, we propose polynomial-time algorithms to solve them. For the last problem, we prove that it is NP-hard when all jobs have release dates.

Keywords: Scheduling, Single Machine, Delivery Time, Deteriorating jobs

1. Introduction

In the classical scheduling theory, the processing times of jobs are considered to be constant and independent of their starting times. However, this assumption is not appropriate for the modeling of many modern industrial processes where the processing time of a job may deteriorate while waiting to be processed. Such situations can be found in maintenance scheduling, steel production, cleaning assignment, fire fighting, hospital emergency wards, resource allocation, where any delay in processing a job may increase the time necessary for its completion. When the machine gradually loses efficiency, a job that is processed later requires a longer processing time. Such problems are generally known as scheduling with deterioration effects.

Scheduling deteriorating jobs was first considered by Browne and Yechiali [1] who assumed that the processing times of jobs are non-decreasing, start time dependent linear functions. They provided the optimal solution when the objective is to minimize the expected makespan. In addition, they solved a special case when the objective function is to minimize the total weighted completion time. Mosheiov [2] considered simple linear deterioration where jobs have a fixed job-dependent growth rate but no basic processing time. He showed that most commonly applied performance criteria, such as the makespan, the total flow time, the total lateness, the sum of weighted completion times, the maximum lateness, the maximum tardiness, and the number of tardy jobs, remain polynomial solvable. Since then, machine scheduling problems with time dependent processing times have received increasing attention. Wang et al. [11] considered single machine scheduling problems with deteriorating jobs and resource allocation in a group technoloy environment. They proved that the problems of the weighted combination of makespan and total resource cost minimization remain polynomial solvable under certain conditions. Lee et al. [12] considered a single machine deteriorating job scheduling problem with jobs release times where its objective is to minimize the makespan, they prensented a branch-and-bound algorithm to derive the optimal solution for the problem. Moreover, they proposed easy implemented heuristic algorithms. An extensive survey of different models and problems was provided by Alidaee and Womer [3]. Cheng, Ding and Lin [4] recently presented an updated survey of the results on scheduling problems with time-dependent processing times. Other recent results of scheduling models considering deterioration effects can be found in Lodree and Gerger[7], Cheng and Sun [8], Ji et al. [9] and Wang et al. [10].

In our model, each job J_j must be processed on the machine and then spend an additional amount of time $q_j \geq 0$ being delivered. This delivery can be interpreted as an additional processing requirement on a non-bottleneck machine (a machine that can process an arbitrary number of jobs at once), or as a physical delivery (travel) time; but the key property is that different jobs' deliveries can overlap in time. In a schedule, let C_j denote the completion time of job J_j, thus the delivery completion time of a job J_j is

$C_j + q_j$, we will use D_j to represent $C_j + q_j$, where D_j denote the delivery-completion time of job J_j.

The presentation of this paper is organized as follows. In Section 2, we formulate the problem under consideration and introduce the notation used throughout this paper. In Section 3, we present the main results of the paper. We conclude the paper and discuss the future research in Section 4.

2. Problem Formulation and Notation

The problems considered in this paper can be formally described as follows. Each job is to be processed without interruption on a single machine. The machine can handle only one job at a time. Each job J_j has a positive deteriorating rate b_j, a subsequent nonnegative delivery time q_j and a weight ω_j indicating the relative importance of the job. The actual processing time of job J_j in a schedule is given by $p_j = b_j t$, where t represents its starting time. For the former three problems, all jobs are simultaneously available at time $t_0 > 0$. Note that the assumption $t_0 > 0$ is made here, in order to avoid the trivial case of $t_0 = 0$ (when $t_0 = 0$, the completion time of each job will be 0). The objectives are to minimize the time by which all jobs are delivered, the maximum weighted delivery completion time and the total weighted delivery completion time. Following the three-field notation introduced by Graham et al.[5], the corresponding problems are denoted by

$$1 \mid p_j = b_j t, r_j = t_0, q_j \mid D_{\max},$$

$$1 \mid p_j = b_j t, r_j = t_0, q_j \mid \max \omega_j D_j$$

and $1 \mid p_j = b_j t, r_j = t_0, q_j \mid \sum \omega_j D_j$ respectively. We also consider the case of scheduling deteriorating jobs with release dates on a single machine, the problem is denoted by $1 \mid r_j, p_j = b_j t, q_j \mid D_{\max}$.

3. Main Results

In this section, we start with the following lemma which can be used in what follows.

Lemma 3.1 ([2]) Let σ be a schedule for the single machine scheduling problem $1 \mid p_j = b_j t, r_j = t_0 \mid C_{\max}$, then the completion time of the j th job in σ is $C_j(\sigma) = t_0 \prod_{i=1}^{j} (1 + b_i), j = 1, \ldots, n$.

3.1. Problem $1 \mid p_j = b_j t, r_j = t_0, q_j \mid D_{\max}$

Theorem 3.1 For problem $1 \mid p_j = b_j t, r_j = t_0, q_j \mid D_{\max}$, sorting the jobs by nonincreasing delivery times yields an optimum schedule.

Proof. Consider an optimal schedule σ^*, if σ^* contains jobs are not sequenced in nonincreasing order of delivery times, then we must have a pair of jobs J_i, J_j such that job J_i starts at time S, is followed by job J_j, and $q_i < q_j$. So we have

$$D_i(\sigma^*) = S + b_i S + q_i,$$

$$D_j(\sigma^*) = S + b_i S + b_j(S + b_i S) + q_j$$

Consider a schedule σ which is obtained from σ^* by interchanging jobs J_i and J_j. Under σ, we have

$$D_j(\sigma) = S + b_j S + q_j,$$

$$D_i(\sigma) = S + b_j S + b_i(S + b_j S) + q_i$$

then we obtain that

$$D_j(\sigma^*) > D_j(\sigma),$$

$$D_j(\sigma^*) > D_i(\sigma).$$

This contradicts the optimality of σ^* since all other completion times are unchanged. It follows that sorting the jobs by nonincreasing delivery times yields an optimum schedule for this problem. This completes the proof.

3.2. Problem $1 \mid p_j = b_j t, r_j = t_0, q_j \mid \max \omega_j D_j$

The complexity of problem $1 \mid p_j = b_j t, r_j = t_0, q_j \mid \max \omega_j D_j$ remains open. However, each of the following two cases is polynomial solvable:

Theorem 3.2 If $q_j = q$ for all $j = 1, \ldots, n$, we get an optimal schedule by applying the following rule: schedule jobs in order of nonincreasing weights.

Proof. We can obtain the correctness of Theorem 3.2 by using simple interchange arguments.

Example 3.1

There are five jobs J_1, J_2, J_3, J_4, J_5, the jobs are scheduling on the single machine in order of nonincreasing weights.The delivery time $q_j = 5$ for all $j = 1, \ldots, n$, the deteriorating rates and weights of jobs are given as follows , where $t_0 = 1$.

$$J_1 : b_1 = 2, w_1 = 15,$$

$$J_2 : b_2 = 1, w_2 = 10,$$

$$J_3 : b_3 = 1, w_3 = 8,$$

$$J_4 : b_4 = 2, w_4 = 5,$$

$$J_5 : b_5 = 3, w_5 = 1,$$

Next, we compute the weighted delivery completion times of jobs.

$$C_1 = t_0(1+b_1) = 1\times 3 = 3,$$

$$w_1 D_1 = w_1(C_1+q_1) = 15\times(3+5) = 120,$$

$$C_2 = t_0(1+b_1)(1+b_2) = 6,$$

$$w_2 D_2 = w_2(C_2+q_2) = 10\times(6+5) = 110,$$

$$C_3 = t_0(1+b_1)(1+b_2)(1+b_3) = 12,$$

$$w_3 D_3 = w_3(C_3+q_3) = 8\times(12+5) = 136,$$

$$C_4 = t_0(1+b_1)(1+b_2)(1+b_3)(1+b_4) = 36,$$

$$w_4 D_4 = w_4(C_4+q_4) = 5\times(36+5) = 205,$$

$$C_5 = t_0(1+b_1)(1+b_2)(1+b_3)(1+b_4)(1+b_5) = 144,$$

$$w_5 D_5 = w_5(C_5+q_5) = 1\times(144+5) = 149,$$

Then, for this case, we obtain the optimal value of the problem is $\max \omega_j D_j = w_4 D_4 = 205$.

Theorem 3.3 Delivery times are distinct, scheduling jobs in order of nonincreasing weights, and if jobs satisfy the following inequality: $\omega_1 q_1 \ge \cdots \ge \omega_n q_n$, then we get an optimal schedule.

Proof. Consider an optimal schedule σ^*, if σ^* contains jobs are not sequenced in nonincreasing order of weights, then we must have a pair of jobs J_i, J_j such that job J_i starts at time S, is followed by job J_j, and $\omega_i < \omega_j, \omega_i q_i < \omega_j q_j$. So we have

$$\omega_i D_i(\sigma^*) = S\omega_i(1+b_i) + \omega_i q_i,$$

$$\omega_j D_j(\sigma^*) = S\omega_j(1+b_i)(1+b_j) + \omega_j q_j.$$

Consider a schedule σ which is obtained from σ^* by interchanging jobs J_i and J_j. Under σ, we have

$$\omega_j D_j(\sigma) = S\omega_j(1+b_j) + \omega_j q_j,$$

$$\omega_i D_i(\sigma) = S\omega_i(1+b_i)(1+b_j) + \omega_i q_i$$

then we obtain that

$$\omega_j D_j(\sigma^*) > \omega_j D_j(\sigma),$$

$$\omega_j D_j(\sigma^*) > \omega_i D_i(\sigma).$$

This contradicts the optimality of σ^* since all other delivery completion times are unchanged. It follows that sorting the jobs by nonincreasing weights, if the jobs satisfy the following inequality: $\omega_1 q_1 \ge \cdots \ge \omega_n q_n$, yields an optimum schedule for this problem. This completes the proof.

3.3. Problem $1 \mid p_j = b_j t, r_j = t_0, q_j \mid \sum \omega_j D_j$

Theorem 3.4 Scheduling the jobs in a nondecreasing order of the ratio $\frac{b_i}{(1+b_i)\omega_i}$ gives an optimal schedule for problem $1 \mid p_j = b_j t, r_j = t_0, q_j \mid \sum \omega_j D_j$.

Proof. Consider an optimal schedule σ^*, if σ^* contains jobs are not sequenced in non-decreasing order of the ratio $\frac{b_i}{(1+b_i)\omega_i}$, then we must have a pair of jobs J_i, J_j such that job J_i starts at time S, and is followed by job J_j, and $\frac{b_i}{(1+b_i)\omega_i} > \frac{b_j}{(1+b_j)\omega_j}$. So we have

$$D_i(\sigma^*) = S + b_i S + q_i,$$

$$D_j(\sigma^*) = S + b_i S + b_j(S + b_i S) + q_j$$

Consider a schedule σ which is obtained from σ^* by interchanging jobs J_i and J_j. Under σ, we have

$$D_j(\sigma) = S + b_j S + q_j,$$

$$D_i(\sigma) = S + b_j S + b_i(S + b_j S) + q_i$$

then we obtain that

$$\omega_i D_i(\sigma^*) + \omega_j D_j(\sigma^*) - \omega D_i(\sigma) + \omega_j D_j(\sigma)$$

$$= \omega_i(S(1+b_i)+q_i) + \omega_j(S(1+b_i)(1+b_j)+q_j)$$

$$-\omega_i(S(1+b_i)(1+b_j)+q_i) - \omega_j(S(1+b_j)+q_j)$$

$$= S(\omega_j(1+b_j)b_i - \omega_i(1+b_i)b_j)$$

$$= S(1+b_i)(1+b_j)\omega_i\omega_j\left(\frac{b_i}{(1+b_i)\omega_i} - \frac{b_j}{(1+b_j)\omega_j}\right) > 0$$

This contradicts the optimality of σ^* since all other jobs' function values are unchanged. It follows that sorting the jobs by nondecreasing order of the ratio $\frac{b_i}{(1+b_i)\omega_i}$ yields an optimum schedule for a single machine. This completes the proof.

Corollary 3.1 If $\omega_j = 1, j = 1,\ldots,n$, arranging the jobs in a nondecreasing order of deterioration rates provides an optimal schedule for problem $1 \mid p_j = b_j t, r_j = t_0, q_j \mid \sum D_j$.

3.4. Problem $1 \mid r_j, p_j = b_j t, q_j \mid D_{\max}$

Theorem 3.5 The problem $1 \mid r_j, p_j = b_j t, q_j \mid D_{\max}$ is NP-hard.

Proof. The problem $1 \mid r_j, p_j = b_j t, q_j \mid D_{\max}$ is clearly in NP. We reduce the Equal Products Problem, which is NP-hard (see [6]), to $1 \mid r_j, p_j = b_j t, q_j \mid D_{\max}$.

The Equal Products Problem is defined as follows: Given a set of n positive integers $a_1, a_2, \ldots, a_n$ such that $\prod_{j=1}^{n} a_j = A^2$, does there exist a subset

$$S_1 \subseteq S = \{1,2,\ldots,n\} \text{ such that } \prod_{j\in S_1} a_j = A \text{ ?}$$

Given an arbitrary instance of the Equal Products Problem, we construct an instance of the scheduling problem as follows: there are $n+1$ jobs, their deteriorating rates, delivery times and release dates are given by

$$b_j = a_j - 1, q_j = A^2, r_j = 1, j = 1,2,\cdots,n$$

$$b_{n+1} = A - 1, q_{n+1} = A^3, r_{n+1} = A$$

The threshold value is given by $Y = A^3 + A^2$.

Clearly, the above reduction can be done in time polynomial-time. We show that the Equal Products Problem has a solution if and only if there is a schedule π of the scheduling problem such that $D_{\max} \le Y$.

First, we assume that the Equal Products Problem has a solution S_1 such that $\prod_{j\in S_1} a_j = A$. We assign the jobs in $\{J_j : j \in S_1\}$ to be processed on the machine before J_{n+1} and all other jobs are processed on the machine after J_{n+1}. It is easy to see that $D_{\max} = A^3 + A^2 = Y$.

Conversely, suppose that there is a schedule π such that $D_{\max} \le Y$. We need to show that the Equal Products Problem has a solution.

Claim 1. The starting time of job J_{n+1} is equal to A.

Since the release date of J_{n+1} is A, the starting time of job J_{n+1} is at least A. If the starting time of job J_{n+1} is greater than A strictly,

$$D_{n+1} > A(1 + b_{n+1}) + q_{n+1}$$
$$= A^2 + A^3$$
$$= Y$$

it is a contradiction. Claim 1 follows.

Let $S_1 = \{j \mid C_j \le A\}$, $S_2 = \{j \mid t_j \ge A^2\}$, where C_j, t_j denote the completion time and the starting time of J_j respectively. Obviously,

$$\prod_{j\in S_1}(1+b_j) = \prod_{j\in S_1} a_j \le A.$$

Since

$$D_{\max} = A^2 \prod_{j\in S_2}(1+b_j) + A^2$$
$$= A^2 \prod_{j\in S_2} a_j + A^2$$
$$\le A^3 + A^2,$$

then

$$\prod_{j\in S_2} a_j \le A, \quad \prod_{j\in S_1} a_j \cdot \prod_{j\in S_2} a_j = A^2.$$

Hence, we have

$$\prod_{j\in S_1} a_j = \prod_{j\in S_2} a_j = A.$$

Thus, S_1 is a solution of the Equal Products Problem. This completes the proof.

Next, the special case of the problem is polynomial solvable.

Corollary 3.2 If $q_j = q, j = 1,\ldots,n$, then problem $1 \mid r_j, p_j = b_j t, q \mid D_{\max}$ is solved optimally in polynomial time in order of nondecreasing release times.

4. Conclusions

In this paper, we focus on single machine scheduling problems with delivery times under simple linear deterioration, the objectives are to minimize maximum (weight) delivery completion time, the total (weight) delivery completion time, respectively. When the release dates of all jobs are identical, we provide an optimal $O(nlogn)$ time algorithm for them or special cases. When the release dates of all jobs are distinct, we showed the problem $1 \mid r_j, p_j = b_j t, q_j \mid D_{\max}$ is NP-hard. For future research, it would be interesting to focus on scheduling deteriorating jobs with other objectives. Analysis of scheduling problems with jobs of more general deterioration types is another worthy topic.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (11201259), the Special Funds of the National Natural Science Foundation of China (61340045) and the Doctoral Fund of the Ministry of Education (20123705120001&20123705110003).

References

[1] S.Browne, U.Yechiali, Scheduling deteriorating jobs on a single processor, *Operations Research,* vol.38, no.3, pp.495-498, 1990.

[2] G.Mosheiov, Scheduling jobs under simple linear deterioration, *Computers Operations Research,* vol.21, no.6, pp.653-659, 1994.

[3] B.Alidaee, N.K.Womer, Scheduling with time dependent processing times: Review and extension, *Journal of the Operational Research Society*, vol.50, no.7, pp.711-720, 1999.

[4] T.C.E.Cheng, Q.Ding, B.M.T.Lin, A concise survey of scheduling with time-dependent processing times, *European Journal of Operational Research,* vol.152, no.1, pp.1-13, 2004.

[5] R.L.Graham, E.L.Lawler, J.K.Lenstra, A.H.G.Rinnooy Kan, Optimization and approximation in deterministic sequencing and scheduling: A survey, *Annals of Discrete Mathematics,* vol.5, pp. 287-326, 1979.

[6] S.Gawiejnowicz, Time-Dependent Scheduling, Springer, Berlin, 2008.

[7] E.Lodree, C.Gerger, A note on the optimal sequence position for a rate-modifying activity under simple linear deterioration, *European Journal of Operational Research*, vol.201, no.2, pp.644–648, 2010.

[8] Y.Cheng, S.Sun, Scheduling linear deteriorating jobs with rejection on a single machine, *European Journal of Operational Research,* vol.194, no.1, pp.18–27, 2009.

[9] M.Ji, C.J. Hsu, D.L. Yang, Single-machine scheduling with deteriorating jobs and aging effects under an optional maintenance activity consideration, *Journal of Combinatorial Optimization*, vol.26, no.3, pp.437-447,2013.

[10] X.R.Wang, J.J.Wang, Single-machine scheduling with convex resource dependent processing times and deteriorating jobs, *Applied Mathematical Modelling,* vol. 37, no. 4, pp. 2388-2393, 2013.

[11] D.Wang, Y. Huo, P.Ji, Single-machine group scheduling with deteriorating jobs and allotted resource, Optimization Letters, vol. 8, no.2.pp.591-605, 2014.

[12] W.C.Lee, C.C.Wu, Y.H.Chung, Scheduling deteriorating jobs on a single machine with release times, Journal Computers and Industrial Engineering, vol. 54, no.3, pp.441-452,2008.