Int J Adv Manuf Technol (2012) 59:539–545
DOI 10.1007/s00170-011-3528-7

ORIGINAL ARTICLE

A note on single-machine scheduling problems with the effects of deterioration and learning

Li-Yan Wang · En-Min Feng

Received: 14 April 2009 / Accepted: 5 April 2011 / Published online: 11 August 2011

Abstract This paper studies two single-machine scheduling problems with the effect of deterioration and learning. In this model, the processing times of jobs are defined as functions of their starting times and positions in a sequence. For the following two objective functions: the weighted sum of completion times and the maximum lateness, this paper proposes two heuristics according to the corresponding single machine problems without learning effect. This paper also gives the worst-case error bound for the heuristics and provides computational results to evaluate the performance of the heuristics.

Keywords Scheduling · Single machine · Deteriorating jobs · Learning effect

1 Introduction

In classical scheduling problems, the processing times of jobs are assumed to be constant values. However, there are many situations that the processing times of jobs may be subject to change due to deterioration and/or learning phenomena. Machine scheduling problems with deteriorating jobs and/or learning effect have been paid more attention in recent years. Extensive surveys of research related to scheduling deteriorating jobs can be found in Alidaee and Womer [1] and Cheng et al. [2]. An extensive survey of different scheduling models and problems involving jobs with learning effects can be found in Biskup [3]. More recent papers which have considered scheduling jobs with deteriorating jobs and/or learning effect include Wu et al. [4], Shiau et al. [5], and Eren and Guner [6]. Wu et al. [4] considered single-machine total weighted completion time scheduling problem under linear deterioration. They proposed a branch-and-bound method and several heuristic algorithms to solve the problem. Shiau et al. [5] considered two-machine flowshop scheduling to minimize mean flow time with simple linear deterioration. Toksar and Guner [6] considered the bicriteria parallel machine scheduling with a learning effect. They introduced a mixed nonlinear integer programming formulation for the problem. Lee et al. [7] and Wang et al. [8] developed a new deterioration model where the actual job processing time is a function of jobs already processed. Lee et al. [7] showed that the single-machine makespan problem remains polynomially solvable under the proposed model. Wang et al. [8] showed that the total completion time minimization problem for $a \geq 1$ remains polynomially solvable under the proposed model, where a denotes the deterioration rate. For the case of $0 < a < 1$, they showed that an optimal schedule of the total completion time minimization problem is V-shaped with respect to normal job processing times. They also used the classical smallest processing time first rule as a heuristic algorithm for the case of $0 < a < 1$ and analyze its worst-case bound.

However, to the best of our knowledge, apart from the recent paper of Lee [9], Wang [10, 11], Wang and Cheng [12, 13], Toksar and Guner [14], and Wang

L.-Y. Wang (✉)
School of Science, Shenyang Aerospace University,
Shenyang 110136, China
e-mail: wlyhxh@126.com

L.-Y. Wang · E.-M. Feng
Department of Applied Mathematics,
Dalian University of Technology,
Dalian 116024, China

et al. [15], it has not been investigated the scheduling problems with the effects of deterioration and learning. The phenomena of learning effect and deteriorating jobs occurring simultaneously can be found in many real-life situations. For example, as manufacturing becomes increasingly competitive, in order to provide customers with greater product varieties, organizations are moving toward shorter production runs and frequent product changes. The learning and forgetting that workers undergo in this environment have thus become increasingly important as workers tend to spend more time in rotating among tasks and responsibilities prior to becoming fully proficient. These workers are often interrupted by product and process changes, causing deterioration in performance, which we will refer to, for simplicity, as forgetting. Considering learning and forgetting effects in measuring productivity should be helpful in improving the accuracy of production planning and productivity estimation (Nembhard and Osothsilp [16]).

In this paper, we investigate the implications of these phenomena occurring simultaneously for two sing-machine scheduling problems. Specifically, we generalize the results of Wang et al. [15] to a more general context. The remaining part of this paper is organized as follows: In Section 2, we formulate the model. In Sections 3 and 4, we consider two single-machine scheduling problems. In Section 5, we present computational experiments to evaluate the performance of the heuristic algorithms. The last section is the conclusion.

2 Problem formulation

The focus of this paper is to study the effects of deterioration and learning simultaneously. The learning effect model provided by Pegels' learning curve [21] is combined with the proportional linear deterioration model [13] in which the basic job processing time is proportional to the deteriorating rate to yield our model. The model is described as follows: There are given n independent and non-preemptive jobs available for processing on a single machine. All the jobs will be processed starting at time $t_0 \geq 0$ without overlapping and idle time between them. Associated with each job j $(j = 1, 2, ..., n)$, there is a normal processing time p_j, a due date d_j, and a weight w_j. Let $p_{jr}(t)$ be the processing time of job J_j if it is started at time t and scheduled in position r in a sequence. As in Wang et al. [15], we assume that the actual processing time of job j if scheduled in position r is given by

$$p_{jr}(t) = p_j(\alpha a^{r-1} + \beta)(b + ct), \tag{1}$$

where p_j is the basic (normal) processing time of job J_j and a denotes the learning index with $0 < a \leq 1$, $b \geq 0, c \geq 0$. A schedule is a sequence of the jobs that specifies the processing order of the jobs on the machine. Under a given schedule $\pi = (1, 2, \ldots, n)$, the completion time of job J_j is given by $C_j = C_j(\pi)$. Let $\sum w_jC_j$ and $L_{\max} = \max\{C_j - d_j | j = 1, 2, \ldots, n\}$ represent the total weighted completion time and the maximum lateness of a given permutation. In the remaining part of the paper, the problem considered will be denoted using the three-field notation schema $\alpha|\beta|\gamma$ introduced by Graham et al. [17].

3 The weighted sum of completion times minimization problem

First, we give some lemmas; they are useful for the following theorems:

Lemma 1 *For a given schedule $\pi = [J_1, J_2, \ldots, J_n]$ of $1|p_{jr}(t) = p_j(\alpha a^{r-1} + \beta)(b + ct)|\gamma$, if the first job starts at time $t_0 \geq 0$, then the completion time C_j of job J_j is equal to*

$$C_j = (t_0 + \frac{b}{c}) \prod_{i=1}^{j} (1 + p_i c(\alpha a^{i-1} + \beta)) - \frac{b}{c}. \tag{2}$$

Lemma 2 (Zhao et al. [18]) *For the problem $1|p_j(t) = p_j(b + ct)|\sum w_jC_j$, an optimal schedule can be obtained by sequencing the jobs in non-decreasing order of $\frac{p_j}{w_j(1+p_j)}$ (i.e., the weighted shortest processing time first (WSPT) rule).*

From Lemma 2, we can use WSPT rule as a heuristic algorithm for the general problem $1|p_{jr}(t) = p_j(\alpha a^{r-1} + \beta)(b + ct)|\sum w_jC_j$.

Theorem 1 *Let S^* be an optimal schedule and S be a WSPT schedule for the problem $1|p_{jr}(t) = p_j(\alpha a^{r-1} + \beta)(b + ct)|\sum w_jC_j$. Then $\rho_1 = \sum w_jC_j(S)/\sum w_jC_j(S^*) \leq 1/(\alpha a^{n-1} + \beta)$, and the bound is tight.*

Proof Without loss of generality, we can suppose that $\frac{p_1}{w_1(1+p_1)} \leq \frac{p_2}{w_2(1+p_2)} \leq \ldots \leq \frac{p_n}{w_n(1+p_n)}$. Then we have

$$\sum w_jC_j(S)$$
$$= \sum_{j=1}^{n} w_j \left[\left(t_0 + \frac{b}{c} \right) \prod_{i=1}^{j} (1 + p_i c(\alpha a^{i-1} + \beta)) - \frac{b}{c} \right]$$

$$\leq \sum_{j=1}^{n} w_j \left[\left(t_0 + \frac{b}{c}\right) \prod_{i=1}^{j} (1 + p_i c\,(\alpha + \beta)) - \frac{b}{c} \right]$$
$$= \sum_{j=1}^{n} w_j \left[\left(t_0 + \frac{b}{c}\right) \prod_{i=1}^{j} (1 + p_i c) - \frac{b}{c} \right],$$

$$\sum w_j C_j(S^*)$$
$$= \sum_{j=1}^{n} w_j \left[\left(t_0 + \frac{b}{c}\right) \prod_{i=1}^{j} \left(1 + p_i c \left(\alpha a^{i-1} + \beta\right)\right) - \frac{b}{c} \right]$$
$$\geq \sum_{j=1}^{n} w_j \left[\left(t_0 + \frac{b}{c}\right) \prod_{l=1}^{j} \left(1 + c p_{[i]} \left(\alpha a^{n-1} + \beta\right)\right) - \frac{b}{c} \right]$$
$$= \left(\alpha a^{n-1} + \beta\right) \sum_{j=1}^{n} w_j$$
$$\times \left[\left(t_0 + \frac{b}{c}\right) \prod_{i=1}^{j} \left(\frac{1}{\alpha a^{n-1} + \beta} + c\alpha_{[i]} \right) - \frac{b}{c(\alpha a^{n-1} + \beta)} \right]$$
$$\geq \left(\alpha a^{n-1} + \beta\right) \sum_{j=1}^{n} w_j \left[\left(t_0 + \frac{b}{c}\right) \prod_{i=1}^{j} (1 + c p_{[i]}) - \frac{b}{c} \right]$$
$$\geq \left(\alpha a^{n-1} + \beta\right) \sum_{j=1}^{n} w_j \left[\left(t_0 + \frac{b}{c}\right) \prod_{i=1}^{j} (1 + c p_i) - \frac{b}{c} \right]$$

hence

$$\rho_1 = \sum w_j C_j(S) / \sum w_j C_j(S^*) \leq 1/(\alpha a^{n-1} + \beta).$$

It is not difficult to see that the bound is tight, since if $a = 1$, we have $\frac{\sum w_j C_j(S)}{\sum w_j C_j(S^*)} = 1$. This result is intuitive because when $a = 1$, the WSPT schedule is optimal. □

Obviously, $\rho_1 = \sum w_j C_j(S) / \sum w_j C_j(S^*)$ depends on the parameter values.

4 The maximum lateness minimization problem

Lemma 3 (Zhao et al. [18]) *For the problem $1|p_j(t) = p_j(b + ct)|L_{\max}$, an optimal schedule can be obtained by sequencing the jobs in non-decreasing order of d_j (i.e., the smallest due date (EDD) rule).*

Lemma 4 (Wang et al. [15]) *For the problem $1|p_{jr}(t) = p_j(\alpha a^{r-1} + \beta)(b + ct)|C_{\max}$, an optimal schedule can be obtained by sequencing the jobs in non-decreasing order of p_j (i.e., the shortest processing time first (SPT) rule).*

In order to solve the problem approximately, from Lemma 3, we can use the EDD rule as a heuristic for the problem $1|p_{jr}(t) = p_j(\alpha a^{r-1} + \beta)(b + ct)|L_{\max}$. To develop a worst-case performance ratio for the heuristic, we have to avoid cases involving nonpositive $L_{\max}$. Similar to Cheng and Wang [19], the worst-case error bound is defined as follows:

$$\rho_2 = \frac{L_{\max}(S) + d_{\max}}{L_{\max}(S^*) + d_{\max}},$$

where S and $L_{\max}(S)$ denote the heuristic schedule and the corresponding maximum lateness, respectively, while S^* and $L_{\max}(S^*)$ denote the optimal schedule and the minimum maximum lateness value, respectively, and $d_{\max} = \max\{d_j | j = 1, 2, \ldots, n\}$.

Theorem 2 *Let S^* be an optimal schedule and S be an EDD schedule for the problem $1|p_{jr}(t) = p_j(\alpha a^{r-1} + \beta)(b + ct)|L_{\max}$. Then*

$$\rho_2 = \frac{L_{\max}(S) + d_{\max}}{L_{\max}(S^*) + d_{\max}} \leq \frac{\left(t_0 + \frac{b}{c}\right) \prod_{i=1}^{n} (1 + c p_i) - \frac{b}{c}}{C^*_{\max}},$$

and the bound is tight, where $C^*_{\max}$ is the optimal makespan of the problem $1|p_{jr}(t) = p_j(\alpha a^{r-1} + \beta)(b + ct)|C_{\max}$.

Proof Without loss of generality, supposing that $d_1 \leq d_2 \leq \ldots \leq d_n$, we have

$$L_{\max}(S) = \max \left\{ \left(t_0 + \frac{b}{c}\right) \prod_{i=1}^{j} (1 + c p_i (\alpha a^{i-1} + \beta)) - \frac{b}{c} - d_j | j = 1, 2, \ldots, n \right\}$$
$$\leq \max \left\{ \left(t_0 + \frac{b}{c}\right) \prod_{i=1}^{j} (1 + c p_i) - \frac{b}{c} - d_j | j = 1, 2, \ldots, n \right\} = L'_{\max}(S),$$

where $L'_{\max}(S)$ is the optimal value of the problem $1|p_j(t) = p_j(b + ct)|L_{\max}$.

$$L_{\max}(S^*) = \max \left\{ \left(t_0 + \frac{b}{c}\right) \prod_{i=1}^{j} (1 + p_i c (\alpha a^{i-1} + \beta)) - \frac{b}{c} - d_{[j]} | j = 1, 2, \ldots, n \right\}$$

$$= \max\Bigg\{\left(t_0+\frac{b}{c}\right)\prod_{i=1}^{j}(1+cp_{[i]}) - \frac{b}{c} - d_{[j]} - \left(t_0+\frac{b}{c}\right)\prod_{i=1}^{j}(1+cp_{[i]}) + \frac{b}{c}+\left(t_0+\frac{b}{c}\right)\prod_{i=1}^{j}(1+p_i c(\alpha a^{i-1}+\beta)) - \frac{b}{c} \Big| j=1,2,\ldots,n\Bigg\}$$

$$\geq \max\Bigg\{\left(t_0+\frac{b}{c}\right)\prod_{i=1}^{j}(1+cp_{[i]}) - \frac{b}{c} - d_{[j]} \Big| j=1,2,\ldots,n\Bigg\} - (t_0+\frac{b}{c})\prod_{i=1}^{j}(1+cp_{[i]}) + \frac{b}{c}+\left(t_0+\frac{b}{c}\right)\prod_{i=1}^{j}(1+p_i c(\alpha a^{i-1}+\beta)) - \frac{b}{c}$$

$$\geq L'_{\max}(S)-\left(t_0+\frac{b}{c}\right)\prod_{i=1}^{n}(1+cp_i)+\frac{b}{c}+C^*_{\max},$$

hence,

$$L_{\max}(S)-L_{\max}(S^*)\leq\left(t_0+\frac{b}{c}\right)\prod_{i=1}^{n}(1+cp_i)-\frac{b}{c}-C^*_{\max},$$

and so

$$\begin{aligned}\rho_2 &= \frac{L_{\max}(S)+d_{\max}}{L_{\max}(S^*)+d_{\max}}\\ &\leq 1+\frac{\left(t_0+\frac{b}{c}\right)\prod_{i=1}^{n}(1+cp_i)-\frac{b}{c}-C^*_{\max}}{L_{\max}(S^*)+d_{\max}}\\ &\leq 1+\frac{\left(t_0+\frac{b}{c}\right)\prod_{i=1}^{n}(1+cp_i)-\frac{b}{c}-C^*_{\max}}{C^*_{\max}}\\ &\leq \frac{\left(t_0+\frac{b}{c}\right)\prod_{i=1}^{n}(1+cp_l)-\frac{b}{c}}{C^*_{\max}},\end{aligned}$$

where $C^*_{\max}$ can be obtained by the SPT rule (see Lemma 4).

It is not difficult to see that the bound is tight, since if $a=1$, we have $C_{\max}=(t_0+\frac{b}{c})\prod_{l=1}^{n}(1+cp_l)-\frac{b}{c}$ and $\rho_2=\frac{L_{\max}(S)+d_{\max}}{L_{\max}(S^*)+d_{\max}}=1$. This result is intuitive because when $a=1$, the EDD schedule is optimal. □

5 Computational experiments

Computational experiments were conducted to evaluate the effectiveness of the heuristics of WSPT and

Table 1 Computational results of the heuristics for $\tau=0.25$

a	n	ρ_1		$\frac{1}{\alpha a^{n-1}+\beta}$	ρ_2		$\frac{\prod_{l=1}^{n}(1+p_l)-1}{C^*_{\max}}$
		Mean	Max		Mean	Max	Mean
0.15	6	1.0043	1.0284	1.9998	1.0081	1.0248	3.1052
	7	1.0144	1.0587	1.9999	1.0135	1.0425	5.4208
	8	1.0592	1.0886	2.0000	1.0148	1.0672	9.9395
	9	1.0213	1.1420	2.0000	1.0339	1.1532	15.7421
	10	1.0343	1.1284	2.0000	1.0508	1.1148	26.1052
	11	1.0504	1.0787	2.0000	1.0635	1.1235	37.4208
	12	1.0602	1.1851	2.0000	1.0535	1.1762	57.8956
0.50	6	1.0582	1.1219	1.9394	1.0040	1.1141	4.7158
	7	1.0935	1.1164	1.9692	1.0236	1.1014	7.2543
	8	1.0049	1.1342	1.9845	1.01051	1.0410	12.4032
	9	1.0579	1.1789	1.9922	1.0664	1.1950	19.6228
	10	1.0282	1.1619	1.9961	1.0996	1.2362	28.9121
	11	1.0393	1.1564	1.9980	1.0440	1.1641	43.7158
	12	1.0235	1.1516	1.9990	1.0240	1.1741	62.9854
0.85	6	1.0032	1.0249	1.3853	1.0687	1.1061	4.1458
	7	1.0137	1.1081	1.4522	1.0150	1.1043	8.8537
	8	1.0157	1.1778	1.5145	1.0117	1.1509	14.3855
	9	1.0131	1.1602	1.5717	1.0131	1.1624	24.4145
	10	1.0117	1.1019	1.6239	1.0167	1.1261	35.5458
	11	1.0141	1.1124	1.6711	1.0140	1.1343	45.8537
	12	1.0515	1.1479	1.7133	1.0127	1.1447	69.1254

Table 2 Computational results of the heuristics for $\tau = 0.5$

a	n	ρ_1		$\frac{1}{\alpha a^{n-1}+\beta}$	ρ_2		$\frac{\prod_{l=1}^{n}(1+p_l)-1}{C^*_{\max}}$
		Mean	Max		Mean	Max	Mean
0.15	6	1.0230	1.1410	1.9998	1.0024	1.0311	5.0965
	7	1.0285	1.1045	1.9999	1.0076	1.1378	9.2620
	8	1.0314	1.1528	2.0000	1.0071	1.1407	15.9691
	9	1.0317	1.1512	2.0000	1.0030	1.1528	25.7365
	10	1.0120	1.1510	2.0000	1.0324	1.1511	38.6965
	11	1.0265	1.1459	2.0000	1.0376	1.1078	50.1620
	12	1.0845	1.1159	2.0000	1.0176	1.1378	72.2967
0.50	6	1.0137	1.0458	1.9394	1.0581	1.0733	6.9076
	7	1.0003	1.0021	1.9692	1.0447	1.0145	10.8040
	8	1.0284	1.1017	1.9845	1.0719	1.1006	19.4152
	9	1.0191	1.1069	1.9922	1.0191	1.0429	28.3193
	10	1.0317	1.0628	1.9961	1.0081	1.0533	38.9076
	11	1.0103	1.0748	1.9980	1.0110	1.0258	50.8040
	12	1.0120	1.0598	1.9990	1.0000	1.0000	85.8040
0.85	6	1.0227	1.1465	1.3853	1.0838	1.1034	7.0598
	7	1.0123	1.12193	1.4522	1.0270	1.1307	15.1657
	8	1.0258	1.1565	1.5145	1.0429	1.1744	25.3542
	9	1.0258	1.1394	1.5717	1.0260	1.1306	36.4482
	10	1.0027	1.1165	1.6239	1.1038	1.1734	49.9598
	11	1.0023	1.0093	1.6711	1.0170	1.0407	58.1657
	12	1.0513	1.0993	1.7133	1.0070	1.0278	96.1657

EDD. The heuristic algorithms were coded in VC++ 6.0 and ran the computational experiments on a Pentium 4-2.4G personal computer with a RAM size of 1G. For all the tests, the values $t_0 = 0$. In addition, $a = 0.15, 0.50$, and 0.85, respectively. For each job J_j, the job deterioration rate p_j was generated from a uniform distribution over [1, 100], and the weight w_j was generated from a uniform distribution over [1, 10]. For each job J_j, the due date d_j was generated from a uniform distribution over $[1, \tau \frac{b}{c}(\prod_{l=1}^{n}(1 + c\alpha_l) - 1)$,

Table 3 Computational results of the heuristics for $\tau = 1$

a	n	ρ_1		$\frac{1}{\alpha a^{n-1}+\beta}$	ρ_2		$\frac{\prod_{l=1}^{n}(1+p_l)-1}{C^*_{\max}}$
		Mean	Max		Mean	Max	Mean
0.15	6	1.0035	1.1204	1.9998	1.0002	1.0424	8.0239
	7	1.0060	1.0195	1.9999	1.0044	1.1033	15.5601
	8	1.0020	1.0118	2.0000	1.0023	1.0719	24.9593
	9	1.0062	1.0725	2.0000	1.0059	1.0693	35.7481
	10	1.0035	1.0804	2.0000	1.0021	1.0124	48.0239
	11	1.0070	1.0095	2.0000	1.0011	1.0087	55.3323
	12	1.0020	1.0160	2.0000	1.0021	1.0127	76.5342
0.50	6	1.0095	1.0115	1.9394	1.0022	1.0423	8.8928
	7	1.0065	1.0403	1.9692	1.0147	1.0547	15.6122
	8	1.0031	1.0307	1.9845	1.0087	1.1101	25.4174
	9	1.0033	1.0083	1.9922	1.0132	1.0177	37.3129
	10	1.0012	1.0055	1.9961	1.0022	1.0423	49.8928
	11	1.0016	1.0140	1.9980	1.0089	1.0213	59.6122
	12	1.0078	1.0012	1.9990	1.0000	1.0000	80.2598
0.85	6	1.0010	1.0014	1.3853	1.0000	1.0000	9.1557
	7	1.0108	1.0124	1.4522	1.0956	1.0143	15.8152
	8	1.0012	1.0103	1.5145	1.0133	1.0160	24.4076
	9	1.0015	1.0140	1.5717	1.0015	1.0018	38.3697
	10	1.0012	1.0019	1.6239	1.0011	1.0040	51.1557
	11	1.0008	1.0044	1.6711	1.0010	1.0040	66.2135
	12	1.0002	1.0047	1.7133	1.0000	1.0000	98.1435

where $\tau \in \{0.25, 0.5, 1\}$ and $\alpha = \beta = 0.5, b = c = 1$. For each heuristic, seven different job sizes, $n = 6$, 7, 8, 9, 10, 11, and 12 were used. As a consequence, 42 experimental conditions were examined, and 20 replications were randomly generated for each condition. A total of 1,260 problems were tested. In order to study the effects of these parameters as well as to construct accurate and easily implemented algorithms, two heuristic algorithms are presented in this section. Each algorithm consists of two phases; the first phase involves generating an initial solution in a simple way, and the second phase further improves the quality of the solution by a neighborhood search, which provides good solutions and offers possibilities to be enhanced [20]. In the first step, jobs are sorted in non-decreasing order of the ratio $\frac{p_j}{w_j(1+p_j)}$ to obtain an initial solution. The second step is to improve the initial solution by using pairwise interchanges. In order to study the impact of the parameters, the mean and maximum of the ratio of the optimal solution and the WSPT (EDD) solution and the worst-case error bound are reported in Tables 1, 2, and 3. It is observed from Tables 1, 2, and 3 that the mean and maximum of the ratio of the optimal solution and the WSPT solution and the worst-case error bound for WSPT algorithm increase as the learning effect is stronger. It is noticed from Tables 1, 2, and 3 that the mean and maximum of the ratio of the optimal solution and the EDD solution decrease as the tardiness factor τ becomes larger. It is also found that the minimum and mean ratios equal one for some cases. In these cases, it is very easy to find a schedule such that all jobs can be finished before their due dates, yielding zero maximum tardiness. In addition, the ratio of the two solutions increases as the job size increases. This phenomenon is due to the fact that the learning effect becomes even stronger as the number of processed jobs grows. Our main goal was to evaluate the performance of the heuristics by comparing the heuristic solutions with the optimal solutions. The results are summarized in Tables 1, 2, and 3. From Tables 1, 2, and 3, we see that the performance of the heuristic algorithms is good.

6 Conclusions

We have considered in this paper two single-machine scheduling problems with the effect of deterioration and learning. For the weighted sum of completion times minimization problem and the maximum lateness minimization problem, we gave two heuristics according to the corresponding problems without learning effect. We also gave the worst-case error bound for the heuristics. Computational results show that the heuristic algorithms are very effective and efficient in obtaining near-optimal solutions. Future research may focus on determining the computational complexity of these two problems as they remain open, or proposing more sophisticated heuristics.

Acknowledgements We are grateful to the editor and two anonymous referees for their helpful comments on an earlier version of this paper. This research was supported by the National Natural Science Foundation of China (Grant No. 11001181).

References

1. Alidaee B, Womer NK (1999) Scheduling with time dependent processing times: review and extensions. J Oper Res Soc 50:711–720
2. Cheng TCE, Ding Q, Lin BMT (2004) A concise survey of scheduling with time-dependent processing times. Eur J Oper Res 152:1–13
3. Biskup D (2008) A state-of-the-art review on scheduling with learning effects. Eur J Oper Res 188:315–329
4. Wu C-C, Lee W-C, Shiau Y-R (2007) Minimizing the total weighted completion time on a single machine under linear deterioration. Int J Adv Manuf Technol 33:1237–1243
5. Shiau Y-R, Lee W-C, Wu C-C, Chang C-M (2007) Two-machine flowshop scheduling to minimize mean flow time under simple linear deterioration. Int J Adv Manuf Technol 34:774–782
6. Eren T, Guner E (2008) A bicriteria parallel machine scheduling with a learning effect. Int J Adv Manuf Technol. doi:10.1007/s00170-008-1436-2
7. Lee W-C, Wu C-C, Liu H-C (2009) A note on single-machine makespan problem with general deteriorating function. Int J Adv Manuf Technol. doi:10.1007/s00170-008-1421-9
8. Wang J-B, Wang L-Y, Wang D, Huang X, Wang X-R (2009) A note on single-machine total completion time problem with general deteriorating function. Int J Adv Manuf Technol. doi:10.1007/s00170-008-1918-2
9. Lee W-C (2004) A note on deteriorating jobs and learning in single-machine scheduling problems. Int J Bus Econ 3: 83–89
10. Wang J-B (2006) A note on scheduling problems with learning effect and deteriorating jobs. Int J Syst Sci 37:827–833
11. Wang J-B (2007) Single-machine scheduling problems with the effects of learning and deterioration. Omega 35:397–402
12. Wang J-B, Cheng TCE (2007) Scheduling problems with the effects of deterioration and learning. Asia-Pac J Oper Res 24:245–261
13. Wang X, Cheng TCE (2007) Single-machine scheduling with deteriorating jobs and learning effects to minimize the makespan. Eur J Oper Res 178:57–70
14. Toksar MD, Guner E (2007) Minimizing the earliness/tardiness costs on parallel machine with learning effects and deteriorating jobs: a mixed nonlinear integer programming approach. Int J Adv Manuf Technol. doi:10.1007/s00170-007-1128-3
15. Wang J-B, Huang X, Wang X-Y, Yin N, Wang L-Y (2009) Learning effect and deteriorating jobs in the single ma-

chine scheduling problems. Appl Math Model. doi:10.1016/j.apm.2009-01-004
16. Nembhard DA, Osothsilp N (2002) Task complexity effects on between-individual learning/forgetting variability. Int J Ind Ergon 29:297–306
17. Graham RL, Lawler EL, Lenstra JK, Rinnooy Kan AHG (1979) Optimization and approximation in deterministic sequencing and scheduling: a survey. Ann Discrete Math 5: 287–326
18. Zhao C, Zhang Q, Tang H (2003) Scheduling problems under linear deterioration. Acta Automatic Sinica 29:531–535
19. Cheng TCE, Wang G (2000) Single machine scheduling with learning effect considerations. Ann Oper Res 98:273–290
20. Gupta AK, Sivakumar AI (2005) Job shop scheduling techniques in semiconductor manufacturing. Int J Adv Manuf Technol 27:1163–1169
21. Pegels C-C (1969) On start-up or learning curves: an expanded view. AIIE Trans 1:316–322