Some Single-Machine Scheduling Problems with a Mixed Learning Function

Peng-Jen Lai[a], Chin-Chia Wu[b,*] and Wen-Chiung Lee[b]

Abstract
Recently, the topic of scheduling with learning effects has kept growing attention, and a survey is further provided to classify the learning models in scheduling into two types, namely position-based learning and sum-of- processing-times-based learning. However, the actual processing time of a given job fast drops to zero as the number of jobs increases in the first model and when the normal job processing times are large in the second model. Motivated by this observation, this paper extends both models to a more general learning model where the actual job processing time is a function of the sum of the logarithm of the processing times of the jobs already processed and general position-based learning. Under the proposed learning model, this paper shows that some single-machine scheduling problems can be solved in polynomial time.

Keywords**:** scheduling; learning effect; single-machine

1. Introduction

In classical scheduling theory, the job processing times are assumed to be fixed and known from the first job to be processed to the last job to be completed. However, in many realistic situations, the efficiency of the production facility (e.g., a machine or a worker) improves continuously with time. As a result, the production time of a given product is shorter if it is scheduled (and so processed) later. For instance, Biskup [1] pointed out that the repeated processing of similar tasks improves worker skills because workers are able to perform setup, deal with machine operations and software or handle raw materials and components at a faster pace. This phenomenon is called as the *learning effect* in the literature.

Since the pioneering studies of Biskup [1], and Cheng and Wang [2] that introduced Wright's [3] learning curve into scheduling problems, many researchers have paid more attention on this relatively young but vivid area of scheduling research. For example, Biskup [1] considered two single-machine scheduling problems and showed that both problems could be solved in polynomial time. Following Biskup's [1] model, under some assumption that a job's processing time is a decreasing function of its position in a sequence, Mosheiov [4] found that some problems with a learning effect are more complex than those in the traditional problem, such as minimizing the sum of weighted completion times, the maximum lateness, and the number of tardy jobs. Mosheiov and Sidney [5] considered a model in which the learning effects gained from doing some jobs are stronger than those from the other jobs, i.e., the so-called job-dependent learning model. Lee *et al*. [6] studied a bicriterion single-machine scheduling problem, and Lee and Wu [7] investigated the problem of minimizing the total completion time in a two-machine flowshop. Wang [8] studied a model in which the job processing times are functions of their starting times and positions in the sequence. Wu *et al*. [9] took the learning effect into consideration for the single-machine maximum lateness scheduling problem. They proposed a branch-and-bound algorithm, incorporating several dominance properties to derive the optimal solution. In addition, they provided two heuristic algorithms for this problem. The first one is based on the earliest due date (EDD) rule and a pairwise neighborhood search. The second one is based on the simulated annealing (SA) approach. Our computational results show that the SA algorithm is surprisingly accurate for a small to medium number of jobs. Koulamas and Kyparisis [10] addressed a general sum-of-job-processing--times -based learning effect model for scheduling, in which employees learn more if they perform a job with a longer processing time. Wang *et al*. [11] studied some scheduling problems with a time-dependent learning effect. They provided several examples to show that the classical scheduling rules do not yield optimal solutions for the problems to minimize the weighted sum of completion times, maximum lateness and number of tardy jobs. They also analyzed their worst-case error bounds for the classical scheduling rules. Eren and G ü ner [12] considered a two-machine flowshop with position-based learning where the objective is to minimize a weighted sum of the total completion time and makespan. They applied integer programming to solve problems up to 30 jobs, and utilized a heuristic algorithm and a tabu search based heuristic algorithm to handle large-sized problems. Wu and Lee [13] extended position-based and sum-of-job- processing-times-based learning models in which the actual job processing time not only depends on its scheduled position, but also depends on the sum of the processing times of jobs already processed. Cheng *et al*. [14] introduced a new scheduling model with learning effects in which the actual processing time of a job is a function of the total normal processing times of the jobs already processed and of the job's

Manuscript received November 22, 2008. This paper is support by NSC under grant number NSC 97-2221-E-060 -MY2.

[a]Peng-Jen Lai is with National Kaohsiung Normal University, Kaohsiung, Taiwan, ROC. (e-mail: laibird@nknucc.nknu.edu.tw).

[*]Corresponding author: [b]Chin-Chia Wu is with Feng Chia, University Taichung, Taiwan, ROC. (e-mail: cchwu@fcu.edu.tw; phone : 886- 4 - 2456 - 7250 ext 4017).

[b]Wen-Chiung Lee is with Feng Chia University, Taichung, Taiwan, ROC. (e-mail: wclee@fcu.edu.tw).

scheduled position.

In addition, Janiak and Rudek [15] proposed an experience-based learning effect. They proved that the problem to minimize the makespan under the Bachman and Janiak [16] model remains polynomially solvable when the experience-based approach is applied, but the same problem under the more general Cheng and Wang [2] model becomes strongly NP-hard in the presence of the new learning effect. Janiak and Rudek [17] introduced a new model of learning into the scheduling field that relaxes one of the rigid constraints by assuming that each job provides a different experience to the processor. They formulated the shape of the learning curve as a non-increasing *k*-stepwise function. Furthermore, they proved that the makespan problem is polynomially solvable if every job provides the same experience to the processor, and it becomes NP-hard if the experiences are different.

Recently, Biskup [18] provided a comprehensive review of scheduling research with learning considerations. In particular, he classified the learning models into two types, namely *position-based learning* and *sum-of-processing-times-based learning*. He further claimed that position-based learning assumes that learning takes place by processing time independent operations like setting up machines. This seems to be a realistic assumption for the case where the actual processing time of a job is mainly machine-driven and has no (or near to zero) human interference. The sum-of-processing- times-based approach takes into account the experience workers have gained from producing the jobs. This might, e.g., be the case for offset printing, where running the press itself is a highly complicated and error-prone process. In addition, Wu and Lee [19] studied the impact of the learning effect on the problem to minimize the total completion time in an *m*-machine permutation flowshop.

However, the actual processing time of a given job drops to zero precipitously as the number of jobs increases in the first model and when the normal job processing times are large in the second model classified in Biskup [18]. Motivated by this observation, we propose a new learning model where the actual job processing time is a function of the sum of the logarithm of the processing times of the jobs already processed and general position-based learning. The use of the logarithm function can be justified on the grounds that learning, like other human activities, is subject to the law of diminishing return. The remainder of this paper is organized as follows. We present the problem formulation in the next section. In Section 3 we provide polynomial-time solution procedures for some single-machine problems. Finally, we conclude the findings in the last section.

2. Problem statements

In this section, we formally describe the proposed model in the single-machine case. Consider a set of n jobs ready to be processed on a single machine. Each job *j* has a normal processing time p_j and a due date d_j. Due to the phenomenon of learning, the actual processing time of job *j* is

$$p_{j[r]} = p_j(z(r) + \sum_{l=1}^{r-1} \alpha_l \log p_{[r-l]})^a$$

if it is scheduled in the *r*th position in a sequence, where $[i]$ denotes the *i*th position in the sequence, here $a \le 0$ and (1) $z(r)$ is a non-decreasing function of the job position r, $\alpha_1, \alpha_2, \ldots, \alpha_n$ are a sequence of coefficients with $0 \le \alpha_1 \le \alpha_2 \le \ldots \le \alpha_n$ (then $b \ge 1$, where

$u = z(r) + \sum_{l=1}^{r-1} \alpha_{r-1} \log p_{[l]}$,

$v = z(r+1) - z(r) + \sum_{l=1}^{r=1} (\alpha_{l+1} - \alpha_l) \log p_{[r-l]}$, $b = 1 + \frac{v}{u}$) or (2) $z(r)$ is a non-increasing function of the job position r, $\alpha_1, \alpha_2, \ldots, \alpha_n$ are a sequence of coefficients with $\alpha_1 \ge \alpha_2 \ge \ldots \ge \alpha_n \ge 0$

(then $0 \le b \le 1$ **).** Furthermore, 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*, respectively. Under the proposed model, the actual job processing time might have a decreasing learning rate depending on the values of $z(r)$, $\alpha_1, \alpha_2, \ldots, \alpha_n$ and *a*,. Moreover, let $y = \frac{\alpha_1 \log p_i}{u}$ and $K = \log p_i$. Before presenting our results, we have some observations and some lemmas which are given below.

Observation 1. Let $u = z(r) + \sum_{l=1}^{r-1} \alpha_{r-1} p_{[l]}$, $b = 1 + \frac{v}{u}$ and

$v = z(r+1) - z(r) + \sum_{l=1}^{r-1} (\alpha_{l+1} - \alpha_l) p_{[r-l]}$.

(1) $z(r)$ is a non-decreasing function of the job position r, $\alpha_1, \alpha_2, \ldots, \alpha_n$ are a sequence of coefficients with $0 \le \alpha_1 \le \alpha_2 \le \ldots \le \alpha_n$ then $b \ge 1$. $(u \ge 0)$

(2) $z(r)$ is a non-increasing function of the job position r, $\alpha_1, \alpha_2, \ldots, \alpha_n$ are a sequence of coefficients with $\alpha_1 \ge \alpha_2 \ge \ldots \ge \alpha_n \ge 0$, then $0 \le b \le 1$ ($u \le 0$).

Observation 2. If $\alpha_1 \ge \alpha_2 \ge \cdots \ge \alpha_n \ge 0$ and $\min_{z=1,n} z(r) \ge \sigma_3$ large sufficiently and α_1, K small sufficient then we can get $y < 1$ and small.

Observation 3. Suppose that z(r) is a non-increasing function of the job position r, $\alpha_1, \alpha_2, \ldots, \alpha_n$ are a sequence of coefficients with $\alpha_1 \ge \alpha_2 \ge \ldots \ge \alpha_n \ge 0$ (implies $b \le 1$) if $\alpha_1 \log(\max_{1 \le i \le n} p_i) \le \min_{1 \le r \le n-1} (z(r) - z(r+1))$, then $b + y \le 1$.

Lemma 1. Let g be

$g(\lambda) = \lambda[1-(b+y)^a] - [1-(b+(1+\frac{\log\lambda}{\log p_i})y)^a]$. If $b \ge 1$, $a \le -1$ and $K = \log p_i \ge 1$ then $g(\lambda) \ge 0$ for $\lambda \ge 1$.

Lemma 2. Suppose $K = \log p_i > 0$, $y \ge 0$, $0 < b < 1$. Suppose furthermore that $b + y \le 1$. If $a < 0$ and $|a|$ small enough (e.g. $\frac{-a}{K} \le \frac{1-(b+y)}{y}$) then

$$g(\lambda) = \lambda[1-(b+y)^a] - [1-(b+(1+\frac{\log\lambda}{K})y)^a] \ge 0$$ for $\lambda \ge 1$.

Observation 4. If z(r) is a non-increasing function of the job position r, $\alpha_1, \alpha_2, \ldots, \alpha_n$ are a sequence of coefficients with $\alpha_1 \ge \alpha_2 \ge \ldots \ge \alpha_n \ge 0$.(then $0 \le b \le 1$). And suppose

$$|a| \le \frac{1}{\alpha_1}[\min_{r=1,\cdots,n}(z(r)-z(r+1)) + \sum_{l=1}^{r-1}\min_{l=1,\cdots,n}(\alpha_l - \alpha_{l+1})\min_{i=1,\cdots,n}\log p_i]$$

or $|a| \le \min_{i=1,\cdots,n}\log p_i$. Then a satisfies

$$\frac{-a}{K} \le \frac{1-(b+y)}{y}.$$

Lemma 3. If $k \ge 1$, $y \ge 0$, $b \ge 1$, $K = \log p_i \ge 1$ and $a \le 0$. Then

$$g(\lambda) = (\lambda - 1) + \lambda k[1-(b+y)^a] - \frac{1}{k}[1-(b+(1+\frac{\log(\lambda k)}{K})y)^a] \ge 0$$ for $\lambda \ge 1$.

3. Some solvable single-machine problems

In this section we prove the properties for the optimal solutions for some single-machine problems using the pairwise job interchange technique. Let S and S' be two job schedules where the difference between S and S' is a pairwise interchange of two adjacent jobs *i* and *j*, i.e., $S = (\pi, i, j, \pi')$ and $S' = (\pi, j, i, \pi')$, where π and π' each denote a partial sequence. Furthermore, we assume that there are *r*-1 scheduled jobs in π. In addition, let *A* denote the completion time of the last job in π. Under *S*, the completion times of jobs *i* and *j* are respectively

$$C_i(S) = A + p_i(z(r) + \sum_{l=1}^{r-1}\alpha_l \log p_{[r-l]})^a \quad (1)$$

and

$$C_j(S) = A + p_i(z(r) + \sum_{l=1}^{r-1}\alpha_l \log p_{[r-l]})^a.$$
$$+ p_j(z(r+1) + \sum_{l=1}^{r-1}\alpha_{l+1}\log p_{[r-l]} + \alpha_1 \log p_i)^a \quad (2)$$

Similarly, the completion times of jobs *j* and *i* in S' are respectively

$$C_j(S') = A + p_j(z(r) + \sum_{l=1}^{r-1}\alpha_l \log p_{[r-l]})^a \quad (3)$$

and

$$C_i(S') = A + p_i(z(r) + \sum_{l=1}^{r-1}\alpha_l \log p_{[r-l]})^a$$
$$+ p_i(z(r+1) + \sum_{l=1}^{r-1}\alpha_{l+1}\log p_{[r-l]} + \alpha_1 \log p_j)^a \quad (4)$$

Property 1. If $z(r)$ is a non-decreasing function of the job position r, $\alpha_1, \alpha_2, \ldots, \alpha_n$ are a sequence of coefficients with $0 \le \alpha_1 \le \alpha_2 \le \ldots \le \alpha_n$ (then $b \ge 1$), $a \le -1$ and $\min_{i=1,\cdots,n}\log p_i \ge 1$. Then the shortest processing time (SPT) first rule yields an optimal schedule for the

$1/p_{j[r]} = p_j(z(r) + \sum_{l=1}^{r-1}\alpha_l \log p_{[r-l]})^a / C_{\max}$ problem.

Proof: Suppose that $p_i \le p_j$. To show that S dominates S', it suffices to show that $C_j(S) \le C_i(S')$. Taking the difference between Equations (2) and (4), we have

$$C_i(S') - C_j(S) = (p_j - p_i)(z(r) + \sum_{l=1}^{r-1}\alpha_l \log p_{[r-l]})^a + p_i(z(r+1) + \sum_{l=1}^{r-1}\alpha_{l+1}\log p_{[r-l]}$$
$$+ \alpha_1 \log p_j)^a - p_j(z(r+1) + \sum_{l=1}^{r-1}\alpha_{l+1}\log p_{[r-l]} + \alpha_1 \log p_i)^a \quad (5)$$

Substituting

$u = z(r) + \sum_{l=1}^{r-1}\alpha_{r-l}\log p_{[l]}$,

$v = z(r+1) - z(r) + \sum_{l=1}^{r-1}(\alpha_{l+1} - \alpha_l)\log p_{[r-l]}$, $\lambda = \frac{p_j}{p_i}$,

$b = 1 + \frac{v}{u}$ and $y = \frac{\alpha_1 \log p_i}{u}$ into Equation (5), we have

$$C_i(S') - C_j(S) = p_i u^a \{\lambda[1-(b+y)^a]$$
$$-[1-(b+(1+\frac{\log\lambda}{\log p_i})y)^a]\} \quad (6)$$

Since $\lambda = \frac{p_j}{p_i} \ge 1$, $y = \frac{\alpha_1 \log p_i}{u} \ge 0$, $b \ge 1$ and $a \le -1$, we have from Lemma 1 that $C_i(S') - C_j(S) \ge 0$. Therefore, repeating this interchange argument for all the jobs not sequenced in the SPT order completes the proof of the property.

Property 2. If $z(r)$ is a non-decreasing function of the job position r, $\alpha_1, \alpha_2, \ldots, \alpha_n$ are a sequence of coefficients with $0 \le \alpha_1 \le \alpha_2 \le \ldots \le \alpha_n$ (then $b \ge 1$) , $a \le -1$ and $\min_{i=1,\cdots,n}\log p_i \ge 1$. The SPT order yields an optimal schedule for the

$1/p_{j[r]} = p_j(z(r) + \sum_{l=1}^{r-1}\alpha_l p_{[r-l]})^a / \sum C_i$ problem.

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

Property 3. If $z(r)$ is a non-decreasing function of the job position r, $\alpha_1, \alpha_2, \ldots, \alpha_n$ are a sequence of coefficients with $0 \le \alpha_1 \le \alpha_2 \le \ldots \le \alpha_n$ (then $b \ge 1$), $a \le -1$ and $\min_{i=1,\cdots,n} \log p_i \ge 1$. The SPT order yields an optimal schedule for the

$$1/\, p_{j[r]} = p_j(z(r) + \sum_{l=1}^{r-1} \alpha_l p_{[r-l]})^a \,/\, \sum L_i \text{ problem.}$$

Proof: The total lateness $L = \sum_{r=1}^{n}(C_{[r]} - d_{[r]})$ is minimized if $TC = \sum_{r=1}^{n} C_{[r]}$ is minimized since $\sum_{r=1}^{n} d_{[r]}$ is a constant. Thus, it is a straightforward result from Property 2.

Property 4. If $z(r)$ is a non-increasing function of the job position r, $\alpha_1, \alpha_2, \ldots, \alpha_n$ are a sequence of coefficients with $\alpha_1 \ge \alpha_2 \ge \ldots \ge \alpha_n \ge 0$ (then $0 \le b \le 1$), and $\alpha_1 \log(\max_{1 \le i \le n} p_i) \le \min_{1 \le r \le n-1}(z(r) - z(r+1))$ (then $b + y \le 1$), $a \le -1$, $|a|$ small enough (e.g. If

$$|a| \le \frac{1}{\alpha_1}[\min_{r=1,\cdots,n}(z(r) - z(r+1)) + \sum_{l=1}^{r-1} \min_{l=1,\cdots,n}(\alpha_l - \alpha_{l+1}) \min_{i=1,\cdots,n} \log p_i]$$

, or if $|a| \le \min_{i=1,\cdots,n} \log p_i$) and $\min_{i=1,\cdots,n} \log p_i \ge 1$. Then the shortest processing time yields an optimal schedule for the

$$1/\, p_{j[r]} = p_j(z(r) + \sum_{l=1}^{r-1} \alpha_l \log p_{[r-l]})^a \,/\, C_{\max} \text{ problem.}$$

Proof: Suppose that $p_i \le p_j$. To show that S' dominates S, it suffices to show that $C_j(S) \ge C_i(S')$. Taking the difference between Equations (2) and (4), we have

$$\begin{aligned} C_i(S') - C_j(S) = {} & (p_j - p_i)(z(r) + \sum_{l=1}^{r-1} \alpha_l \log p_{[r-l]})^a \\ & + p_i(z(r+1) + \sum_{l=1}^{r-1} \alpha_{l+1} \log p_{[r-l]} + \alpha_1 \log p_j)^a \\ & - p_j(z(r+1) + \sum_{l=1}^{r-1} \alpha_{l+!} \log p_{[r-l]} + \alpha_1 \log p_i)^a \qquad (7) \end{aligned}$$

Substituting $y = \frac{\alpha_1 \log p_i}{u}$, $u = z(r) + \sum_{l=1}^{r-1} \alpha_{r-l} \log p_{[l]}$,

$v = z(r+1) - z(r) + \sum_{l=1}^{r-1}(\alpha_{l+1} - \alpha_l) \log p_{[r-l]}$, $\lambda = \frac{p_j}{p_i}$,

$b = 1 + \frac{v}{u}$ and into Equation (7), we have

$$C_i(S') - C_j(S) = p_i u^a \{\lambda[1-(b+y)^a] \; -[1-(b+(1+\frac{\log\lambda}{\log p_i})y)^a]\} \quad (8)$$

Since $\lambda = \frac{p_j}{p_i} \ge 1$, $y = \frac{\alpha_1 \log p_i}{u} \ge 0$, $b \ge 1$ and $a \le -1$, we have from Lemma 2 that $C_i(S') - C_j(S) \ge 0$. Therefore, repeating this interchange argument for all the jobs not sequenced in the SPT order completes the proof of the property.

Property 5. $z(r)$ is a non-increasing function of the job position r, $\alpha_1, \alpha_2, \ldots, \alpha_n$ are a sequence of coefficients with $\alpha_1 \ge \alpha_2 \ge \ldots \ge \alpha_n \ge 0$ (then $0 \le b \le 1$), and $\alpha_1 \log(\max_{1 \le i \le n} p_i) \le \min_{1 \le r \le n-1}(z(r) - z(r+1))$ (then $b + y \le 1$), $a \le -1$, $|a|$ small enough (e.g. If

$$|a| \le \frac{1}{\alpha_1}[\min_{r=1,\cdots,n}(z(r) - z(r+1)), + \sum_{l=1}^{r-1} \min_{l=1,\cdots,n}(\alpha_l - \alpha_{l+1}) \min_{i=1,\cdots,n} \log p_i]$$

or if $|a| \le \min_{i=1,\cdots,n} \log p_i$) and $\min_{i=1,\cdots,n} \log p_i \ge 1$. The SPT order yields an optimal schedule for the

$$1/\, p_{j[r]} = p_j(z(r) + \sum_{l=1}^{r-1} \alpha_l p_{[r-l]})^a \,/\, \sum C_i \text{ problem.}$$

Proof: The proof is omitted since it is similar to that of Property 4.

Property 6. $z(r)$ is a non-increasing function of the job position r, $\alpha_1, \alpha_2, \ldots, \alpha_n$ are a sequence of coefficients with $\alpha_1 \ge \alpha_2 \ge \ldots \ge \alpha_n \ge 0$ (then $0 \le b \le 1$), and $\alpha_1 \log(\max_{1 \le i \le n} p_i) \le \min_{1 \le r \le n-1}(z(r) - z(r+1))$ (then $b + y \le 1$), $a \le -1$, $|a|$ small enough (e.g. If

$$|a| \le \frac{1}{\alpha_1}[\min_{r=1,\cdots,n}(z(r) - z(r+1)) + \sum_{l=1}^{r-1} \min_{l=1,\cdots,n}(\alpha_l - \alpha_{l+1}) \min_{i=1,\cdots,n} \log p_i],$$

or if $|a| \le \min_{i=1,\cdots,n} \log p_i$) and $\min_{i=1,\cdots,n} \log p_i \ge 1$. The SPT order yields an optimal schedule for the

$$1/\, p_{j[r]} = p_j(z(r) + \sum_{l=1}^{r-1} \alpha_l p_{[r-l]})^a \,/\, \sum L_i \text{ problem.}$$

Proof: The total lateness $L = \sum_{r=1}^{n}(C_{[r]} - d_{[r]})$ is minimized if $TC = \sum_{r=1}^{n} C_{[r]}$ is minimized since $\sum_{r=1}^{n} d_{[r]}$ is a constant. Thus, it is a straightforward result from Property 5.

Next, we show that the WSPT rule provides the optimal solution for the total weighted completion time problem under the proposed deterioration model if if the processing times and the weights are agreeable, i.e., $\frac{p_j}{p_i} \ge \frac{w_j}{w_i} \ge 1$.

Property 7. If $z(r)$ is a non-decreasing function of the job position r, $\alpha_1, \alpha_2, \ldots, \alpha_n$ are a sequence of coefficients with $0 \le \alpha_1 \le \alpha_2 \le \ldots \le \alpha_n$ (then $b \ge 1$), $a \le -1$ and

ISBN: 978-988-17012-7-5

$\min_{i=1,\cdots,n} \log p_i \geq 1$. The SPT rule yields an optimal schedule for the $1/p_{j[r]} = p_j(z(r)+\sum_{l=1}^{r-1}\alpha_l p_{[r-l]})^a / \sum w_i C_i$ problem if the processing times and the weights are agreeable.

Proof: Suppose that $\frac{p_j}{p_i} \geq 1$. Since $p_i \leq p_j$, it implies from Lemma1 that $C_j(S) \leq C_i(S')$. Thus, to show that S dominates 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 Equations (1) to (4), we have

$$\begin{aligned}
&[w_j C_j(S') + w_i C_i(S')] - [w_i C_i(S) + w_j C_j(S)] \\
&= (w_i p_j - w_j p_i)(z(r) + \sum_{l=1}^{r-1}\alpha_l \log p_{[r-1]})^a \\
&+ w_j p_j[(z(r) + \sum_{l=1}^{r-1}\alpha_l \log p_{[r-l]})^a - (z(r+1) \\
&+ \sum_{l=1}^{r-1}\alpha_{l+1}\log p_{[r-l]} + \alpha_1 \log p_i)^a] \\
&- w_i p_i[(z(r) + \sum_{l=1}^{r-1}\alpha_l \log p_{[r-l]})^a - (z(r+1) \\
&+ \sum_{l=1}^{r-1}\alpha_{l+1}\log p_{[r-l]} + \alpha_1 \log p_j)^a \qquad (9)
\end{aligned}$$

Substituting

$$u = z(r) + \sum_{l=1}^{r-1}\alpha_l \log p_{[r-1]}$$

$$v = z(r+1) - z(r) + \sum_{l=1}^{r-1}(\alpha_{l+1} - \alpha_l)\log p_{[r-1]}$$

$y = \frac{\alpha_1 \log p_i}{u}$, $\lambda = \frac{p_j / w_j}{p_i / w_i}$, $b = 1 + \frac{v}{u}$, and $k = \frac{w_j}{w_i}$ into Equation (9), we have

$$\begin{aligned}
&[w_j C_j(S') + w_i C_i(S')] - [w_i C_i(S) + w_j C_j(S)] \\
&= w_j p_i u^a \{(\lambda - 1) + \lambda k[1 - (b+y)^a] \\
&\qquad - \frac{1}{k}[1 - (b + (1 + \frac{\log(\lambda k)}{\log p_i})y)^a]\} \qquad (10)
\end{aligned}$$

Since $\lambda = \frac{p_j / w_j}{p_i / w_i} \geq 1$, $k = \frac{w_j}{w_i} \geq 1$, $kK \geq 1$, $y = \frac{\alpha_1 p_i}{u} \geq 0$, $b \geq 1$ and $a \leq -1$, we have from Lemma 3 that $w_j C_j(S') + w_i C_i(S') \geq w_i C_i(S) + w_j C_j(S)$. Thus, S dominates S'. Repeating this interchange argument for all the jobs not sequenced in the WSPT order completes the proof of Property 7.

4. Conclusions

Scheduling problems with learning consideration have captured many scheduling researchers' attention in recent years. Biskup [1] proposed a position-based learning model while Koulamas and Kyparisis [10] addressed a general sum-of-job-processing- -times -based learning effect model. However, we found in both models that the actual processing time of a given job fast drops to zero as the number of jobs increases in the position-based model or when the normal job processing times are large in the sum-of-processing- times-based model. Motivated by this observation, we proposed a new learning model where the actual job processing time is a function of the sum of the logarithm of the processing times of the jobs already processed. In particular, we showed that the problems to minimize the makespan, total completion time and total lateness in a single machine are polynomially solvable under the proposed learning model. In addition, we showed that the problems to minimize the total weighted completion time is polynomially solvable under some agreeable conditions.

Overall, in the proposed model we overcame some draws in position-based and sum-of-processing- times-based models, but we made some restrictions on normal job processing times. Future studies may focus on relaxing these restrictions or multiple-machine systems.

References

[1] D. Biskup, (1999) Single-machine scheduling with learning considerations. *European Journal of Operational Research* 115, 173–178.

[2] T.C.E. Cheng, G. Wang, (2000) Single machine scheduling with learning effect considerations. *Annals of Operations Research* 98, 273-290.

[3] T.P. Wright, (1936) Factors affecting the cost of airplanes. *Journal of Aeronautical Science* 3, 122-128.

[4] G. Mosheiov, (2001) Scheduling problems with a learning effect, *European Journal of Operational Research* 132, 687-693.

[5] G. Mosheiov, J.B. Sidney, (2003) Scheduling with general job-dependent learning curves. *European Journal of Operational Research* 147, 665-670.

[6] W.C. Lee, C.C. Wu, H.J. Sung, (2004) A bi-criterion single-machine scheduling problem with learning considerations, *Acta Informatica* 40, 303-315.

[7] W.C. Lee, C.C. Wu, (2004) Minimizing total completion time in a two-machine flowshop with a learning effect. *International Journal of Production Economics* 88, 85-93.

[8] J.B. Wang, (2007) Single-machine scheduling problems with the effects of learning and deterioration. *OMEGA-The International Journal of Management Science* 35, 397-402.

[9] C.C. Wu, W.C. Lee, T. Chen, (2007) Heuristic algorithms for solving the maximum lateness scheduling problem with learning considerations. Computers and Industrial Engineering 52, 124-132.

[10] C. Koulamas, G.J. Kyparisis, (2007) Single-machine and two-machine flowshop scheduling with general learning functions. *European Journal of Operational Research* 178, 402-407.

[11] J.B. Wang, C.T. Ng, T.C.E. Cheng, L.L. Liu, (2007) Single-machine scheduling with a time-dependent learning effect. *International Journal of Production Economics* 88, 85–93.

[12] T. Eren, E. Güner, (2008) A bicriterion flowshop scheduling with a learning effect. *Applied Mathematical Modelling* 32, 1719–1733

[13] C.C. Wu, W.C. Lee, (2008) Single-machine scheduling problems with a learning effect, Applied Mathematical Modelling 32, 1191–1197.Cheng

[14] T.C.E. Cheng, C.C. Wu, W.C. Lee, (2008) Some scheduling problems with sum-of-processing-times-based and job-position-based learning effects, Information Sciences 178, 2476–2487.

[15] A. Janiak, R. Rudek, (2007) The learning effect: Getting to the core of the problem. *Information Processing Letters* 103, 183–187.

[16] A. Bachman, A. Janiak, (2004) Scheduling jobs with position dependent processing times. *Journal of the Operational Research Society* 55, 257–264.

[17] A. Janiak, R. Rudek, (2007) A new approach to the learning effect: Beyond the learning curve restrictions. *Computers and Operations Research*, doi: 10.1016/j.cor.2007.04.007.

[18] D. Biskup, (2008) A state-of-the-art review on scheduling with learning effect. *European Journal of Operational Research* 188, 315–329

[19] C.C. Wu, W.C. Lee, (2009) A note on the total completion time problem in a permutation flowshop with a learning effect. *European Journal of Operational Research* 192, 343–347.