# Single Machine Scheduling with Time-Dependent Learning Effect and Non-Linear Past-Sequence-Dependent Setup Times

## Yuling Yeh[1], Chinyao Low[2], Wen-Yi Lin[2]

[1]Department of Marketing and Logistics, Nan Kai University of Technology, Nantou, Chinese Taipei
[2]Institute of Industrial Engineering and Management, National Yunlin University of Science and Technology, Douliou, Chinese Taipei
Email: yip@nkut.edu.tw, lowcy@yuntech.edu.tw

## Abstract

**This paper studies a single machine scheduling problem with time-dependent learning and setup times. Time-dependent learning means that the actual processing time of a job is a function of the sum of the normal processing times of the jobs already scheduled. The setup time of a job is proportional to the length of the already processed jobs, that is, past-sequence-dependent (psd) setup time. We show that the addressed problem remains polynomially solvable for the objectives, *i.e.*, minimization of the total completion time and minimization of the total weighted completion time. We also show that the smallest processing time (SPT) rule provides the optimum sequence for the addressed problem.**

## Keywords

**Scheduling, Time-Dependent Learning, Setup Time, Past-Sequence-Dependent, Total Completion Time**

## 1. Introduction

In classical scheduling problems, it is reasonable and necessary to consider scheduling problems with setup times. In many realistic situations, the setup times are considered either sequence independent or sequence dependent. In the first case, the setup times are usually added to the job processing times while in the second case, the setup time for the job currently being scheduled depends on the previous one or ones already scheduled. Koulamas and Kyparisis [1] first introduced a scheduling problem with past-sequence-dependent setup times, *i.e.*, the setup time is dependent on all already scheduled jobs. They showed that a standard single machine scheduling problem with psd setup times can be solvable in polynomial time when the objectives are the makespan, the total completion time and the total absolute differences in completion times, respectively. They also

extended their results to nonlinear psd setup times.

Recently, there is a growing interest in the literature to study scheduling problems with a learning effect [1]-[9] [10]-[12], and some researches take setup times into the study problem as well, such as Kuo and Yang [13] considered a single machine scheduling with past-sequence-dependent setup times and job-independent learning effect and showed the problem remains polynomially solvable for the objectives of the makespan, the total completion time, the total absolute differences in completion times and the sum of earliness, tardiness and common due-date penalty. Wang [14] considered a single-machine scheduling problem with past-sequence-dependent setup times and time-dependent learning effect. He proved that the problem with minimization of some objectives, such as makespan, the total completion time and the sum of the quadratic job completion time can be solved in polynomial time, respectively. Wang [15] considered a single-machine scheduling problem with exponential time-dependent learning effect and past-sequence-dependent setup times. The author indicated that the smallest processing time (SPT) rule can provide an optimum schedule for some performance measures, such as makespan, the total completion time and the sum of the quadratic job completion time, respectively. Although applying learning concepts into the setup or processing operations have been extensively studied in scheduling literature, however, few of them take both considerations into account simultaneously. In this paper, we study a single machine scheduling problem with a learning effects model that includes the psd setup times and the actual processing time of a job as a function of the sum of the normal processing times of the jobs already scheduled. The optimal sequences are developed for the two objectives, minimization of the total completion time and the total weighted completion time.

## 2. Notations and Problem Description

In this section, addressing single machine scheduling problems, the actual processing time of a job is assumed to be a function of the sum of the normal processing times of the jobs already scheduled and the setup time of a job is proportional to the length of the jobs already processed. Let $p_i$ denote the normal processing time of job $i$. In addition, let $p_{[k]}$ denote the normal processing time of a job if it is scheduled in the $k$th position in a sequence. For the proposed learning effect model, the actual processing time of a job $j$ which is scheduled in the position $r$ in a sequence, $p_{[k]}$, is presented as

$$p_{j[r]} = p_j \left( 1 - \frac{\sum_{i=1}^{r-1} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a = p_j \left( \frac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a, \quad j, r = 1, \cdots, n, \tag{1}$$

where $a\,(a \geq 1)$ is learning effect indexes. Like Koulamas and Kyparisis [1], we assume that the non-linear past-sequence-dependent setup times of job $J_j$ if it is scheduled in position $r$ is given as follows:

$$s_{[r]} = b^{(r-1)} \sum_{j=1}^{r-1} p_{[j]}, \quad r = 2, 3, \cdots, n \quad \text{and} \quad s_{[1]} = 0 \tag{2}$$

where $b\,(0 < b < 1)$ is a normalizing constant.

## 3. The Total Completion Time Criterion

Before you begin to format your paper, first write and save the content as a separate text file. Keep your text and graphic files separate until after the text has been formatted and styled. Do not use hard tabs, and limit use of hard returns to only one return at the end of a paragraph. Do not add any kind of pagination anywhere in the paper. Do not number text heads—the template will do that for you. In this section, we consider a single machine scheduling problem with the objective of minimizing the total completion time. We show that the problem $1 / LE, S_{psd} / \sum_{i=1}^{n} C_i$ can be scheduled optimally by the SPT rule. Before proving Theorem 1, two lemmas are presented as follows.

**Lemma 1.** $(b^r + 1) - ax(1-x)^{a-1} - (1-x)^a > 0$, for $0 \leq x \leq 1, a \geq 1, 0 < b < 1$ and $r = 1, 2, \cdots, n-1$.

**Proof.** Let $f(x) = (b^r + 1) - ax(1-x)^{a-1} - (1-x)^a$. Then we have $f'(x) = a(a-1)x(1-x)^{a-2} \geq 0$ for $0 \leq x \leq 1, a \geq 1$ and $r = 1, 2, \cdots, n-1$.

Hence, $f(x)$ is increasing on $0 \leq x \leq 1$. Since $0 < b < 1$ and $r = 1, 2, \cdots, n-1$, we have $f(x) \geq f(0) = b^r > 0$. Thus, the proof is completed.

**Lemma 2.** $(b^r +1)(\lambda -1)+(1-\lambda x)^a - \lambda(1-x)^a \geq 0$, for $\lambda \geq 1, 0 \leq x \leq 1, a \geq 1, 0 < b < 1$ and $r = 1, 2, \cdots, n-1$.

**Proof.** Let $g(\lambda) = (b^r +1)(\lambda -1)+(1-\lambda x)^a - \lambda(1-x)^a$. Then we have $g'(\lambda) = (b^r +1) - ax(1-\lambda x)^{a-1} - (1-x)^a$ and $g''(\lambda) = a(a-1)x^2(1-\lambda x)^{a-2}$.

Since $\lambda \geq 1, a \geq 1, 0 \leq x \leq 1$ and $r = 1, 2, \cdots, n-1$, then the value of $g''(\lambda)$ is a non-negative number. That is, $g''(\lambda) \geq 0$. It implies that $g'(\lambda)$ is an increasing function. In addition, from Lemma 1, we have $g'(1) = (b^r +1) - ax(1-x)^{a-1} - (1-x)^a > 0$. Therefore, $g'(\lambda) \geq g'(1) > 0$ for $\lambda \geq 1, a \geq 1, 0 \leq x \leq 1$ and $r = 1, 2, \cdots, n-1$. Thus, it implies that $g(\lambda)$ is an increasing function for $\lambda \geq 1, a \geq 1, 0 \leq x \leq 1$ and $r = 1, 2, \cdots, n-1$. Since $g(\lambda) \geq g(1) = 0$, it is implies that $(b^r +1)(\lambda -1)+(1-\lambda x)^a - \lambda(1-x)^a \geq 0$, for $\lambda \geq 1, a \geq 1, 0 \leq x \leq 1$ and $r = 1, 2, \cdots, n-1$. Thus, the proof is completed.

**Theorem 1.** *For the minimization of total completion time on a single machine scheduling problem* $1/LE, S_{psd}/\sum_{i=1}^n C_i$, *there exists an optimal schedule that is obtained by sequencing jobs in non-decreasing order of* $p_i$.

**Proof.** For two adjacent jobs $J_i$ and $J_j$, assuming the processing time $p_i \leq p_j$. Let $S_1 = [A, J_i, J_j, B]$ and $S_2 = [A, J_j, J_i, B]$ be two job schedules. Where the difference between $S_1$ and $S_2$ is a pairwise inter-change of two adjacent jobs $J_i$ and $J_j$. A and B are partial schedules and A or B may be empty. We assume that there are $r-1$ jobs in $S_1$. Thus, jobs $J_i$ and $J_j$ are at the positions $r$th and $(r+1)$th in $S_1$. In other words, $J_j$ and $J_i$ are at the positions $r$th and $(r+1)$th in $S_2$. Let $C_k(S_1)$ and $C_k(S_1)$ denote the completion time of the job $J_k$ in the sequence $S_1$ and $S_2$, respectively. In order to prove the $1/LE, S_{psd}/\sum_{i=1}^n C_i$ the problem is minimized by sequencing the jobs in a SPT order, sufficient to show that (a) $C_j(S_1) \leq C_i(S_2)$ and (b) $C_i(S_1)+C_j(S_1) \leq C_j(S_2)+C_i(S_2)$.

First, the proof of part (a) is given as follows.

$$C_j(S_1) = \sum_{i=1}^{r-1}\left(1+\sum_{k=i}^r b^k\right)p_{j[i]} + (b^r +1)p_i\left(\frac{\sum_{i=r}^n P_{[i]}}{\sum_{i=1}^n p_i}\right)^a + p_j\left(\frac{\sum_{i=r}^n P_{[i]} - p_i}{\sum_{i=1}^n p_i}\right)^a \tag{3}$$

$$C_i(S_2) = \sum_{i=1}^{r-1}\left(1+\sum_{k=i}^r b^k\right)p_{j[i]} + (b^r +1)p_j\left(\frac{\sum_{i=r}^n P_{[i]}}{\sum_{i=1}^n p_i}\right)^a + p_i\left(\frac{\sum_{i=r}^n P_{[i]} - p_j}{\sum_{i=1}^n p_i}\right)^a \tag{4}$$

we have

$$C_i(S_2)-C_j(S_1) = (b^r +1)(p_j - p_i)\left(\frac{\sum_{i=r}^n P_{[i]}}{\sum_{i=1}^n p_i}\right)^a - p_j\left(\frac{\sum_{i=r}^n P_{[i]} - p_i}{\sum_{i=1}^n p_i}\right)^a + p_i\left(\frac{\sum_{i=r}^n P_{[i]} - p_j}{\sum_{i=1}^n p_i}\right)^a. \tag{5}$$

By substituting $t = \frac{\sum_{i=r}^n P_{[i]}}{\sum_{i=1}^n p_i}$, $x = \frac{p_i}{\sum_{i=r}^n P_{[i]}}$ and $\lambda = \frac{p_j}{p_i}$. Then Equation (5) is equivalent to

$$C_i(S_2)-C_j(S_1) = p_i t^a [(b^r +1)(\lambda -1) - \lambda(1-x)^a + (1-\lambda x)^a].$$

From Lemma 2, we have $C_i(S_2)-C_j(S_1) \geq 0$. That is, $C_j(S_1) \leq C_i(S_2)$ if $p_i \leq p_j$.

Note the proof of part (a) also shows that **the makespan is minimized** by the SPT rule. Furthermore, the proof of part (b) is given as follows.

$$C_i(S_1)+C_j(S_1) = \sum_{i=1}^{r-1}\left(1+\sum_{k=i}^{r-1} b^k\right)p_{j[i]} + p_i\left(\frac{\sum_{i=r}^n P_{[i]}}{\sum_{i=1}^n p_i}\right)^a + \sum_{i=1}^{r-1}\left(1+\sum_{k=i}^r b^k\right)p_{j[i]}$$

$$+(b^r +1)p_i\left(\frac{\sum_{i=r}^n P_{[i]}}{\sum_{i=1}^n p_i}\right)^a + p_j\left(\frac{\sum_{i=r}^n P_{[i]} - p_i}{\sum_{i=1}^n p_i}\right)^a \tag{6}$$

and

$$C_j(S_2) + C_i(S_2) = w_j \left[ \sum_{i=1}^{r-1} \left( 1 + \sum_{k=i}^{r-1} b^k \right) p_{j[i]} + p_j \left( \frac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a \right]$$

$$+ w_i \left[ \sum_{i=1}^{r-1} \left( 1 + \sum_{k=i}^{r} b^k \right) p_{j[i]} + (b^r + 1) p_j \left( \frac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a + p_i \left( \frac{\sum_{i=r}^{n} p_{[i]} - p_j}{\sum_{i=1}^{n} p_i} \right)^a \right] \qquad (7)$$

$$+ (b^r + 1) p_j \left( \frac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a + p_i \left( \frac{\sum_{i=r}^{n} p_{[i]} - p_j}{\sum_{i=1}^{n} p_i} \right)^a$$

we have

$$[C_j(S_2) + C_i(S_2)] - [C_i(S_1) + C_j(S_1)]$$

$$= (p_j - p_i) \left( \frac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a + (b^r + 1)(p_j - p_i) \left( \frac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a + p_i \left( \frac{\sum_{i=r}^{n} p_{[i]} - p_j}{\sum_{i=1}^{n} p_i} \right)^a - p_j \left( \frac{\sum_{i=r}^{n} p_{[i]} - p_i}{\sum_{i=1}^{n} p_i} \right)^a$$

Since $p_j - p_i \geq 0$ and $\left( \frac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a > 0$, the first term is non-negative. From (a), the sum of the second to the fourth terms are non-negative as well. Therefore, this implies that $C_j(S_2) + C_i(S_2) \geq C_i(S_1) + C_j(S_1)$.

This completes the proof of (b) and thus of the theorem.

Hence, the optimal job-sequence of the single machine scheduling problem $1/LE, S_{psd} / \sum_{i=1}^{n} C_i$ can be obtained by an algorithm which sequences the jobs in a SPT order. That is, the problem $1/LE, S_{psd} / \sum_{i=1}^{n} C_i$ can be solved in polynomial time.

## 4. The Total Weighted Completion Time Problem

For the problem to minimize the total weighted completion time, we show that an optimal solution if the processing times and the weights are agreeable, *i.e.*, $p_i \leq p_j$ implies $w_i \geq w_j$ for all the jobs $J_i$ and $J_j$. The result is stated in the following theorem. Before proving Theorem 2, two lemmas are introduced as follows.

Lemma 3. $1 - \lambda_1 a x (1 - \alpha x)^{a-1} - \lambda_2 (1 - x)^a \geq 0$ for $a \geq 1, 0 \leq \lambda_2 \leq 1, \alpha \geq 1$ and $0 \leq x \leq 1$.

Proof. Let $f(x) = 1 - \lambda_1 a x (1 - \alpha x)^{a-1} - \lambda_2 (1 - x)^a$. Then we have
$f'(x) = -\lambda_1 a (1 - \alpha x)^{a-1} + \lambda_1 a(a-1)\alpha x (1 - \alpha x)^{a-2} + \lambda_2 a (1 - x)^{a-1} \geq 0$ for $a \geq 1, 0 \leq \lambda_2 \leq 1, \alpha \geq 1$ and $0 \leq x \leq 1$.

Hence, $f(x)$ is increasing on the value of $x$.

Since $f(x) \geq f(0) = 1 - \lambda_2 \geq 0$, for $a \geq 1$, $0 \leq \lambda_1 \leq \lambda_2 \leq 1$, $\alpha \geq 1$ and $0 \leq x \leq 1$. Thus, completes the proof.

Lemma 4. $(\alpha - 1) + \lambda_1 (1 - \alpha x)^a - \lambda_2 \alpha (1 - x)^a \geq 0$, for $0 \leq \lambda_1 \leq \lambda_2 \leq 1, 0 \leq x \leq 1, \alpha \geq 1$ and $\alpha \geq 1$.

Proof. Let $f(\alpha) = (\alpha - 1) + \lambda_1 (1 - \alpha x)^a - \lambda_2 \alpha (1 - x)^a$. Then we have $f'(\alpha) = 1 - \lambda_1 a x (1 - \alpha x)^{a-1} - \lambda_2 (1 - x)^a$, and $f''(\alpha) = \lambda_1 a(a-1) x^2 (1 - \alpha x)^{a-2} \geq 0$. Since $a \geq 1, 0 \leq \lambda_1 \leq \lambda_2 \leq 1, \alpha \geq 1, 0 \leq x \leq 1$, the value of $f''(\alpha)$ is a non-negative number. That is, $f''(\alpha) \geq 0$. It implies that $f'(\alpha)$ is an increasing function. In addition, from Lemma 3, we have $f'(1) = 1 - \lambda_1 a x (1 - x)^{a-1} - \lambda_2 (1 - x)^a \geq 0$. Therefore, $f'(\alpha) \geq f'(1) \geq 0$ for $a \geq 1$, $0 \leq \lambda_1 \leq \lambda_2 \leq 1, \alpha \geq 1$ and $0 \leq x \leq 1$. Hence, $f(\alpha)$ is an increasing function for $a \geq 1$ $0 \leq \lambda_1 \leq \lambda_2 \leq 1, \alpha \geq 1, 0 \leq x \leq 1$. Also, $f(\alpha) \geq f(1) = (\lambda_1 - \lambda_2)(1 - x)^a \geq 0$ for $0 \leq \lambda_1 \leq \lambda_2 \leq 1, 0 \leq x \leq 1$, $a \geq 1$ and $a \geq 1$. Thus, the proof is completed.

Theorem 2. For minimization of the total weighted completion time on a single machine scheduling problem $1/LE, S_{psd} / \sum_{i=1}^{n} w_i C_i$, if the jobs have agreeable weights, *i.e.*, $p_i \leq p_j$ implies $w_i \geq w_j$ for all the jobs $J_i$ and $J_j$, where an optimal schedule is obtained by sequencing jobs in a non-decreasing order of $p_i / w_i$.

Proof. Since $p_i \leq p_j$, is observed from Theorem 1 where $C_i(S_1) \leq C_i(S_2)$. We only need to show that $w_i C_i(S_1) + w_j C_j(S_1) \leq w_j C_j(S_2) + w_i C_i(S_2)$. From Equations (6)-(7), we have

$$w_j C_j(S_2) + w_i C_i(S_2) - w_i C_i(S_1) - w_j C_j(S_1)$$

$$= w_j \left[ \sum_{i=1}^{r-1} \left( 1 + \sum_{k=i}^{r-1} b^k \right) p_{j[i]} + p_j \left( \frac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a \right] + w_i \left[ \sum_{i=1}^{r-1} \left( 1 + \sum_{k=i}^{r} b^k \right) p_{j[i]} + (b^r + 1) p_j \left( \frac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a \right.$$

$$\left. + p_i \left( \frac{\sum_{i=r}^{n} p_{[i]} - p_j}{\sum_{i=1}^{n} p_i} \right)^a \right] - w_i \left[ \sum_{i=1}^{r-1} \left( 1 + \sum_{k=i}^{r-1} b^k \right) p_{j[i]} + p_i \left( \frac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a \right]$$

$$- w_j \left[ \sum_{i=1}^{r-1} \left( 1 + \sum_{k=i}^{r} b^k \right) p_{j[i]} + (b^r + 1) p_i \left( \frac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a + p_j \left( \frac{\sum_{i=r}^{n} p_{[i]} - p_i}{\sum_{i=1}^{n} p_i} \right)^a \right]$$

$$= b^r (w_i - w_j) \sum_{i=1}^{r-1} p_{j[i]} + b^r (w_i p_j - w_j p_i) \left( \frac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a + (w_i + w_j)(p_j - p_i) \left( \frac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a$$

$$+ w_i p_i \left( \frac{\sum_{i=r}^{n} p_{[i]} - p_j}{\sum_{i=1}^{n} p_i} \right)^a - w_j p_j \left( \frac{\sum_{i=r}^{n} p_{[i]} - p_i}{\sum_{i=1}^{n} p_i} \right)^a .$$

By substituting $\lambda_1 = \dfrac{w_i}{w_i + w_j}$, $\lambda_2 = \dfrac{w_j}{w_i + w_j}$, $t = \dfrac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i}$, $x = \dfrac{p_i}{\sum_{i=r}^{n} p_{[i]}}$ and $\alpha = \dfrac{p_j}{p_i}$. Then

$$(w_i + w_j)(p_j - p_i) \left( \frac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a + w_i p_i \left( \frac{\sum_{i=r}^{n} p_{[i]} - p_j}{\sum_{i=1}^{n} p_i} \right)^a - w_j p_j \left( \frac{\sum_{i=r}^{n} p_{[i]} - p_i}{\sum_{i=1}^{n} p_i} \right)^a \quad \text{can be rewritten as}$$

$$(w_i + w_j)(p_j - p_i) \left( \frac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a + w_i p_i \left( \frac{\sum_{i=r}^{n} p_{[i]} - p_j}{\sum_{i=1}^{n} p_i} \right)^a - w_j p_j \left( \frac{\sum_{i=r}^{n} p_{[i]} - p_i}{\sum_{i=1}^{n} p_i} \right)^a \quad \text{(from Lemma 4).}$$

$$= (w_i + w_j) t^a p_i [(\alpha - 1) + \lambda_1 (1 - \alpha x)^a - \lambda_2 \alpha (1 - x)^a] \geq 0$$

From $p_i/w_i < p_j/w_j$, we have $b^r (w_i p_j - w_j p_i) \left( \dfrac{\sum_{i=r}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i} \right)^a \geq 0$. In addition, from $p_i \leq p_j$ which im-

plies $w_i \geq w_j$, we have $b^r (w_i - w_j) \sum_{i=1}^{r-1} p_{j[i]} \geq 0$.

Hence, $w_j C_j(S_2) + w_i C_i(S_2) - w_i C_i(S_1) - w_j C_j(S_1) \geq 0$. That is, $w_j C_j(S_2) + w_i C_i(S_2) \geq w_i C_i(S_1) + w_j C_j(S_1)$ Thus, the proof is completed.

Hence, the optimal job-sequence of the scheduling problem $1/LE, S_{psd}/\sum_{i=1}^{n} w_i C_i$ can be obtained by an algorithm which sequences the jobs in a SPT order. That is, the problem of $1/LE, S_{psd}/\sum_{i=1}^{n} w_i C_i$ can be solved in polynomial time.

## 5. Conclusion

In this study, we analyzed a single machine scheduling problem with time-dependent learning and setup times. Time-dependent learning means that the actual processing time of a job is a function of the sum of the normal processing times of the jobs already scheduled. The setup time of a job is proportional to the length of the already processed jobs, that is, past-sequence-dependent (psd) setup time. The problem addressed with the two objectives, *i.e.*, minimization of the total completion time and total weighted completion time, was studied in depth. We proved that the SPT rule can provide the optimal schedule for both the total completion time and total weighted completion time objectives. We also show that both the total completion time problem remains poly-

nomially solvable, as does the total weighted completion time problem, under certain agreeable conditions.

## References

[1] Koulamas, C. and Kyparisis, G.J. (2008) Single-Machine Scheduling Problems with Past-Sequence-Dependent Setup Times. *European Journal of Operational Research*, **187**, 1045-1049. http://dx.doi.org/10.1016/j.ejor.2006.03.066

[2] Biskup, D. (1999) Single-Machine Scheduling with Learning Considerations. *European Journal of Operations Research*, **115**, 173-178. http://dx.doi.org/10.1016/S0377-2217(98)00246-X

[3] Biskup, D. and Herrmann, J. (2008) Single-Machine Scheduling against Due Dates with Past-Sequence-Dependent Setup Times. *European Journal of Operational Research*, **191**, 587-592. http://dx.doi.org/10.1016/j.ejor.2007.08.028

[4] Koulamas, C. and Kyparisis, G.J. (2007) Single-Machine and Two-Machine Flowshop Scheduling with General Learning Functions. *European Journal of Operational Research*, **178**, 402-407. http://dx.doi.org/10.1016/j.ejor.2006.01.030

[5] Mosheiov, G. (2001) Scheduling Problems with a Learning Effect. *European Journal of Operational Research*, **132**, 687-693. http://dx.doi.org/10.1016/S0377-2217(00)00175-2

[6] Mosheiov, G. and Sidney, J. (2003) Scheduling with General Job-Dependent Learning Curves. *European Journal of Operational Research*, **147**, 665-670. http://dx.doi.org/10.1016/S0377-2217(02)00358-2

[7] Kuo, W.H. and Yang, D.L. (2006) Minimizing the Total Completion Time in a Single-Machine Scheduling Problem with a Time-Dependent Learning Effect. *European Journal of Operational Research*, **174**, 1184-1190. http://dx.doi.org/10.1016/j.ejor.2005.03.020

[8] Kuo, W.H. and Yang, D.L. (2006) Minimizing the Makespan in a Single Machine Scheduling Problem with a Time-Based Learning Effect. *Information Processing Letters*, **27**, 64-67. http://dx.doi.org/10.1016/j.ipl.2005.09.007

[9] Kuo, W.H. and Yang, D.L. (2006) Single-Machine Group Scheduling with a Time-Dependent Learning Effect. *Computers and Operations Research*, **33**, 2099-2112. http://dx.doi.org/10.1016/j.cor.2004.11.024

[10] Lee, W.C. and Wu, C.C. (2004) Minimizing Total Completion Time in a Two-Machine Flowshop with a Learning Effect. *International Journal of Production Economics*, **88**, 85-93. http://dx.doi.org/10.1016/S0925-5273(03)00179-8

[11] Lee, W.C. and Wu, C.C. (2008) Single-Machine Scheduling Problems with a Learning Effect. *Applied Mathematical Modelling*, **32**, 1191-1197. http://dx.doi.org/10.1016/j.apm.2007.03.001

[12] Wang, J.B. (2008) Single-Machine Scheduling with General Learning Functions. *Computers and Mathematics with Applications*, **56**, 1941-1947. http://dx.doi.org/10.1016/j.camwa.2008.04.019

[13] Kuo, W.H. and Yang, D.L. (2007) Single Machine Scheduling with Past-Sequence-Dependent Setup Times and Learning Effects. *Information Processing Letters*, **102**, 22-26. http://dx.doi.org/10.1016/j.ipl.2006.11.002

[14] Wang, J.B. (2008) Single Machine Scheduling with Past-Sequence-Dependent Setup Times and Time-Dependent Learning Effect. *Computers and Industrial Engineering*, **55**, 584-591. http://dx.doi.org/10.1016/j.cie.2008.01.017

[15] Wang, J.B. (2009) Single Machine Scheduling with Exponential Time-Dependent Learning Effect Past-Sequence-Dependent Setup Times. *Computers & Mathematics with Applications*, **57**, 9-16. http://dx.doi.org/10.1016/j.camwa.2008.09.025