SINGLE-MACHINE SCHEDULING PROBLEMS WITH PAST-SEQUENCE-DEPENDENT DELIVERY TIMES AND DETERIORATION AND LEARNING EFFECTS SIMULTANEOUSLY

SUH-JENQ YANG[1], JIA-YUARN GUO[1], HSIN-TAO LEE[1]
AND DAR-LI YANG[2]

[1]Department of Industrial Management
Nan Kai University of Technology
No. 568, Zhongzheng Rd., Caotun Township, Nantou County 54243, Taiwan
{ sjyang; jxg87; htlee }@nkut.edu.tw

[2]Department of Information Management
National Formosa University
No. 64, Wunhua Rd., Huwei Township, Yun Lin 632, Taiwan
dlyang@nfu.edu.tw

ABSTRACT. *This paper investigates single-machine scheduling problems with simultaneous considerations of past-sequence-dependent delivery times and the effects of deterioration and learning. The job delivery time is assumed to be proportional to the sum of processing times of all already scheduled jobs. We show that the makespan and the total completion time minimization problems can be optimally solved in polynomial time. We also prove that the total weighted completion time, the maximum lateness, and the maximum tardiness minimization problems remain polynomially solvable with agreeable conditions.*
Keywords: Scheduling, Learning effect, Job deterioration, Past-sequence-dependent delivery times

1. **Introduction.** In the last years, scheduling problems with the past-sequence-dependent (*p-s-d*) setup times and delivery times have been extensively studied by researchers. The *p-s-d* setup time and the *p-s-d* delivery time were probably first introduced in scheduling by Koulamas and Kyparisis [10,11]. They assumed that the job setup time (delivery time) is assumed to be proportional to the sum of processing times of all already scheduled jobs.

On the other hand, the job processing time may change due to various factors such as the deterioration effect or the learning effect. The deterioration effect in scheduling can be defined as a job which takes more time when processed later than when processed earlier, while in scheduling with the learning effect, the actual processing time of a job is shorter if it is scheduled later in a sequence.

Although the scheduling problems with the *p-s-d* delivery times and the deterioration effect and the learning effect have been separately studied in the literature, they have never been simultaneously considered. Job deterioration and learning may co-exist in many realistic scheduling situations. For related works, the reader may refer to [21,22]. The motivation for this study stems from the metal hot forging process that presses an ingot to create various size and shape in a forging machine. Before the pressing process, the ingot needs to preheat to the required temperature. Generally, the ingots

are preheating in a batch processor. Thus, the longer the ingot waits for the processing on the forging machine, the lower the temperature drops and the longer the processing time needs [1,4,14,15]. On the other hand, the forging machine is operated by a skilled worker. The worker learns how to produce more efficiently during the process. As a result, deteriorating jobs and the learning effect simultaneously exist in the metal forging process [19]. In addition, after the produce has been processed by the forging machine, it should be exposed on an environment to drop its temperature and take a post-processing operation before it is delivered to the customer. Consequently, this paper addresses the problems of single-machine scheduling with p-s-d delivery times and the deterioration and learning effects simultaneously. The performance measures include the makespan, the total completion time, the total weighted completion time, the maximum lateness, and the maximum tardiness.

The remaining part of this paper is organized as follows. In Section 2, a literature review is given. We formulate the model of the paper in Section 3. In Section 4, several single-machine scheduling problems are examined. In Section 5, we give a numerical example for solving the problem under study. The last section presents the conclusions.

2. **Literature Review.** In two recent papers, Koulamas and Kyparisis [10,11] proposed the concepts of p-s-d setup times and p-s-d delivery times in machine scheduling problems, respectively. Koulamas and Kyparisis [10] considered single-machine scheduling problems with p-s-d setup times for minimizing the makespan, the total completion time, and the total absolute differences in completion times. They assumed that the job setup time is dependent on all already scheduled jobs. Their study showed that all the problems studied are polynomially solvable. They also extended their results to non-linear p-s-d setup times. Biskup and Herrmann [3] extended the study of Koulamas and Kyparisis [10] to the problems with due dates to minimize the total lateness, the total tardiness, the maximum lateness, and the maximum tardiness problems. They proved that these problems remain polynomially solvable with agreeable due dates. Later, Koulamas and Kyparisis [11] stressed single-machine scheduling problems with p-s-d job delivery times. They assumed that the job delivery time is needed to remove any waiting time-induced adverse effects on the job's condition prior to delivering it to the customer. For example, an electronic component may be exposed to certain electromagnetic and/or radioactive fields while waiting in the machine's pre-processing area and regulatory authorities require the component to be "treated" for an amount of time proportional to the job's exposure time to these fields. This treatment can be performed after the component has been processed by the machine but before it is delivered to the customer [11]. Such a post-processing operation is usually called the job "delivery time". They further assumed that the job delivery time is to be proportional to the job's waiting time in order to model the mandated post-processing job treatment. They proved that all the problems studied can be optimally solved in polynomial time algorithms when the objective functions were the makespan, the total completion time, the maximum lateness, and the number of tardy jobs. Yang et al. [20] considered single-machine scheduling problems with the p-s-d delivery times and the learning effect. They first showed that the makespan, the total completion time, and the total waiting time minimization problems can be solved in polynomial time. They also showed that the total weighted completion time and the total weighted waiting time minimization problems remain polynomially solvable under certain conditions.

On the other hand, scheduling with deteriorating jobs and/or learning effects has been widely studied. For more details on scheduling with learning effects, the reader may refer to the recent surveys by Biskup [2] and Janiak and Rudek [9], while scheduling problems

with time-dependent job processing times are discussed in the book by Gawiejnowicz [7]. In addition, Kuo and Yang [12] first investigated single-machine scheduling problems with p-s-d setup times and the learning effect. They proved that the makespan, the total absolute differences in completion times, and the sum of earliness, tardiness and common due-date penalties minimization problems are polynomially solvable. Wang [17] considered the time-dependent learning model with p-s-d setup times on a single-machine to minimize the makespan, the total completion time, the sum of the quadratic job completion times, the total weighted completion time, and the maximum lateness. Wang et al. [18] studied single-machine scheduling problems with p-s-d setup times and the effects of deterioration and learning. They proved that the minimization problems of the makespan, the total completion time, and the sum of the δth ($\delta \geq 0$) power of job completion times can be solved by the smallest deterioration rate (SDR) rule, respectively. Cheng et al. [5] introduced a new scheduling model in which deteriorating and learning effects and p-s-d setup times are considered simultaneously. They derived polynomial time optimal solutions for some single-machine problems with or without the presence of certain conditions. Mani et al. [13] focused on the parametric analysis for a single-machine scheduling problem with p-s-d setup times to minimize the total absolute differences in completion times. Cheng et al. [6] proposed a scheduling model in which the actual processing time of a job is a function of the logarithm of the total processing time of all already scheduled jobs and the setup times are proportional to the actual processing times of the already scheduled jobs. Under the proposed model, they provided optimal solutions for some single-machine scheduling problems.

3. **Notation and Model Formulation.** There are n independent jobs $J = \{J_1, J_2, \ldots, J_n\}$ available at time zero which have to be processed on a single-machine. Preemption is not allowed and the machine is only able to process one job at a time. Each job J_j has a normal processing time p_j, $j = 1, 2, \ldots, n$. For any schedule S, let $J_{[j]}$ denote the job in the jth position in S, and $p_{[j]}$ and $C_{[j]}$ denote the normal processing time and the completion time of job $J_{[j]}$ in S, respectively. Due to the effects of deterioration and learning, the actual processing time of job J_j when scheduled in position r is given by

$$p_{jr} = p_j \left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a r^b, \quad j, r = 1, 2, \ldots, n \tag{1}$$

where $a \geq 1$ and $b < 0$ are, respectively, the deteriorating factor and the learning factor.

Following Koulamas and Kyparisis [11], the processing of job $J_{[j]}$ must be followed by a p-s-d delivery time $q_{[j]}$, which can be computed as

$$q_{[j]} = \gamma W_{[j]} = \gamma \sum_{i=1}^{j-1} p_{i[i]}, \quad j = 2, 3, \ldots, n \text{ and } q_{[1]} = 0, \tag{2}$$

where $\gamma \geq 0$ is a normalizing constant, $W_{[j]}$ denotes the waiting time of job $J_{[j]}$, and $p_{i[i]}$ denotes the actual processing time of job J_i scheduled in the ith position in a schedule. Obviously, on a single-machine setting with a continuously available machine and all the jobs simultaneously available at time zero, $W_{[1]} = 0$ and $W_{[j]} = \sum_{i=1}^{j-1} p_{i[i]}$, $j = 2, 3, \ldots, n$. In addition, it is assumed that the post-processing operation of any job $J_{[j]}$ modeled by its delivery time $q_{[j]}$ is performed "off-line", consequently, it is not affected by the availability of the machine and it can commence immediately upon completion of the main operation

resulting in $C_{[1]} = p_{[1]}$ and

$$C_{[j]} = W_{[j]} + p_{j[j]} + q_{[j]} = (1 + \gamma) \sum_{i=1}^{j-1} p_{i[i]} + p_j \left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a j^b, \quad j = 2, 3, \ldots, n. \quad (3)$$

For convenience, we denote the p-s-d delivery time given in Equation (2) by q_{psd}. We consider the minimization of the following objective functions: the makespan $C_{\max} = \max_{j=1,2,\ldots,n} \{C_j\}$, the total completion time $\sum_{j=1}^{n} C_j$, the total weighted completion time $\sum_{j=1}^{n} \alpha_j C_j$, the maximum lateness $L_{\max} = \max_{j=1,2,\ldots,n} \{C_j - d_j\}$, where d_j is the due-date of job J_j, and the maximum tardiness $T_{\max} = \max\{0, L_{\max}\}$. Following Graham et al. [8], we denote the corresponding scheduling problems as $1/p_{jr}, q_{psd}/C_{\max}$, $1/p_{jr}, q_{psd}/\sum C_j$, $1/p_{jr}, q_{psd}/\sum \alpha_j C_j$, $1/p_{jr}, q_{psd}/L_{\max}$, and $1/p_{jr}, q_{psd}/T_{\max}$, respectively.

4. **Problems Analysis.** In this section, we first show that the makespan and the total completion time minimization problems are polynomially solvable. We then show that the total weighted completion time, the maximum lateness, and the maximum tardiness minimization problems remain polynomially solvable with agreeable conditions.

Before presenting the main results, we introduce two useful lemmas that help find the optimal schedule for the problem under study.

Lemma 4.1. $\left[1 + ax_1 t(1+t)^{a-1} \left(\frac{r+1}{r}\right)^b\right] - \left[x_2(1+t)^a \left(\frac{r+1}{r}\right)^b\right] \geq 0$ *if* $0 < x_2 \leq x_1 \leq 1$, $a \geq 1$, $b < 0$, $t > 0$, *and* $r = 1, 2, \ldots, n - 1$.

Proof: Let $g(t) = \left[1 + ax_1 t(1+t)^{a-1} \left(\frac{r+1}{r}\right)^b\right] - \left[x_2(1+t)^a \left(\frac{r+1}{r}\right)^b\right]$. Taking the first derivative of $g(t)$ with respect to t, we obtain

$$g'(t) = ax_1(1+t)^{a-1}\left(\frac{r+1}{r}\right)^b + a(a-1)x_1 t(1+t)^{a-2}\left(\frac{r+1}{r}\right)^b - ax_2(1+t)^{a-1}\left(\frac{r+1}{r}\right)^b.$$

Since $0 < x_2 \leq x_1 \leq 1$, $a \geq 1$, $b < 0$, $t > 0$, and $r = 1, 2, \ldots, n - 1$, we have $g'(t) \geq 0$ and $g(t) > g(0) = 1 - x_2\left(\frac{r+1}{r}\right)^b \geq 0$. Hence, $g(t)$ is non-decreasing on $0 < x_2 \leq x_1 \leq 1$, $a \geq 1$, $b < 0$, $t > 0$, and $r = 1, 2, \ldots, n - 1$.

Thus, $\left[1 + ax_1 t(1+t)^{a-1} \left(\frac{r+1}{r}\right)^b\right] - \left[x_2(1+t)^a \left(\frac{r+1}{r}\right)^b\right] \geq 0$ if $0 < x_2 \leq x_1 \leq 1$, $a \geq 1$, $b < 0$, $t > 0$, and $r = 1, 2, \ldots, n - 1$.

Lemma 4.2. $\left[\lambda + x_1(1+\lambda t)^a \left(\frac{r+1}{r}\right)^b\right] - \left[1 + \lambda x_2(1+t)^a \left(\frac{r+1}{r}\right)^b\right] \geq 0$ *if* $0 < x_2 \leq x_1 \leq 1$, $\lambda \geq 1$, $a \geq 1$, $b < 0$, $t > 0$, *and* $r = 1, 2, \ldots, n - 1$.

Proof: Let $f(\lambda) = \left[\lambda + x_1(1+\lambda t)^a \left(\frac{r+1}{r}\right)^b\right] - \left[1 + \lambda x_2(1+t)^a \left(\frac{r+1}{r}\right)^b\right]$. Taking the first and second derivatives of $f(\lambda)$ with respect to λ, we have

$$f'(\lambda) = \left[1 + ax_1 t(1+\lambda t)^{a-1}\left(\frac{r+1}{r}\right)^b\right] - \left[x_2(1+t)^a\left(\frac{r+1}{r}\right)^b\right]$$

and

$$f''(\lambda) = a(a-1)x_1 t^2(1+\lambda t)^{a-2}\left(\frac{r+1}{r}\right)^b.$$

Since $0 < x_2 \leq x_1 \leq 1$, $\lambda \geq 1$, $a \geq 1$, $b < 0$, $t > 0$, and $r = 1, 2, \ldots, n - 1$, we have $f''(\lambda) \geq 0$. Hence, $f'(\lambda)$ is non-decreasing on $0 < x_2 \leq x_1 \leq 1$, $\lambda \geq 1$, $a \geq 1$, $b < 0$, $t > 0$,

and $r = 1, 2, \ldots, n - 1$. In addition, by Lemma 4.1, we obtain

$$f'(\lambda) \geq f'(1) = \left[1 + ax_1 t(1 + t)^{a-1} \left(\frac{r+1}{r}\right)^b\right] - \left[x_2(1 + t)^a \left(\frac{r+1}{r}\right)^b\right] \geq 0,$$

for $0 < x_2 \leq x_1 \leq 1$, $\lambda \geq 1$, $a \geq 1$, $b < 0$, $t > 0$, and $r = 1, 2, \ldots, n - 1$. Hence, $f(\lambda)$ is non-decreasing on $0 < x_2 \leq x_1 \leq 1$, $\lambda \geq 1$, $a \geq 1$, $b < 0$, $t > 0$, and $r = 1, 2, \ldots, n - 1$.

Thus, $\left[\lambda + x_1(1 + \lambda t)^a \left(\frac{r+1}{r}\right)^b\right] - \left[1 + \lambda x_2(1 + t)^a \left(\frac{r+1}{r}\right)^b\right] \geq 0$ if $0 < x_2 \leq x_1 \leq 1$, $\lambda \geq 1$, $a \geq 1$, $b < 0$, $t > 0$, and $r = 1, 2, \ldots, n - 1$.

4.1. The $1/p_{jr}$, $q_{psd}/C_{\max}$ problem.

Theorem 4.1. *For the $1/p_{jr}$, $q_{psd}/C_{\max}$ problem, an optimal schedule can be obtained by sequencing the jobs in the smallest normal processing time first (SPT) rule.*

Proof: This theorem can be proved by a pair-wise interchange of jobs. Let S_1 and S_2 be two job schedules where the difference between S_1 and S_2 is a pair-wise interchange of two adjacent jobs J_j and J_k. That is, $S_1 = (\pi_1, J_j, J_k, J_l, \pi_2)$ and $S_2 = (\pi_1, J_k, J_j, J_l, \pi_2)$, where π_1 and π_2 are partial sequences and π_1 and π_2 may be empty. We assume that jobs J_j, J_k and J_l are scheduled in positions r, $(r + 1)$ and $(r + 2)$ in S_1, respectively. To show S_1 dominates S_2, it suffices to show that $C_k(S_1) \leq C_j(S_2)$.

Suppose that $p_j \leq p_k$. By definition, the completion times of jobs J_j and J_k in S_1 and jobs J_k and J_j in S_2 are, respectively, given by

$$C_j(S_1) = (1 + \gamma) \sum_{i=1}^{r-1} p_{i[i]} + p_j \left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a r^b, \tag{4}$$

$$C_k(S_1) = (1+\gamma) \sum_{i=1}^{r-1} p_{i[i]} + (1+\gamma)p_j \left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a r^b + p_k \left(1 + \sum_{i=1}^{r-1} p_{[i]} + p_j\right)^a (r+1)^b, \tag{5}$$

$$C_k(S_2) = (1 + \gamma) \sum_{i=1}^{r-1} p_{i[i]} + p_k \left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a r^b, \tag{6}$$

and

$$C_j(S_2) = (1+\gamma) \sum_{i=1}^{r-1} p_{i[i]} + (1+\gamma)p_k \left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a r^b + p_j \left(1 + \sum_{i=1}^{r-1} p_{[i]} + p_k\right)^a (r+1)^b. \tag{7}$$

Taking the difference between Equations (5) and (7), it is derived that

$$C_j(S_2) - C_k(S_1) = \gamma(p_k - p_j) \left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a r^b + (p_k - p_j) \left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a r^b$$

$$+ p_j \left(1 + \sum_{i=1}^{r-1} p_{[i]} + p_k\right)^a (r+1)^b - p_k \left(1 + \sum_{i=1}^{r-1} p_{[i]} + p_j\right)^a (r+1)^b. \tag{8}$$

Let $\lambda = \frac{p_k}{p_j} \geq 1$, $x = \left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a \geq 1$, and $t = \frac{p_j}{\left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)} > 0$. Then, we have that

$$C_j(S_2) - C_k(S_1) = \gamma p_j(\lambda - 1)xr^b$$

$$+ p_j x r^b \left\{\left[\lambda + (1 + \lambda t)^a \left(\frac{r+1}{r}\right)^b\right] - \left[1 + \lambda(1 + t)^a \left(\frac{r+1}{r}\right)^b\right]\right\}. \tag{9}$$

Clearly, $\gamma p_j (\lambda - 1) x r^b \geq 0$. Hence, by Lemma 4.2, we obtain $C_j(S_2) \geq C_k(S_1)$.

Therefore, repeating this interchange argument for all the jobs which are not sequenced in the *SPT* rule yields Theorem 4.1.

4.2. The $1/p_{jr}, q_{psd}/\sum C_j$ problem.

Theorem 4.2. *For the $1/p_{jr}, q_{psd}/\sum C_j$ problem, an optimal schedule can be obtained by sequencing the jobs in the SPT rule.*

Proof: We still use the same notations as in the proof of Theorem 4.1. Suppose that $p_j \leq p_k$. To show S_1 dominates S_2, it suffices to show that (i) $w_l(S_1) \leq w_l(S_2)$ and (ii) $C_j(S_1) + C_k(S_1) \leq C_k(S_2) + C_j(S_2)$.

By definition, the waiting times of job J_l in sequences S_1 and S_2 are, respectively, given by

$$w_l(S_1) = \sum_{i=1}^{r-1} p_{i[i]} + p_j \left(1 + \sum_{i=1}^{r-1} p_{[i]} \right)^a r^b + p_k \left(1 + \sum_{i=1}^{r-1} p_{[i]} + p_j \right)^a (r+1)^b \qquad (10)$$

and

$$w_l(S_2) = \sum_{i=1}^{r-1} p_{i[i]} + p_k \left(1 + \sum_{i=1}^{r-1} p_{[i]} \right)^a r^b + p_j \left(1 + \sum_{i=1}^{r-1} p_{[i]} + p_k \right)^a (r+1)^b. \qquad (11)$$

Then the difference between $w_l(S_2)$ and $w_l(S_2)$ is

$$w_l(S_2) - w_l(S_1) = (p_k - p_j) \left(1 + \sum_{i=1}^{r-1} p_{[i]} \right)^a r^b$$
$$+ p_j \left(1 + \sum_{i=1}^{r-1} p_{[i]} + p_k \right)^a (r+1)^b - p_k \left(1 + \sum_{i=1}^{r-1} p_{[i]} + p_j \right)^a (r+1)^b. \qquad (12)$$

By the proof of Theorem 4.1, we have $w_l(S_1) \leq w_l(S_2)$.

Furthermore, by the proof of Theorem 4.1, we have that $C_j(S_2) - C_k(S_1) \geq 0$ and $C_k(S_2) - C_j(S_1) = (p_k - p_j) \left(1 + \sum_{i=1}^{r-1} p_{[i]} \right)^a r^b \geq 0$. Hence, $C_k(S_2) + C_j(S_2) \geq C_j(S_1) + C_k(S_1)$.

Therefore, repeating this interchange argument for all the jobs which are not sequenced in the *SPT* rule yields Theorem 4.2.

4.3. The $1/p_{jr}, q_{psd}/\sum \alpha_j C_j$ problem.

Theorem 4.3. *For the $1/p_{jr}, q_{psd}/\sum \alpha_j C_j$ problem, if $\frac{p_k}{p_j} \geq 1 \geq \frac{\alpha_k}{\alpha_j}$ for any two jobs J_j and J_k, $j, k = 1, 2, \ldots, n$, an optimal schedule can be obtained by sequencing the jobs in a non-decreasing order of p_j/α_j (i.e., the WSPT rule).*

Proof: Here, we still use the same notations as in the proof of Theorem 4.1. To show S_1 dominates S_2, it suffices to show that (i) $w_l(S_1) \leq w_l(S_2)$ and (ii) $\alpha_j C_j(S_1) + \alpha_k C_k(S_1) \leq \alpha_k C_k(S_2) + \alpha_j C_j(S_2)$.

The proof of part (i) is given in Theorem 4.2. In addition, by the proof of Theorem 4.1, we have that

$$[\alpha_k C_k(S_2) + \alpha_j C_j(S_2)] - [\alpha_j C_j(S_1) + \alpha_k C_k(S_1)]$$

$$= (\alpha_j p_k + \alpha_k p_j)\left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a r^b + (\alpha_k + \alpha_j)(p_k - p_j)\left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a r^b$$

$$+ \alpha_j p_j \left(1 + \sum_{i=1}^{r-1} p_{[i]} + p_k\right)^a (r+1)^b - \alpha_k p_k \left(1 + \sum_{i=1}^{r-1} p_{[i]} + p_j\right)^a (r+1)^b. \tag{13}$$

Let $x_1 = \frac{\alpha_j}{\alpha_j + \alpha_k}$, $x_2 = \frac{\alpha_k}{\alpha_j + \alpha_k}$, $\lambda = \frac{p_k}{p_j}$, $x = \left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a \geq 1$, and $t = \frac{p_j}{\left(1+\sum_{i=1}^{r-1} p_{[i]}\right)} > 0$. If $\frac{p_k}{p_j} \geq 1 \geq \frac{\alpha_k}{\alpha_j}$, then $0 < x_2 \leq x_1 \leq 1$ and $\lambda \geq 1$. Thus,

$$[\alpha_k C_k(S_2) + \alpha_j C_j(S_2)] - [\alpha_j C_j(S_1) + \alpha_k C_k(S_1)]$$

$$= (\alpha_j \lambda + \alpha_k)p_j x r^b + (\alpha_k + \alpha_j)p_j x r^b \left\{ \left[\lambda + x_1(1 + \lambda t)^a \left(\frac{r+1}{r}\right)^b \right] \right.$$

$$\left. - \left[1 + \lambda x_2(1 + t)^a \left(\frac{r+1}{r}\right)^b \right] \right\}. \tag{14}$$

Clearly, $(\alpha_j \lambda + \alpha_k)p_j x r^b > 0$. Hence, by Lemma 4.2, we have $\alpha_j C_j(S_1) + \alpha_k C_k(S_1) \leq \alpha_k C_k(S_2) + \alpha_j C_j(S_2)$.

Therefore, we have that the optimal schedule can be obtained by sequencing the jobs in a non-decreasing order of p_j/α_j.

4.4. The $1/p_{jr}$, $q_{psd}/L_{\max}$ problem.

Theorem 4.4. *For the $1/p_{jr}$, $q_{psd}/L_{\max}$ problem, if jobs have agreeable due-dates, i.e., $p_j \leq p_k$ implies $d_j \leq d_k$ for any jobs J_j and J_k, $j, k = 1, 2, \ldots, n$, an optimal schedule can be obtained by sequencing the jobs in a non-decreasing order of d_j (i.e., Earliest Due-Date rule, EDD rule).*

Proof: We still use the same notations as in the proof of Theorem 4.1. Then the lateness of jobs J_j and J_k in S_1 and jobs J_k and J_j in S_2 are respectively given by

$$L_j(S_1) = (1+\gamma)\sum_{i=1}^{r-1} p_{i[i]} + p_j \left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a r^b - d_j, \tag{15}$$

$$L_k(S_1) = (1+\gamma)\sum_{i=1}^{r-1} p_{i[i]} + (1+\gamma)p_j \left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a r^b + p_k \left(1 + \sum_{i=1}^{r-1} p_{[i]} + p_j\right)^a (r+1)^b - d_k, \tag{16}$$

$$L_k(S_2) = (1+\gamma)\sum_{i=1}^{r-1} p_{i[i]} + p_k \left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a r^b - d_k, \tag{17}$$

and

$$L_j(S_2) = (1+\gamma)\sum_{i=1}^{r-1} p_{i[i]} + (1+\gamma)p_k \left(1 + \sum_{i=1}^{r-1} p_{[i]}\right)^a r^b + p_j \left(1 + \sum_{i=1}^{r-1} p_{[i]} + p_k\right)^a (r+1)^b - d_j. \tag{18}$$

If $d_j \leq d_k$, then we obtain $L_k(S_2) \leq L_j(S_2)$. That is, $L_j(S_2) = \max\{L_k(S_2), L_j(S_2)\}$ if $d_j \leq d_k$. Moreover, if $p_j \leq p_k$ and $d_j \leq d_k$, from Theorem 4.1, we have that $L_j(S_1) \leq L_j(S_2)$ and $L_k(S_1) \leq L_j(S_2)$. Hence, $\max\{L_j(S_1), L_k(S_1)\} \leq \max\{L_k(S_2), L_j(S_2)\}$.

Therefore, we see that the optimal schedule can be obtained by sequencing the jobs in a non-decreasing order of d_j.

4.5. The $1/p_{jr}$, $q_{psd}/T_{\max}$ problem.
The maximum tardiness $T_{\max}$ is defined as $T_{\max} = \max\{0, L_{\max}\}$. Then, the results of Theorem 4.4 can be transferred directly to the problem of $1/p_{jr}$, $q_{psd}/T_{\max}$. Consequently, we have the following theorem.

Theorem 4.5. *For the $1/p_{jr}$, $q_{psd}/T_{\max}$ problem, if jobs have agreeable due-dates, i.e., $p_j \leq p_k$ implies $d_j \leq d_k$ for any jobs J_j and J_k, $j, k = 1, 2, \ldots, n$, an optimal schedule can be obtained by sequencing the jobs in the EDD rule.*

5. Numerical Example.
We demonstrate the results of the paper in the following example:

Example 5.1. *Consider 5 jobs with $p_1 = 3$, $p_2 = 4$, $p_3 = 5$, $p_4 = 6$, $p_5 = 7$, $d_1 = 5$, $d_2 = 6$, $d_3 = 7$, $d_4 = 8$, $d_5 = 9$, $\alpha_1 = 6$, $\alpha_2 = 5$, $\alpha_3 = 4$, $\alpha_4 = 3$, $\alpha_5 = 2$. The deteriorating factor and the learning factor are $a = 1.0$ and $b = -0.3$, respectively. The normalizing constant for p-s-d delivery time is $\gamma = 0.05$. According Theorems 4.1-4.5, we know that the optimal schedule is $[J_1, J_2, J_3, J_4, J_5]$ for the following objective functions: $C_{\max}$, $\sum C_j$, $\sum \alpha_j C_j$, $L_{\max}$ and $T_{\max}$. In addition, by Equation (3), we obtained that the completion times of jobs are $C_{[1]} = 3.0$, $C_{[2]} = 16.146$, $C_{[3]} = 45.565$, $C_{[4]} = 98.464$, and $C_{[5]} = 183.103$. Then, we see that the optimal solution for the example is as follows:*

$$C_{\max} = 183.103,$$
$$\sum C_j = 3 + 16.146 + 45.565 + 98.464 + 183.103 = 346.278,$$
$$\sum \alpha_j C_j = 6*3 + 5*16.146 + 4*45.565 + 3*98.464 + 2*183.103 = 942.588,$$
$$L_{\max} = \max_{j=1,2,\ldots,n} \{C_j - d_j\}$$
$$= \max\{3 - 5, 16.146 - 6, 45.565 - 7, 98.464 - 8, 183.103 - 9\} = 174.103,$$
$$T_{\max} = \max\{0, L_{\max}\} = 174.103.$$

6. Conclusions.
In this paper we considered single-machine scheduling problems with p-s-d delivery times and deterioration and learning effects simultaneously to minimize the makespan, the total completion time, the total weighted completion time, the maximum lateness, and the maximum tardiness problems. The job delivery time was assumed to be proportional to the job's waiting time. We showed that the makespan and the total completion time minimization problems can be optimally solved by polynomial time algorithms. We also proved that the total weighted completion time, the maximum lateness, and the maximum tardiness minimization problems remain polynomially solvable under certain conditions. It is worthwhile for future research to investigate p-s-d delivery times scheduling with different jobs deterioration and/or the effect of learning, in multi-machine setting, or optimizing other performance measures.

REFERENCES

[1] A. Bachman and A. Janiak, Scheduling deteriorating jobs dependent on resources for the makespan minimization, *Proc. of 2000: Selected Papers of the Symposium on Operations Research*, Dresden, pp.29-34, 2000.

[2] D. Biskup, A state-of-the-art review on scheduling with learning effects, *European Journal of Operational Research*, vol.188, pp.315-329, 2008.

[3] D. Biskup and J. Herrmann, Single-machine scheduling against due dates with past-sequence-dependent setup times, *European Journal of Operational Research*, vol.191, pp.587-592, 2008.

[4] J. A. Buzacott and J. R. Callahan, The capacity of the soaking pit-rolling mill complex in steel production, *INFOR*, vol.9, pp.87-95, 1971.

[5] T. C. E. Cheng, W.-C. Lee and C.-C. Wu, Scheduling problems with deteriorating jobs and learning effects including proportional setup times, *Computers & Industrial Engineering*, vol.58, pp.326-331, 2010.

[6] T. C. E. Cheng, W.-C. Lee and C.-C. Wu, Single-machine scheduling with deteriorating jobs and past-sequence-dependent setup times, *Applied Mathematical Modelling*, vol.35, pp.1388-1395, 2011.

[7] S. Gawiejnowicz, *Time-Dependent Scheduling*, Springer-Verlag Inc., Berlin, 2008.

[8] R. L. Graham, E. L. Lawler, J. K. Lenstra and 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.

[9] A. Janiak and R. Rudek, Experience based approach to scheduling problems with the learning effect, *IEEE Trans. on Systems, Man, and Cybernetics – Part A: Systems and Humans*, vol.39, pp.344-357, 2009.

[10] C. Koulamas and G. J. Kyparisis, Single-machine scheduling problems with past-sequence-dependent setup times, *European Journal of Operational Research*, vol.187, pp.1045-1049, 2008.

[11] C. Koulamas and G. J. Kyparisis, Single-machine scheduling problems with past-sequence-dependent delivery times, *International Journal of Production Economics*, vol.126, pp.264-266, 2010.

[12] W.-H. Kuo and D.-L. Yang, Single machine scheduling with past-sequence-dependent setup times and learning effects, *Information Processing Letters*, vol.102, pp.22-26, 2007.

[13] V. Mani, P.-C. Chang and S.-H Chen, A parametric analysis for single machine scheduling with past-sequence-dependent setup times, *International Journal of Innovative Computing, Information and Control*, vol.6, no.3(A), pp.1113-1121, 2010.

[14] C. T. Ng, T. C. E. Cheng, A. Bachman and A. Janiak, Three scheduling problems with deteriorating jobs to minimize the total completion time, *Information Processing Letters*, vol.81, pp.327-333, 2002.

[15] M. Posner, Single-server queues with service time dependent on waiting time, *Operations Research*, vol.21, pp.610-616, 1973.

[16] L. Sun, Single-machine scheduling problems with deteriorating jobs and learning effects, *Computers & Industrial Engineering*, vol.57, pp.843-846, 2009.

[17] J.-B. Wang, Single-machine scheduling with past-sequence-dependent setup times and time-dependent learning effect, *Computers & Industrial Engineering*, vol.55, pp.584-591, 2008.

[18] J.-B. Wang, Y. Jiang and G. Wang, Single-machine scheduling with past-sequence-dependent setup times and effects of deterioration and learning, *The International Journal of Advanced Manufacturing Technology*, vol.41, pp.1221-1226, 2009.

[19] S.-J. Yang, Single-machine scheduling problems simultaneous with deteriorating and learning effects under deteriorating multi-maintenance activities consideration, *Computers & Industrial Engineering*, vol.62, pp.271-275, 2012.

[20] S.-J. Yang, C.-J. Hsu, T.-R. Chang and D.-L. Yang, Single-machine scheduling with past-sequence-dependent delivery times and learning effect, *Journal of the Chinese Institute of Industrial Engineers*, vol.28, pp.247-255, 2011.

[21] Y. Yin, D. Xu and J. Wang, Some single-machine scheduling problems with general effects of learning and deterioration, *Computers and Mathematics with Applications*, vol.61, pp.100-108, 2011.

[22] Y. Yin, M. Liu, J. Hao and M. Zhou, Single machine scheduling with job position-dependent learning and time-dependent deterioration, *IEEE Trans. on Systems, Man, and Cybernetics, Part A: Systems and Humans*, vol.42, pp.192-200, 2012.