Computers and Mathematics with Applications 62 (2011) 2095–2100

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

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

A note on single-machine scheduling with general learning effect and past-sequence-dependent setup time

Wen-Chiung Lee

Department of Statistics, Feng Chia University, Taichung, Taiwan

ARTICLE INFO

Article history:
Received 13 October 2010
Received in revised form 8 June 2011
Accepted 25 June 2011

Keywords:
Scheduling
Learning effect
Past-sequence-dependent setup times
Single-machine

ABSTRACT

In this paper, we study a scheduling model with the consideration of both the learning effect and the setup time. Under the proposed model, the learning effect is a general function of the processing time of jobs already processed and its scheduled position, and the setup time is past-sequence-dependent. We then derive the optimal sequences for two single-machine problems, which are the makespan and the total completion time. Moreover, we showed that the weighted completion time, the maximum lateness, the maximum tardiness, and the total tardiness problems remain polynomially solvable under agreeable conditions.

1. Introduction

In classical scheduling problems, job processing times are assumed to be a constant. Recent studies in many industries showed that unit costs decline as firms produce more of a product and gain knowledge or experience. For instance, Biskup [1] pointed out that repeated processing of similar tasks improves the worker skills; workers are able to perform setup, to deal with machine operations or software, or to handle raw materials and components at a greater pace. This phenomenon is known as "learning effect" in the literature.

Gawiejnowicz [2], Biskup [1], and Cheng and Wang [3] were among the pioneers who brought the concept of learning effect into scheduling problems. Many researchers have devoted lots of efforts on this new area since then. For instance, Mosheiov [4] presented several examples to demonstrate that the optimal schedules for the classical problems are no longer valid when the learning effect is taken into consideration. Lee et al. [5] considered a bi-criteria single-machine scheduling problem to minimize the sum of the total completion time and the maximum tardiness. Koulamas and Kyparisis [6] introduced a sum-of-job-processing-time-based learning effect scheduling model in which employees learn more if they perform a job with longer processing time. They showed that two single-machine problems remain polynomially solvable. In addition, they proved that two two-machine flowshop problems are polynomially solvable under the assumption of ordered or proportional job processing times. Wang [7] and Wang et al. [8] derived the optimal solutions for some single-machine problems when the learning effect is expressed as a function of the sum of processing times of jobs already processed. Recently, Biskup [9] reviewed the scheduling problems with learning effects. Moreover, Janiak and Rudek [10] brought a new learning effect model into the scheduling field where the existing approach is generalized in two ways. First, they relaxed one of the rigorous constraints, and thus each job can provide different experience to the processor in their model. Second, they formulated the job processing time as a non-increasing k-stepwise function that in general is not restricted to a certain learning curve, thereby it can accurately fit every possible shape of a learning function. Janiak and Rudek [11] introduced and analyzed a new learning effect model in which the learning curve is S-shaped. The authors provided the NP-hard proofs of the makespan problem for two cases. Cheng et al. [12], Sun [13] and Zhang and Yan [14] presented

E-mail address: wclee@fcu.edu.tw.

0898-1221/$ – see front matter
doi:10.1016/j.camwa.2011.06.057

models with both the learning and deterioration effects. They then provided the optimal solutions for some scheduling problems. Eren [15] proposed a nonlinear mathematical programming model for a single-machine scheduling problem with unequal release dates and learning effects. Janiak and Rudek [16] brought into scheduling a new approach called multi-abilities learning that generalizes the existing ones and models more precisely real-life settings. On this basis, they focused on the makespan problem with the proposed learning model and provide optimal polynomial time algorithms for its special cases. Lee et al. [17] investigated a single-machine problem with the learning effect and release times where the objective is to minimize the makespan. Huang et al. [18] considered the single-machine scheduling problems with time-dependent deterioration and exponential learning effect. They provided the optimal solutions for some single-machine problems. Cheng et al. [19] introduced a new scheduling model in which job deterioration and learning, and setup times are considered simultaneously. They showed some single-machine problems remain polynomially solvable. Rudek [20] analyzed the makespan problem on two-machine flowshop with learning effects. First, he showed that an optimal solution of this problem does not have to be the 'permutation' schedule if the learning effect is taken into consideration. Furthermore, he proved that the permutation and non-permutation versions of this problem are NP-hard even if the learning effect, in a form of a step learning curve, characterizes only one machine. However, if both machines have learning ability and the learning curves are stepwise, then the permutation version of this problem is strongly NP-hard. Janiak and Rudek [21] pointed out that the learning effects take place in multi-agent optimization. They showed that the minimization of a total transmission cost of packets in a computer network that uses a reinforcement learning routing algorithm can be expressed as the single-machine makespan minimization scheduling problem with the learning effect. On this basis, they proved the problem is at least NP-hard.

In addition, Koulamas and Kyparisis [22] presented the concept of "past-sequence-dependent" (p-s-d) setup times. They provided an example in high-tech manufacturing that the setup time is proportional to the processing times of jobs already processed. In addition, Biskup and Herrmann [23] provided another example of wear-out of equipment in which the sum of the processing times of the prior jobs adds to the processing time of the actual job. In the examples above, the worker skills might improve during the manufacturing process. Recently, several researchers have started to consider both the learning effect and past-sequence-dependent setup times simultaneously. For instance, Wang et al. [24] studied the exponential time-dependent learning effect. Wang et al. [25] considered the Biskup [1] position-based learning effect model and provided the optimal solutions for some single-machine problems. Yin et al. [26] and Wang and Li [27] considered both the position-based and sum-of-processing-time-based learning effect model, and showed some single-machine problems remain polynomially solvable. Moreover, Dutton and Thomas [28] pointed out that the learning rates show considerable variation within industries or firms after a study of more than 200 learning curves. The variation extends not only across firms at a given time, but also within firms over time. Thus, it is worth considering the general learning curve. Motivated by this, we consider a past-sequence-dependent scheduling model where the actual job processing time is expressed as a general function of the normal processing time of jobs already processed and its scheduled position at the same time.

2. Some single-machine problems

There are n jobs ready to be processed on a single machine. For each job j, there is a normal processing time p_j, a weight w_j and a due date d_j. Due to the learning effect, the actual processing time of job j is

$$p^A_{[r]} = p_j f\left(\sum_{k=1}^{r-1} \alpha_k p_{[k]}, r\right) \tag{1}$$

for $r = 1, 2, \ldots, n$, if it is scheduled in the rth position in a sequence where $p_{[k]}$ denote the processing time of the job scheduled in the kth position in a sequence and $\alpha_1 \le \alpha_2 \le \cdots \le \alpha_n$. It is assumed that $f : (0, +\infty) \times [1, +\infty) \to (0, 1]$ is a differentiable non-increasing function with respect to both variables x and y, $f_x(x, y_0) = \frac{\partial}{\partial x} f(x, y_0)$ is non-decreasing with respect to x for every fixed y_0 and $f(0, 1) = 1$. In addition, as in [22], the p-s-d setup time of job j if it is scheduled in the rth position of a sequence is

$$s_{j[1]} = 0 \quad \text{and} \quad s^A_{j[r]} = b \sum_{l=1}^{r-1} p^A_{[l]}, \tag{2}$$

where b is a normalizing constant number with $0 < b < 1$ and $p^A_{[k]}$ denote the actual job processing time scheduled in the kth position in a sequence. It can be seen that it is the Wang [29] model if $p^A_{[r]} = p_j f\left(\sum_{k=1}^{r-1} \alpha_k p_{[k]}, r\right) = p_j \left(1 + \sum_{k=1}^{r-1} p_{[k]}\right)^a$, the Wang et al. [24] model if $p^A_{[r]} = p_j f\left(\sum_{k=1}^{r-1} \alpha_k p_{[k]}, r\right) = p_j \left(A a^{\sum_{k=1}^{r-1} p_{[k]}} + B\right)$, the Wang et al. [25] model if $p^A_{[r]} = p_j f\left(\sum_{k=1}^{r-1} \alpha_k p_{[k]}, r\right) = p_j r^a$. Throughout the paper, let C_j, $L_j = C_j - d_j$ and $T_j = \max\{0, C_j - d_j\}$ denote the completion time, the lateness and the tardiness of job j.

Before presenting the main results, we first state some lemmas that will be used in the proofs in the sequel.

Lemma 1. *$(1-\theta)(1+c)f(x_1, y_1) + \theta f(x_1 + \lambda t, y_2) - f(x_1 + \lambda\theta t, y_2) \le 0$ for $\theta \ge 1, \lambda \ge 0, c > 0, t \ge 0, x_1 > 0$, and $0 < y_1 \le y_2$.*

Proof. Let $F(t) = (1-\theta)(1+c)f(x_1, y_1) + \theta f(x_1 + \lambda t, y_2) - f(x_1 + \lambda\theta t, y_2)$. Taking the first derivative of $F(t)$ with respect to t, we have

$$F'(t) = \lambda\theta\frac{\partial}{\partial x}f(x_1 + \lambda t, y_2) - \lambda\theta\frac{\partial}{\partial x}f(x_1 + \lambda\theta t, y_2) \le 0$$

since $f_x(x, y_0)$ is a non-decreasing function of x and $\theta \ge 1$. It implies that $F(t)$ is a non-increasing function. Thus,

$$F(t) \le F(0) = (1-\theta)[(1+c)f(x_1, y_1) - f(x_1, y_2)] \le 0.$$

This completes the proof. □

Lemma 2. *$f(x_1, y_1)(1 + c\delta_2) + \delta_2\lambda t\frac{\partial}{\partial x}f(x_1 + \lambda\theta t, y_2) - \delta_1 f(x_1 + \lambda t, y_2) \ge 0$ for $x_1 > 0, 0 < y_1 \le y_2, t > 0, \theta \ge 1, \lambda \ge 0, c \ge 0$ and $0 < \delta_1 < \delta_2 < 1$.*

Proof. Let $F(\theta) = f(x_1, y_1) + c\delta_2\lambda + \delta_2\lambda t\frac{\partial}{\partial x}f(x_1 + \lambda\theta t, y_2) - \delta_1 f(x_1 + \lambda t, y_2)$.

Then, we have

$$F(\theta) \ge \delta_1[f(x_1, y_2) - f(x_1 + \lambda t, y_2)] + \lambda\delta_2 t\frac{\partial}{\partial x}f(x_1 + \lambda t, y_2)$$

since $\delta_1 < 1$, f is nonnegative, and non-increasing with respect to y. By Mean Value Theorem, there exists ξ $(0 < \xi < 1)$ such that

$$\begin{aligned} F(\theta) &\ge \delta_1\left[\frac{\partial}{\partial x}f(x_1 + \lambda\xi t, y_2)\right](-\lambda t) + \lambda\delta_2 t\frac{\partial}{\partial x}f(x_1 + \lambda t, y_2) \\ &\ge \delta_2\lambda t\left[\frac{\partial}{\partial x}f(x_1 + \lambda t, y_2) - \frac{\partial}{\partial x}f(x_1 + \lambda\xi t, y_2)\right] \\ &\ge 0 \end{aligned}$$

since $0 < \delta_1 < \delta_2 < 1, \lambda \ge 0, t > 0$, and $\frac{\partial}{\partial x}f(x, y_0)$ is non-decreasing with respect to x for every fixed y_0. This completes the proof. □

Lemma 3. *$\delta_2[f(x_1+\lambda\theta t, y_2)+cu/t]-\delta_1[\theta f(x_1+\lambda t, y_2)+cu/t]+(\delta_2 c\theta-\delta_1 c+\theta-1)f(x_1, y_1) \ge 0$ for $x_1 > 0, \theta \ge 1, u \ge 0, 0 \le y_1 \le y_2, \lambda \ge 0, t > 0, c \ge 0$ and $0 < \delta_1 < \delta_2 < 1$.*

Proof. Let $G(\theta) = \delta_2[f(x_1 + \lambda\theta t, y_2) + cu/t] - \delta_1[\theta f(x_1 + \lambda t, y_2) + cu/t] + (\delta_2 c\theta - \delta_1 c + \theta - 1)f(x_1, y_1)$. Taking the first derivative of $G(\theta)$ with respect to θ, we have from Lemma 2 that $G'(\theta) \ge 0$. Thus, we have

$$\begin{aligned} G(\theta) &\ge G(1) \\ &= (\delta_2 - \delta_1)[f(x_1 + \lambda t, y_2) + cf(x_1, y_1) + cu/t] \ge 0 \end{aligned}$$

since $0 < \delta_1 < \delta_2 < 1$. This completed the proof. □

We will prove the following properties using the pairwise interchange technique. Suppose that S and S' are two job schedules and the difference between S and S' is a pairwise interchange of two adjacent jobs i and j. That is, $S = (\pi, i, j, \pi')$ and $S' = (\pi, j, i, \pi')$, where π and π' each denote a partial sequence. Furthermore, we assume that there are $r-1$ jobs in π. In addition, let A denote the completion time of the last job in π. Under the proposed model, the completion times of jobs*i* and j in S and S' are

$$C_i(S) = A + b\sum_{k=1}^{r-1}p_{[k]}^A + p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right), \tag{3}$$

$$\begin{aligned} C_j(S) = {} & A + b\sum_{k=1}^{r-1}p_{[k]}^A + p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) + b\left(\sum_{k=1}^{r-1}p_{[k]}^A + p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right)\right) \\ & + p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]} + \alpha_r p_i, r+1\right), \end{aligned} \tag{4}$$

$$C_j(S') = A + b\sum_{k=1}^{r-1}p_{[k]}^A + p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right), \tag{5}$$

and

$$C_i(S') = A + b\sum_{k=1}^{r-1} p_{[k]}^A + p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) + b\left(\sum_{k=1}^{r-1} p_{[k]}^A + p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right)\right) + p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]} + \alpha_r p_j, r+1\right). \quad (6)$$

Property 1. *The optimal schedule is obtained by the shortest processing time (SPT) rule for the* $1|p_{[r]}^A = p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right), s_{psd}|C_{\max}$ *problem.*

Proof. Suppose $p_j \geq p_i$. To show that S dominates S', it suffices to show that $C_j(S) \leq C_i(S')$. Taking the difference between Eqs. (4) and (6), we have

$$C_i(S') - C_j(S) = (p_j - p_i)(1+b) f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) + p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]} + \alpha_r p_j, r+1\right) - p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]} + \alpha_r p_i, r+1\right). \quad (7)$$

Substituting $x_1 = \sum_{k=1}^{r-1}\alpha_k p_{[k]}$, $\theta = p_j/p_i$, $c = b$, $t = p_i$, $\lambda = \alpha_r$, $y_1 = r$ and $y_2 = r+1$ into Eq. (7), we have from Lemma 1 that $C_j(S') \geq C_i(S)$. This completes the proof. □

Property 2. *The optimal schedule is obtained by the SPT rule for the* $1|p_{[r]}^A = p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right), s_{psd}\left|\sum C_i\right.$ *problem.*

Proof. The proof is omitted since it is similar to that of Property 1. □

We will show that the weighted shortest processing time rule provides the optimal solution for the total weighted completion time problem if the processing times and the weights are agreeable, i.e., $p_i \leq p_j$ implies $w_i \geq w_j$ for all jobs i and j.

Property 3. *The optimal schedule is obtained by the weighted shortest processing time rule for the* $1|p_{[r]}^A = p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right), s_{psd}\left|\sum w_i C_i\right.$ *problem if the processing times and the weights are agreeable.*

Proof. Suppose that $p_i \leq p_j$. It implies from Property 1 that $C_j(S) \leq C_i(S')$. Thus, to show that Sdominates S', it suffices to show that $w_i C_i(S) + w_j C_j(S) \leq w_j C_j(S') + w_i C_i(S')$. From Eqs. (3)–(6), we have from $w_i \geq w_j$, $p_i \leq p_j$ and Lemma 3 that

$$\begin{aligned}
&[w_j C_j(S') + w_i C_i(S')] - [w_i C_i(S) + w_j C_j(S)] \\
&= w_i\left[b\sum_{k=1}^{r-1} p_{[k]}^A + bp_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) + p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]} + \alpha_r p_j, r+1\right)\right] \\
&\quad - w_j\left[b\sum_{k=1}^{r-1} p_{[k]}^A + bp_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) + p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]} + \alpha_r p_i, r+1\right)\right] \\
&\quad + f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right)(w_i + w_j)(p_j - p_i).
\end{aligned} \quad (8)$$

Substituting $x_1 = \sum_{k=1}^{r-1}\alpha_k p_{[k]}$, $\theta = p_j/p_i$, $c = b$, $t = p_i$, $\lambda = \alpha_r$, $u = \sum_{k=1}^{r-1} p_{[k]}^A$, $\delta_1 = w_j/(w_i + w_j)$, $\delta_2 = w_i/(w_i + w_j)$, $y_1 = r$, and $y_2 = r+1$ into Eq. (8), we have from Lemma 3 that $w_i C_i(S) + w_j C_j(S) \leq w_j C_j(S') + w_i C_i(S')$. This completes the proof. □

Property 4. *The optimal schedule is obtained by the earliest due date rule for the* $1|p_{[r]}^A = p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right), s_{psd}|L_{\max}$ *problem if the job processing times and the due dates are agreeable, i.e.,* $d_i \leq d_j$ *implies* $p_i \leq p_j$ *for all jobs i and j.*

Property 5. *The optimal schedule is obtained by the earliest due date rule for the* $1|p_{[r]}^A = p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right), s_{psd}|T_{\max}$ *problem if the job processing times and the due dates are agreeable, i.e.,* $d_i \leq d_j$ *implies* $p_i \leq p_j$ *for all jobs i and j.*

Property 6. *The optimal schedule is obtained by the earliest due date rule for the* $1|p_{[r]}^A = p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right), s_{psd}\left|\sum T_i\right.$ *problem if the job processing times and the due dates are agreeable, i.e.,* $d_i \leq d_j$ *implies* $p_i \leq p_j$ *for all jobs i and j.*

Proof. Suppose that $d_i \leq d_j$. It implies $p_i \leq p_j$ since they are agreeable. The total tardiness of jobs in π are the same since they are processed in the same order. By Property 1, the makespan is minimized by the SPT rule, thus, the total tardiness of partial sequence π' in S will not be greater than that of π' in S'. To prove that the total tardiness of S is less than or equal to that of S', it suffices to show that $T_i(S) + T_j(S) \leq T_j(S') + T_i(S')$.

To compare the total tardiness of jobs i and j in S and in S', we divide it into two cases. In the first case that $A+b\sum_{k=1}^{r-1} p_{[k]}^A + p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) \leq d_j$, we have from Eqs. (3)–(6) that the total tardiness of jobs i and j in S and in S' are

$$\begin{aligned} T_i(S) + T_j(S) = {} & \max\left\{A + b\sum_{k=1}^{r-1} p_{[k]}^A + p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) - d_i, 0\right\} + \max\left\{A + 2b\sum_{k=1}^{r-1} p_{[k]}^A\right. \\ & \left. + (1+b)p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) + p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]} + \alpha_r p_i, r+1\right) - d_j, 0\right\} \end{aligned}$$

and

$$T_j(S') + T_i(S') = \max\left\{A + 2b\sum_{k=1}^{r-1} p_{[k]}^A + (1+b)p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) + p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]} + \alpha_r p_j, r+1\right) - d_i, 0\right\}.$$

Suppose that neither $T_i(S)$ nor $T_j(S)$ is zero. It is the most restrictive case since it comprises the case that either one or both $T_i(S)$ and $T_j(S)$ are zero. From Property 1 and $d_i \leq d_j$, we have

$$\begin{aligned} [T_j(S') + T_i(S')] - [T_i(S) + T_j(S)] = {} & (1+b)(p_j - p_i) f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) \\ & + p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]} + \alpha_r p_j, r+1\right) - p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]} + \alpha_r p_i, r+1\right) \\ & + d_j - A - b\sum_{k=1}^{r-1} p_{[k]}^A - p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) \geq 0. \end{aligned}$$

Thus, $[T_j(S')+T_i(S')]-[T_i(S)+T_j(S)] \geq 0$ in the first case. In the second case that $A+b\sum_{k=1}^{r-1} p_{[k]}^A + p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) > d_j$, the total tardiness of jobs i and j in S and in S' are

$$\begin{aligned} T_i(S) + T_j(S) = {} & \max\left\{A + b\sum_{k=1}^{r-1} p_{[k]}^A + p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) - d_i, 0\right\} + \max\left\{A + 2b\sum_{k=1}^{r-1} p_{[k]}^A\right. \\ & \left. + (1+b)p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) + p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]} + \alpha_r p_i, r+1\right) - d_j, 0\right\} \end{aligned}$$

and

$$\begin{aligned} T_j(S') + T_i(S') = {} & 2A + 3b\sum_{k=1}^{r-1} p_{[k]}^A + (2+b)p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) + p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]} + \alpha_r p_j, r+1\right) \\ & + p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]} + \alpha_r p_j, r+1\right) - d_i - d_j. \end{aligned}$$

Suppose that neither $T_i(S)$ nor $T_j(S)$ is zero. From Property 1, $d_i \leq d_j$ and $p_i \leq p_j$, we have

$$\begin{aligned} [T_j(S') + T_i(S')] - [T_i(S) + T_j(S)] = {} & (p_j - p_i)(b+2) f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]}, r\right) \\ & + p_i f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]} + \alpha_r p_j, r+1\right) - p_j f\left(\sum_{k=1}^{r-1}\alpha_k p_{[k]} + \alpha_r p_i, r+1\right) \geq 0. \end{aligned}$$

Thus, $[T_j(S') + T_i(S')] - [T_i(S) + T_j(S)] \geq 0$ in the second case. This completes the proof. □

3. Conclusions

In this paper, we considered a scheduling model with both the learning effect and past-sequence-dependent setup times. The model under study is the extension of several models in the literature. We showed that the SPT rule yields the optimal solution for the single-machine makespan and the total completion time problems. We also showed that WSPT rule yields the optimal solution for the total weighted completion time if the weights and the processing times are agreeable. Moreover, we proved that the EDD rule yields the optimal solution for the maximum lateness, the maximum tardiness and the total tardiness problems if the due dates and the processing times are agreeable.

Acknowledgments

The authors are grateful to the editor and the referees, whose constructive comments have led to a substantial improvement in the presentation of the paper. This work was supported by the NSC of Taiwan, ROC, under NSC 98-2221-E-035-033-MY2.

References

[1] D. Biskup, Single-machine scheduling with learning considerations, European Journal of Operational Research 115 (1999) 173–178.
[2] S. Gawiejnowicz, A note on scheduling on a single processor with speed dependent on a number of executed jobs, Information Processing Letters 56 (1996) 297–300.
[3] T.C.E. Cheng, G. Wang, Single machine scheduling with learning effect considerations, Annals of Operations Research 98 (2000) 273–290.
[4] G. Mosheiov, Scheduling problems with a learning effect, European Journal of Operational Research 132 (2001) 687–693.
[5] W.C. Lee, C.C. Wu, H.J. Sung, A bi-criterion single-machine scheduling problem with learning considerations, Acta Informatica 40 (2004) 303–315.
[6] C. Koulamas, G.J. Kyparisis, Single-machine and two-machine flowshop scheduling with general learning function, European Journal of Operational Research 178 (2007) 402–407.
[7] J.B. Wang, Single-machine scheduling with general learning functions, Computers and Mathematics with Applications 56 (2008) 1941–1947.
[8] J.B. Wang, L.H. Sun, L.Y. Sun, Scheduling jobs with an exponential sum-of-actual-processing-time-based learning effect, Computers and Mathematics with Applications 60 (2010) 2673–2678.
[9] D. Biskup, A state-of-the-art review on scheduling with learning effect, European Journal of Operational Research 188 (2008) 315–329.
[10] A. Janiak, R. Rudek, A new approach to the learning effect: beyond the learning curve restrictions, Computers & Operations Research 35 (2008) 3727–3736.
[11] A. Janiak, R. Rudek, Experience based approach to scheduling problems with the learning effect, IEEE Transactions on Systems, Man and Cybernetics Part A 39 (2009) 344–357.
[12] T.C.E. Cheng, C.C. Wu, W.C. Lee, Some scheduling problems with deteriorating jobs and learning effects, Computers & Industrial Engineering 54 (2008) 972–982.
[13] L. Sun, Single-machine scheduling problems with deteriorating jobs and learning effects, Computers & Industrial Engineering 57 (2009) 843–846.
[14] X. Zhang, G. Yan, Single-machine group scheduling problems with deteriorated and learning effect, Applied Mathematics and Computation 216 (2010) 1259–1266.
[15] T. Eren, Minimizing the total weighted completion time on a single machine scheduling with release dates and a learning effect, Applied Mathematics and Computation 208 (2009) 355–358.
[16] A. Janiak, R. Rudek, A note on a makespan minimization problem with a multi-abilities learning effect, OMEGA: The International Journal of Management Science 38 (2010) 213–217.
[17] W.C. Lee, C.C. Wu, P.H. Hsu, A single-machine learning effect scheduling problem with release times, OMEGA: The International Journal of Management Science 38 (2010) 3–11.
[18] X. Huang, J.B. Wang, L.Y. Wang, W.J. Gao, X.R. Wang, Single machine scheduling with time-dependent deterioration and exponential learning effect, Computers & Industrial Engineering 58 (2010) 58–63.
[19] T.C.E. Cheng, W.C. Lee, C.C. Wu, Scheduling problems with deteriorating jobs and learning effects including proportional setup times, Computers & Industrial Engineering 58 (2010) 326–331.
[20] R. Rudek, Computational complexity and solution algorithms for flowshop scheduling problems with the learning effect, Computers & Industrial Engineering 61 (2011) 20–31.
[21] A. Janiak, R. Rudek, A note on the learning effect in multi-agent optimization, Expert Systems with Applications 38 (2011) 5974–5980.
[22] C. Koulamas, G.J. Kyparisis, Single-machine scheduling problems with past-sequence-dependent setup times, European Journal of Operational Research 187 (2008) 1045–1049.
[23] D. Biskup, J. Herrmann, Single-machine scheduling against due dates with past-sequence-dependent setup times, European Journal of Operational Research 191 (2008) 587–592.
[24] J.B. Wang, D. Wang, L.Y. Wang, L. Lin, N. Yin, W.W. Wang, Single machine scheduling with exponential time-dependent learning effect and past-sequence-dependent setup times, Computers and Mathematics with Applications 57 (2009) 9–16.
[25] X.R. Wang, J.B. Wang, W.J. Gao, X. Huang, Scheduling with past-sequence-dependent setup times and learning effects on a single machine, International Journal of Advanced Manufacturing Technology 48 (2010) 739–746.
[26] Y.Q. Yin, D.H. Xu, J.Y. Wang, Some single-machine scheduling problems with past-sequence-dependent setup times and a general learning effect, International Journal of Advanced Manufacturing Technology 48 (2010) 1123–1132.
[27] J.B. Wang, J.X. Li, Single machine past-sequence-dependent setup times scheduling with general position-dependent and time-dependent learning effects, Applied Mathematical Modelling 35 (2011) 1388–1395.
[28] J.M. Dutton, A. Thomas, Treating progress function as a managerial opportunity, Academy of Management 9 (1984) 235–247.
[29] J.B. Wang, Single-machine scheduling with past-sequence-dependent setup times and time-dependent learning effect, Computers & Industrial Engineering 55 (2008) 584–591.