

# Scheduling Problems with Integrable Learning and Forgetting Effects

Peng-Jen Lai and Wen-Chiung Lee\*, *Member, IAENG*

**Abstract**-Scheduling problems with learning effect have been widely studied and discussed. However, there are situations that the forgetting effect might exist as well. In this paper, we propose a model with the considerations of both effects. We then derive the optimal schedules for some single-machine scheduling problems.

**Index terms**- forgetting effect, learning effect, makespan, total completion time

## I. Introduction

In many realistic manufacturing situations, the marginal costs decline as firms produce more of a product or the workers gain knowledge or experience, which is known as the learning effect in the literature. For instance, Biskup [1] pointed out that the worker skills improve by processing of similar tasks repeatedly; workers are able to perform setup, to deal with machine operations or software, or to handle raw materials and components at a greater pace. He first brought the idea of learning effect into scheduling problems, and many researchers have devoted to this topic since then. For more scheduling models and problems with learning effects, the readers can refer to the extensive reviews [2-3]. Kuo and Yang [4] studied the learning effect and the deteriorating jobs on a single machine. They provided the optimal solution for the problem to minimize the sum of weighted earliness, tardiness and due-date penalties given that all jobs have a common due date. Lai and Lee [5] proposed a learning effect model in which the actual job processing time is a general function of the normal

time of jobs already processed and its scheduled position. They pointed out that many existing models in literature are the special cases of their proposed model. In addition, they derived the optimal sequences for several single-machine problems. Rudek [6] considered the makespan problem on the flowshop environment. He discussed the computational complexity and provided the solution algorithm for the problem. Wang and Wang [7] provided the worst-case analysis for several flowshop scheduling problems when the learning effect is an exponential function. Yin and Xu [8] considered both the effects of learning and deterioration. They provided the optimal schedules for some single-machine scheduling problems. Zhu et al. [9] studied some single-machine problems with the learning effect and resource allocation in a group technology environment. They provide the optimal solutions for the problems to minimize the weighted sum of makespan and total resource cost, and the weighted sum of total completion time and total resource cost. Kuo et al. [10] tackled two unrelated parallel-machine problems with the consideration of the past-sequence-dependent setup time and learning effects. They showed that the problems to minimize the total absolute deviation of job completion times and the total load on all machines remain polynomially solvable. Zhang et al. [11] derived the optimal scheduling for some single-machine problems with sum-of-logarithm-processing-time-based and position-based learning effects. Bai et al. [12] considered the group technology on a single machine with the effects of learning and deterioration. They derived the optimal schedules for the makespan and the total completion time problems. Furthermore, Bai et al. [13] provided the optimal solutions for some single-machine problems with general exponential learning effects.

Although the scheduling with learning effect has been widely studied, its counterpart, forgetting effect has seldom been discussed in scheduling literature. To the best of our knowledge, Yang and Chand [14] were among the authors

Manuscript received Nov-21-2013. The first author was supported by the NSC of Taiwan, ROC, under NSC-102-2221-E-017-014 and the second author was supported by the NSC of Taiwan, ROC, under NSC-100-2221-E-035-029-MY3.

Peng-Jen Lai is with the Department of Mathematics, National Kaohsiung Normal University, Kaohsiung, Taiwan (email: t1673lai@gmail.com).

Wen-Chiung Lee is with the Department of Statistics, Feng Chia University, Taichung, Taiwan (email: [wlee@fcu.edu.tw](mailto:wlee@fcu.edu.tw)).

who considered both the learning and forgetting effects. They studied the total completion time problem with family setup time on a single machine. Nembhard and Uzumeri [15] pointed out that the industrial organizations are increasingly moving toward shorter product cycle times and shorter runs. Workers in this environment must constantly learn new skill, technology, and processes. Thus, forgetting effects might occur in these situations. Jabera et al. [16] also mentioned that learning and forgetting are mirror images of each other. Globerson et al. [17] and Shtub et al. [18] confirmed the finding that the power-based model is appropriate for capturing the forgetting effects. Lai and Lee [19] mentioned that the forgetting effect might exist when the study of the whole process is complicated and it takes years to learn. Thus, we propose a model with both the learning and forgetting effects in this note. We derive the optimal sequences for some single-machine problems. The remainder of this paper is organized as follows. In section 2, we describe the model. In Section 3, we provide some lemmas and prove that some single-machine problems, such as minimizing the makespan, the total completion time, the total weighted completion time, the maximum lateness, the maximum tardiness, and the total tardiness, are polynomially solvable. The conclusion is given in the final section.

## II. Problem Formulation

There are  $n$  jobs 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 and forgetting effects, the actual processing time of job  $j$ ,  $p_{j[r]}^A$  is

$$p_{j[r]}^A = \begin{cases} p_j(1 - \int_0^r \int_0^r \sum_{k=1}^{r-1} \beta_k p_{[k]} f(x, r) dx dr), & \text{if } r \leq m \\ p_j[1 - \int_0^r \int_0^r \sum_{k=1}^{r-1} \beta_k p_{[k]} f(x, r) dx dr \\ + \int_0^r \int_0^r \sum_{k=1}^{r-1} \beta_k p_{[k]} g(x, r) dx dr], & \text{if } r > m \end{cases} \quad (1)$$

if it is scheduled in the  $r$ th position in a sequence for  $r = 1, 2, \dots, n$ , where  $p_{[k]}$  denote the normal processing

time of the job scheduled in the  $k$ th position in a sequence,  $m$  is the position where the forgetting effect first occurs, and  $\beta_1 \leq \beta_2 \leq \dots \leq \beta_n$  are the weights contributing to the processing time of jobs. It is assumed that  $f : [0, \infty) \times [0, \infty) \rightarrow (0, 1]$  is integrable and non-increasing with respect to  $x$  for every fixed  $y$ , and is non-increasing with respect to  $y$  for every fixed  $x$ , and  $\int_0^\infty \int_0^\infty f(x, r) dx dr < 1$ . It is also assumed that  $g : [0, \infty) \times [1, \infty) \rightarrow (0, 1]$  is integrable and non-decreasing with respect to  $x$ . Furthermore, we assume that

$$f(x, r) \geq g(x, r) \quad (2)$$

for  $x \geq 0, r \geq 0$  and

$$f(x_1, r) - f(x_2, r) \geq g(x_1, r) - g(x_2, r) \quad (3)$$

for  $0 \leq x_1 \leq x_2, 0 \leq r$ . It is seen from Table 1 that many existing models [1, 20-22] are special cases of our proposed model. In this paper, we will use the notation  $C_j, L_j = C_j - d_j$  and  $T_j = \max\{0, C_j - d_j\}$  to denote the completion time, the lateness and the tardiness of job  $j$ .

**Table 1. A list of the special cases (existing models) of our model**

| Model  | Reference                 |
|--|---------------------------|
| $p_{j[r]}^A = p_j r^a, m = n$  | Biskup [1]                |
| $p_{j[r]}^A = p_j(1 + \sum_{l=1}^{r-1} p_{[l]})^a, m = n$                          | Kuo & Yang [20]           |
| $p_{j[r]}^A = p_j(1 - \frac{\sum_{l=1}^{r-1} p_{[l]}}{\sum_{l=1}^n p_l})^a, m = n$ | Koulamas & Kyparisis [21] |
| $p_{j[r]}^A = p_j(\alpha a \sum_{l=1}^{r-1} p_{[l]} + \beta), m = n$               | Wang et al. [22]          |

## III. Single-machine problems

First, we derive some lemmas to be used in the proofs of the properties

**Lemma 1:**  $(\xi - 1)[f(x, y_1) + g(x, y_1)] - \xi[f(x + \lambda t, y_2) + g(x + \lambda t, y_2)] + f(x + \lambda t \xi, y_2) + g(x + \lambda t \xi, y_2) \geq 0$  for  $\xi \geq 1, \lambda \geq 0, t \geq 0$  and  $y_1 \leq y_2$ .

**Proof.** Let  $G(\lambda) = (\xi - 1)[f(x, y_1) + g(x, y_1)] - \xi[f(x + \lambda t, y_2) + g(x + \lambda t, y_2)] + f(x + \lambda t \xi, y_2) + g(x + \lambda t \xi, y_2)$ .

Taking the derivative of  $G(\lambda)$ , we have from Equation (3) that

$$G'(\lambda) = t\xi[f_x(x + \lambda\xi t, y_2) + g_x(x + \lambda\xi t, y_2)] - t\xi[f_x(x + \lambda t, y_2) + g_x(x + \lambda t, y_2)] + t\xi[f_x(x + \lambda\xi t, y_2) - f_x(x + \lambda t, y_2) + g_x(x + \lambda\xi t, y_2) - g_x(x + \lambda t, y_2)] \geq 0$$

In addition, let

$$h(\xi) = (\xi - 1)[f(x, y_1) + g(x, y_1)] - \xi[f(x, y_2) + g(x, y_2)] + f(x, y_2) + g(x, y_2).$$

From Equation (2), we have

$$h'(\xi) = f(x, y_1) + g(x, y_1) - f(x, y_2) - g(x, y_2) \geq 0.$$

With the condition that  $h(1) = 0$ , we have  $h(\xi) \geq 0$  for  $\xi \geq 1$ . Since  $G(0) = h(\xi) \geq 0$  and  $G'(\lambda) \geq 0$ , it follows that  $G(\lambda) \geq 0$ . This completes the proof of Lemma 1.

**Lemma 2**  $(\xi - 1)[f(x, y_1) + g(x, y_1)] + \alpha[f(x + \lambda t\xi, y_2) + g(x + \lambda t\xi, y_2)] - \beta\xi[f(x + \lambda t, y_2) + g(x + \lambda t, y_2)] \geq 0$  for  $\xi \geq 1, \lambda \geq 0, t \geq 0, 0 < \beta \leq \alpha < 1$  and  $y_1 \leq y_2$ .

**Proof:** Let  $H(\xi) = (\xi - 1)[f(x, y_1) + g(x, y_1)] + \alpha[f(x + \lambda t\xi, y_2) + g(x + \lambda t\xi, y_2)] - \beta\xi[f(x + \lambda t, y_2) + g(x + \lambda t, y_2)]$ .

Taking the first derivative of  $H(\xi)$ , we have from Equation (3) and  $\xi \geq 1$  that

$$H'(\xi) = f(x, y_1) + g(x, y_1) + \alpha\lambda t[f_x(x + \lambda t\xi, y_2) + g_x(x + \lambda t\xi, y_2)] - \beta[f(x + \lambda t, y_2) + g(x + \lambda t, y_2)] \geq f(x, y_1) + g(x, y_1) + \alpha\lambda t[f_x(x + \lambda t, y_2) + g_x(x + \lambda t, y_2)] - \beta[f(x + \lambda t, y_2) + g(x + \lambda t, y_2)].$$

Let  $G(\alpha, \beta) = f(x, y_1) + g(x, y_1) + \alpha\lambda t[f_x(x + \lambda t, y_2) + g_x(x + \lambda t, y_2)] - \beta[f(x + \lambda t, y_2) + g(x + \lambda t, y_2)]$ .

It is seen from Equation (3) that  $G(\alpha, \beta)$  is non-increasing with respect to  $\alpha$  and  $\beta$ . Furthermore, let

$$h(t) = f(x, y_1) + g(x, y_1) + \lambda t[f_x(x + \lambda t, y_2) + g_x(x + \lambda t, y_2)] - [f(x + \lambda t, y_2) + g(x + \lambda t, y_2)].$$

Taking the derivative of  $h(t)$ , we have from Equation (3) that

$$h'(t) = \lambda[f_x(x + \lambda t, y_2) + g_x(x + \lambda t, y_2)] + \lambda^2 t[f_{xx}(x + \lambda t, y_2) + g_{xx}(x + \lambda t, y_2)] - \lambda[f_x(x + \lambda t, y_2) + g_x(x + \lambda t, y_2)] = \lambda^2 t[f_{xx}(x + \lambda t, y_2) + g_{xx}(x + \lambda t, y_2)] \geq 0$$

From Equation (2), we have

$$h(0) = f(x, y_1) + g(x, y_1) - [f(x, y_2) + g(x, y_2)] \geq 0.$$

It implies that  $G(1, 1) = h(t) \geq 0$ . Since  $G(\alpha, \beta)$  is non-increasing with respect to  $\alpha$  and  $\beta$ , we have  $G(\alpha, \beta) \geq 0$ . It implies that  $H(\xi)$  is non-decreasing with

respect to  $\xi$ . Therefore,  $H(\xi) \geq H(1) \geq 0$ . This completes the proof of Lemma 2.

**Lemma 3:**  $(\xi - 1)f(x, y_1) + f(x + \lambda t\xi, y_2) + \alpha[f(x + \lambda t, y_2) + g(x + \lambda t, y_2)] + g(x + \lambda t\xi, y_2) \geq 0$  for  $\xi \geq 1, \lambda \geq 0, t \geq 0$  and  $y_1 \leq y_2$ .

**Lemma 4:**  $(\xi - 1)f(x, y_1) + \alpha f(x + \lambda t\xi, y_2) + \alpha g(x + \lambda t\xi, y_2) - \beta\xi[f(x + \lambda t, y_2) + g(x + \lambda t, y_2)] \geq 0$  for  $\xi \geq 1, \lambda \geq 0, t \geq 0, 0 < \beta \leq \alpha < 1$ , and  $y_1 \leq y_2$ .

**Lemma 5:**  $(\xi - 1)f(x, y_1) - \xi f(x + \lambda t, y_2) + f(x + \lambda t\xi, y_2) \geq 0$  for  $\xi \geq 1, \lambda \geq 0, t \geq 0$  and  $y_1 \leq y_2$ .

**Lemma 6:**  $(\xi - 1)f(x, y_1) + \alpha f(x + \lambda t\xi, y_2) - \beta\xi f(x + \lambda t, y_2) \geq 0$  for  $\xi \geq 1, \lambda \geq 0, t \geq 0, y_1 \leq y_2$ , and  $0 < \beta \leq \alpha < 1$ .

The proofs of Lemmas 3-6 are similar to those of Lemmas 1 and 2, and thus omitted.

Next, we will prove the following properties using the pairwise interchange technique. Suppose that  $S_1 = (\pi, j, i, \pi')$  and  $S_2 = (\pi, i, j, \pi')$ , where  $\pi$  and  $\pi'$  each denote a partial sequence. Furthermore, we assume that there are  $r-1$  scheduled jobs in  $\pi$ . In addition, let  $A$  denote the completion time of the last job in  $\pi$ . The completion times of jobs  $j$  and  $i$  in  $S_1$  are

$$C_j(S_1) = \begin{cases} A + p_j f(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r) & \text{if } r \leq m \\ A + p_j f(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r) + p_j g(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r) & \text{if } r > m \end{cases} \quad (4)$$

and

$$C_i(S_1) = \begin{cases} A + p_j f(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r) + p_i f(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_j, r+1) & \text{if } r < m \\ A + p_j f(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r) + p_i [f(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_j, r+1) + g(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_j, r+1)] & \text{if } r = m \\ A + p_j [f(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r) + g(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r)] + p_i [f(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_j, r+1) + g(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_j, r+1)] & \text{if } r > m \end{cases} \quad (5)$$

Similarly, the completion times of jobs  $i$  and  $j$  in  $S_2$  are

$$C_i(S_2) = \begin{cases} A + p_i f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r\right) & \text{if } r \leq m \\ A + p_i \left[ f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r\right) + g\left(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r\right) \right] & \text{if } r > m \end{cases} \quad (6)$$

and

$$C_j(S_2) = \begin{cases} A + p_j f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r\right) + p_j f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_i, r+1\right) & \text{if } r < m \\ A + p_i f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r\right) + p_j \left[ f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_i, r+1\right) + g\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_i, r+1\right) \right] & \text{if } r = m \\ A + p_i \left[ f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r\right) + g\left(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r\right) \right] + p_j \left[ f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_i, r+1\right) + g\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_i, r+1\right) \right] & \text{if } r > m \end{cases} \quad (7)$$

**Property 1.** The optimal schedule for the makespan problem is obtained by the shortest processing time (SPT) rule.

**Proof.** We prove the property by contradiction. Consider an optimal schedule  $S_1$  where the jobs do not follow the SPT rule. In this schedule, there are at least two adjacent jobs, say job  $j$  followed by job  $i$ , such that  $p_j > p_i$ . We now perform an adjacent pairwise interchange of jobs  $i$  and  $j$ , leaving all the other jobs in their original positions, to derive a new sequence  $S_2$ . Taking the difference between Equations (5) and (7), we have

$$C_i(S_1) - C_j(S_2) = \begin{cases} (p_j - p_i) f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r\right) + p_i f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_j, r+1\right) - p_j f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_i, r+1\right) & \text{if } r < m, \end{cases}$$

$$\begin{cases} (p_j - p_i) f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r\right) + p_i \left[ f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_j, r+1\right) + g\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_j, r+1\right) \right] - p_j \left[ f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_i, r+1\right) + g\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_i, r+1\right) \right] & \text{if } r = m, \\ (p_j - p_i) \left[ f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r\right) + g\left(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r\right) \right] + p_i \left[ f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_j, r+1\right) + g\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_j, r+1\right) \right] - p_j \left[ f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_i, r+1\right) + g\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_i, r+1\right) \right] & \text{if } r > m \end{cases} \quad (8)$$

Substituting  $x = \sum_{k=1}^{r-1} \beta_k p_{[k]}$ ,  $\xi = p_j / p_i$ ,  $t = p_i$ ,  $\lambda = \beta_r$ ,

$y_1 = r$  and  $y_2 = r + 1$  into Equation (8), we have from

Lemmas 1, 3 and 5 that  $C_i(S_1) \geq C_j(S_2)$ . This contradicts

the optimality of  $S_1$  and proves that the jobs must be scheduled according to SPT rule in the optimal sequence.

**Property 2.** The optimal schedule for the total completion time problem is obtained by the SPT rule.

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

We show in the next property that the WSPT rule provides an 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 for the total weighted completion time problem is obtained by the WSPT rule if the processing times and the weights are agreeable.

**Proof.** We prove the property by contradiction. Consider an optimal schedule  $S_1$  where the jobs do not follow the WSPT rule. In this schedule, there are at least two adjacent jobs, say job  $j$  followed by job  $i$ , such that  $p_j > p_i$ . We

now perform an adjacent pairwise interchange of jobs  $i$  and  $j$ , leaving all the other jobs in their original positions, to derive a new sequence  $S_2$ . From Equations (4)-(7), we have that  $[w_j C_j(S_1) + w_i C_i(S_1)] - [w_i C_i(S_2) + w_j C_j(S_2)]$

$$= \begin{cases} \left. \begin{aligned} & f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r\right)(w_i + w_j)(p_j - p_i) + w_i p_i f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_r, r+1\right) \\ & - w_j p_j f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_r, r+1\right) \end{aligned} \right\} \text{if } r < m \\ \left. \begin{aligned} & f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r\right)(w_i + w_j)(p_j - p_i) \\ & + w_i p_i \left[ f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_r, r+1\right) + g\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_r, r+1\right) \right] \\ & - w_j p_j \left[ f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_r, r+1\right) + g\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_r, r+1\right) \right] \end{aligned} \right\} \text{if } r = m \quad (9) \\ \left. \begin{aligned} & \left[ f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r\right) + g\left(\sum_{k=1}^{r-1} \beta_k p_{[k]}, r\right) \right] (w_i + w_j)(p_j - p_i) \\ & + w_i p_i \left[ f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_r, r+1\right) + g\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_r, r+1\right) \right] \\ & - w_j p_j \left[ f\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_r, r+1\right) + g\left(\sum_{k=1}^{r-1} \beta_k p_{[k]} + \beta_r p_r, r+1\right) \right] \end{aligned} \right\} \text{if } r > m \end{cases}$$

Substituting  $x = \sum_{k=1}^{r-1} \beta_k p_{[k]}$ ,  $\xi = p_j / p_i$ ,  $t = p_i$ ,  $\lambda = \beta_r$ ,

$\beta = w_j / (w_i + w_j)$ ,  $\alpha = w_i / (w_i + w_j)$ ,  $y_1 = r$ , and  $y_2 = r + 1$  into Equation (9), we have from Lemmas 2, 4 and 6 that  $w_j C_j(S_1) + w_i C_i(S_1) \geq w_i C_i(S_2) + w_j C_j(S_2)$ .

This contradicts the optimality of  $S_1$  and proves that the jobs must be scheduled according to WSPT rule in the optimal sequence.

**Property 4.** The optimal schedule for the maximum lateness problem is obtained by the EDD rule if the processing times and the due dates are agreeable.

**Property 5.** The optimal schedule for the maximum tardiness problem is obtained by the EDD rule if the processing times and the due dates are agreeable.

**Property 6.** The optimal schedule for the total tardiness problem is obtained by the EDD rule if the processing times and the due dates are agreeable.

#### IV. Conclusions

In this paper, we consider a learning-and-forgetting-effect model. We first show many existing models in the literature are special cases of our proposed model. We then derive the

optimal schedules for some single-machine problems, which includes the makespan, the total completion time, the total weighted completion time, the maximum lateness, the maximum tardiness. Considering other models with learning and forgetting effects might be interesting.

#### References

- [1] D. Biskup, Single-machine scheduling with learning considerations, *European Journal of Operational Research* 115 (1999) 173-178.
- [2] D. Biskup, A state-of-the-art review on scheduling with learning effect, *European Journal of Operational Research* 188 (2008) 315-329.
- [3] 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.
- [4] W.H. Kuo, D.L. Yang, A note on due-date assignment and single-machine scheduling with deteriorating jobs and learning effects, *Journal of the Operational Research Society* 62 (2011) 206-210.
- [5] P.J. Lai, W.C. Lee, Single-machine scheduling with general sum-of-processing-time-based and position-based learning effects, *OMEGA-The International Journal of Management Science* 39 (2011) 467-471.
- [6] R. Rudek, Computational complexity and solution algorithms for flowshop scheduling problems with the learning effect, *Computers & Industrial Engineering* 61 (2011) 20-31.
- [7] J.B. Wang, M.Z. Wang, Worst-case analysis for flow shop scheduling problems with an exponential learning effect, *Journal of the Operational Research Society* 63 (2012) 130-137.
- [8] Y.Q. Yin, D.H. Xu, Some single-machine scheduling problems with general effects of learning and deterioration, *Computers and Mathematics with Applications* 61 (2011) 100-108.
- [9] Z.G. Zhu, L.Y. Sun, F. Chu, M. Liu, Single-machine group scheduling with resource allocation and learning effect, *Computers & Industrial Engineering* 60 (2011) 148-157.
- [10] W.H. Kuo, C.J. Hsu, D.L. Yang, Some unrelated parallel machine scheduling problems with past-sequence-dependent setup time and learning effects, *Computers & Industrial Engineering* 61 (2011) 179-183.
- [11] X.G. Zhang, G.L. Yan, W.Z. Huang, G.C. Tang, A note on machine scheduling with sum-of-logarithm-processing-time-based and position-based learning effects, *Information Sciences* 187, (2012) 298-304.
- [12] J. Bai, Z.R. Li, X. Huang, Single-machine group scheduling with general deterioration and learning effects, *Applied Mathematical Modelling* 36 (2012) 1267-1274.
- [13] J. Bai, M.Z. Wang, J.B. Wang, Single machine scheduling with a

- general exponential learning effect, *Applied Mathematical Modelling* 36 (2012) 829–835.
- [14] W.H. Yang, S. Chand, Learning and forgetting effects on a group scheduling problem, *European Journal of Operational Research* 187 (2008) 1033–1044.
- [15] D.A. Nembhard, M.V. Uzumeri, Experiential learning and forgetting for manual and cognitive tasks, *International Journal of Industrial Ergonomics* 25 (2000) 315-326.
- [16] M.Y. Jabera, H.V. Kherb, D.J. Davis, Countering forgetting through training and deployment, *International Journal of Production Economics* 85 (2003) 33–46.
- [17] S. Globerson, N. Levin, A. Shtub, The impact of breaks on forgetting when performing a repetitive task, *IIE Transactions* 21 (1989) 376–381.
- [18] A. Shtub, N. Levin, S. Globerson, Learning and forgetting industrial skills: experimental model, *International Journal of Human Factors in Manufacturing* 3 (1993) 293–305.
- [19] P.J. Lai, W.C. Lee, Single-machine scheduling with learning and forgetting effects, *Applied Mathematical Modelling* 37 (2013) 4509-4516.
- [20] W.H. Kuo, D.L. Yang, Minimizing the total completion time in a single-machine scheduling problem with a time-dependent learning effect, *European Journal of Operational Research* 174 (2006) 1184-1190.
- [21] 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.
- [22] 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.