Research Article

Single-Machine Scheduling with Aging Effects and Optional Maintenance Activity Considerations

Chou-Jung Hsu

Department of Industrial Engineering and Management, Nan Kai University of Technology, Nantou 542, Taiwan

Correspondence should be addressed to Chou-Jung Hsu; jrsheu@nkut.edu.tw

This paper explores a single-machine scheduling with aging effects and the problem regarding optional maintenance activity assignment. The jobs' processing time is assumed to follow a power position-dependent aging model. The optional maintenance activity refers to the situation in which the maintenance activity can be scheduled immediately after processing of any job has been completed except for the last job and the duration of maintenance activity can be of any value from zero to a fixed time interval. A recovery function is proposed to reflect the efficiency of the machine or worker which is improved. The objective of this study is to decide whether and when to implement the maintenance activity into the job sequence, how long the duration of maintenance activity is, and how to schedule so as to minimize the makespan. Once the duration of maintenance activity is known, we introduce an efficient solution for this problem. In addition, when the maintenance activity is completely performed, we showed that the optimal policy is to schedule the maintenance activity in the middle of the task sequence and optimally solved it by lower order algorithm. Finally, we extend the problem to the case of multiple maintenance activities which are completely performed. Hence, the problem is regarded as polynomial time solvable.

1. Introduction

Scheduling problems with aging (deterioration or fatigue) effects have been extensively studied over two decades in various machine environments and performance measures. The processing time of a job is an increasing function of its position to be processed in its starting time. The increasing model reflects many realistic situations such as steel production [1], firefighting [2], national defense [3], medical procedure process [4], maintenance [5], or heavy reliance on labors production system [6], where any delay in processing a job may take a longer time as time passes in accomplishing the job. According to some scheduling problems with dynamic parameters considered by Gupta et al. [7], J. N. D. Gupta and S. K. Gupta [1] explored an interesting scheduling model in which the processing time of a task is a polynomial function of its starting time. Browne and Yechiali [8] presented an optimal solution for expected makespan minimization problem of single-machine scheduling. Since then, many researchers have focused their attention on the topic of time-dependent processing procedure. The reader may refer to the extensive surveys by Cheng et al. [3] and Biskup [9] and a book by Gawiejnowicz [10].

We introduced some papers which have only related scheduling jobs with aging (deterioration) effect and availability constraint simultaneously as follows. Wu and Lee [11] considered the single-machine scheduling problem with an availability constraint under linear deteriorating jobs. The objective was to minimize the makespan. They showed that this problem can be transformed into an integer programming and a linear equation problem and further simplified it to a pure integer programming problem under certain conditions. Ji et al. [12] explored the same problem proposed by Wu and Lee [11] under the jobs with nonresumable consideration. The objectives were to minimize the makespan and the total completion time. They proved that both problems are NP-hard and presented pseudopolynomial time optimal algorithms to solve the problems. Low et al. [13] studied a single machine scheduling problem with an availability constraint under simple linear deterioration. They considered the nonpreemptive case and a preventive maintenance activity; that is, the starting time of the maintenance activity

is known in advance. The objective was also to minimize the makespan. They solved this problem via the pure integer programming technique and proposed heuristics for the small-sized and large-sized problems, respectively. Hsu et al. [14] considered the same problem as that proposed by Low et al. [13]. The objective was to minimize the total completion time. They proved that the complexity of the problem belongs to NP-hard in the strong sense and presented an efficient heuristic. S.-J. Yang and D.-L. Yang [15] explored single-machine scheduling problems with the aging or deteriorating effect under a deteriorating maintenance activity consideration. They developed polynomial time algorithms for all the studied problems. Levin et al. [16] considered the problem of scheduling a maintenance activity on parallel identical machines, under the assumption that all the machines must be maintained simultaneously. The objective was minimum flow time. The problem is shown to be NP-hard, and moreover impossible to approximate unless $P = \text{NP}$. They introduced a pseudopolynomial dynamic programming algorithm, and showed how to convert it into a bicriteria FPTAS for this problem. Finally, they also presented a lower bound and an efficient heuristic for the problem. Zhao and Tang [17] considered a single-machine scheduling problem with job-dependent aging effects and multimaintenance activities. For the objective was makespan, they provided a polynomial time algorithm to solve the problem. Lodree and Geiger [6] explored a single-machine scheduling with time-dependent processing times and a rate-modifying activity. The goal was to derive the optimal policy for assigning a single rate-modifying activity in a sequence to minimize the makespan. They proved that, under certain conditions, the optimal policy is to schedule the rate-modifying activity in the middle of the job sequence. S.-J. Yang and D.-L. Yang [18] studied single-machine scheduling with simultaneous consideration of job-dependent aging effects, multimaintenance activities, and variable maintenance durations to minimize the makespan. They showed that all the studied problems are polynomially solvable. S.-J. Yang and D.-L. Yang [19] further assessed single-machine scheduling with aging or deteriorating effects and deteriorating maintenance activities simultaneously to minimize the total completion time. They showed that the problem remains polynomially solvable if an upper bound on the maintenance frequency is given. Wang and Wei [20] examined identical parallel machines scheduling problems with a deteriorating maintenance activity. In their study model, each machine has a deteriorating maintenance activity. That is, delaying the maintenance increases the time required to perform it for their objective was to minimize the total absolute differences in completion time and the total absolute differences in waiting time. Their problems remained polynomially solvable. Mor and Mosheiov [21] studied a scheduling problem with job-dependent due-window based on common flow allowance where the scheduler has the option to perform a maintenance activity to improve the processing times of the following jobs. Their objective was to minimize the total of earliness, tardiness, the starting time of the due-windows, and the size of the due-windows for each job cost. They claimed that the running time did not exceed $O(n^4)$. Yin et al. [22]

explored a single-machine batch delivery scheduling and the common due date assignment problem with a rate-modifying activity. The objective was to find a common due date for all the jobs, a location of the rate-modifying activity, and a delivery date for each job to minimize the sum of earliness, tardiness, holding, due date, and delivery cost. They provided some properties of the optimal schedule for the problem and presented polynomial algorithms for some special cases.

Production scheduling and maintenance planning are the most common and significant problems faced by the manufacturing industry. In the real production settings, machines are usually not continuously available. For instance, in the manufacturing industry, a machine may not be available in the scheduling period due to a preventive maintenance, tool change, or others for assuring that the product retains its high quality. The stop time interval will affect the objectives of production system. There are numerous papers on this theme. For details on this stream of research, the reader may refer to the comprehensive surveys by Schmidt [23], Wang [24], and Ma et al. [25].

The scheduling problem explored in this paper was motivated by the manufacturing of the metal processing industry [26] and by human operators who experienced fatigue while carrying out tasks. They took rest breaks for recovery and this was applicable to machines that experienced performance degradation over time and required maintenance in order to sustain acceptable production rates [6]. Lee and Leon [26] introduced a class of scheduling problems in which a rate-modifying activity may be performed. The rate-modifying activity is an activity that occupies the machine for a given duration and changes the production rate of the machine. No job can be processed while the rate-modifying activity is being performed. Hence the decisions under consideration are whether and when to schedule the rate-modifying activity and how to sequence the jobs in order to minimize some scheduling performance measures. In metal processing industry, the pertinent manufacturing operations include cold drawing, cold pressing, cold forming, and cold extrusion [14]. Basically, a cold formed product is made from a molten iron ingot extracted from an electric furnace at a very high temperature. The ingot has to be annealed to a suitable temperature before it can be processed on. Therefore, the initial annealing time of the ingot can be regarded as a setup time. As products are made sequentially, the later a product is made, the more the ingot has been cooled and the machine has aged, and the longer time it takes to produce it. Therefore, arranging the maintenance activity may be necessary in order to achieve production objective. In this paper, we investigated a single-machine scheduling problem with an optional availability constraint. We also consider a special case of the proposed problem and extend the proposed problem to multiple availabilities constraint.

2. Notations and Problem Formulation

The problem rendered in this paper can be formally described as follows. There are n independent jobs $J = \{J_1, J_2, \ldots, J_n\}$ to be processed on a single-machine scheduling. The jobs are non-preemptive and they are all available for processing

at time zero. The machine can handle at most one job at a time and cannot stand idle until the last job assigned to it has finished processing unless a maintenance activity is being conducted. Let p_i and $p_{[i]}$ denote the basic processing time for J_i and the basic processing time for the job scheduled in the ith position in a sequence ($i = 1, 2, \ldots, n$), respectively. The job processing time is assumed to follow a power position-dependent aging model [27]. Thus, if J_i is scheduled in the rth position in a sequence, its actual processing time is defined as

$$p_{ir} = p_i r^a, \quad i, r = 1, 2, \ldots, n, \tag{1}$$

where $a > 0$ is aging factor.

For the characteristics of the aging effects, assume that there is at most one maintenance activity which is allowed throughout the planning horizon. It can be scheduled immediately after the processing of any job has been completed except for the last job. The basic duration of the maintenance activity is T. Assume that the actual duration of the maintenance activity is t that is, $0 \le t \le T$. Therefore, the position (starting time) of the maintenance (rest) activity and the actual duration of the maintenance activity are decision variables. One named such category of maintenance activity as the optional maintenance activity (OMA). Let k denote the position of the optional maintenance activity; that is, the optional maintenance activity is scheduled immediately after the kth job has been completed. The recovery function is defined as

$$R(t) = \frac{T - t}{T} \left[p_i r^a - p_i (r - k)^a \right], \tag{2}$$

$$1 \le k \le n - 1, \qquad k + 1 \le r \le n.$$

Therefore, based on the recovery function, the actual processing of J_i, if it is scheduled in position r in a sequence, is given by

$$p_{ir}(k, t)$$
$$= \begin{cases} p_i r^a & 1 \le r \le k, \\ p_i (r - k)^a + R(t) & k + 1 \le r \le n, \, 0 \le t \le T, \end{cases}$$

$$p_{ir}(k, t)$$
$$= \begin{cases} p_i r^a & 1 \le r \le k, \\ p_i r^a \left(1 - \dfrac{t}{T} \right) \\ \quad + p_i (r - k)^a \left(\dfrac{t}{T} \right) & k + 1 \le r \le n, \, 0 \le t \le T. \end{cases} \tag{3}$$

According to the above equation in above, one has the following scenarios.

(1) If $t = 0$ and $1 \le r \le n$, then $p_{ir}(k, 0) = p_i r^a$. This means that there is no maintenance activity which is scheduled throughout the planning horizon.

(2) If $0 < t < T$ and $k + 1 \le r \le n$, then $p_{ir}(k, t) = p_i r^a (1 - (t/T)) + p_i (r - k)^a (t/T)$. This means that the maintenance activity is partially performed in the planning horizon.

(3) If $t = T$ and $k + 1 \le r \le n$, then $p_{ir}(k, T) = p_i (r - k)^a$. This means that the maintenance activity is completely performed in the planning horizon.

By using the general notation for scheduling problems, the problem is denoted by a triplet $1 | p_{ir}(k, t), \text{OMA} | C_{\max}$.

3. Optimal Solutions

Since the length of maintenance activity (break time) affects the extent of recovery and the efficiency of the operations management, in order to achieve the objective, production manager needs to decide an optimal policy depending on whether to implement the maintenance activity, when to implement the maintenance activity, and how long for performing the maintenance activity. In this section, we will analyze the problem accordingly. First, a useful lemma which will be applied to solve the problem is given as follows.

Lemma 1. *Let there be two sequences of numbers x_i and y_i. The sum $\sum_i x_i y_i$ of products of the corresponding elements is the least (largest) if the sequences are monotonic in the opposite (same) sense.*

Proof. See page 261 in Hardy et al. [28]. □

For any job sequence, if the OMA is scheduled at position k ($1 \le k \le n - 1$) and the length is t, then the makespan is calculated as follows:

$$C_{\max}(k, t) = \sum_{r=1}^{k} p_{[r]} r^a + t \\ + \sum_{r=k+1}^{n} p_{[r]} \left[r^a \left(1 - \frac{t}{T} \right) + (r - k)^a \frac{t}{T} \right], \tag{4}$$

where $p_{[r]}$ denotes the basic processing time of a job that is scheduled in the rth position in the sequence. Clearly, $C_{\max}(k, 0) = \sum_{r=1}^{n} p_{[r]} r^a$. The positional weight of rth position in a sequence, r^a, is an increasing function of r. By **Lemma 1**, the following theorem holds.

Theorem 2. *For the $1 | p_{ir}(k, t), \text{OMA} | C_{\max}$ problem, if the maintenance activity is not scheduled (i.e., $t = 0$) into the job sequence, then it is optimally to sequence jobs in nonincreasing order of their basic processing time (p_i).*

Theorem 3. *For the specific schedule $p_{[1]} \ge p_{[2]} \ge \cdots \ge p_{[n]}$, if there exists a k so that $T < \sum_{r=k+1}^{n} p_{[r]} [r^a - (r - k)^a]$, then the OMA should be scheduled into the job sequence.*

Proof. Clearly, the OMA should be scheduled to the position k in the sequence (for $t > 0$) only if the resulting makespan less than that of the optimal solution associating with not scheduling an OMA. That is, $C_{\max}(k, t) < C_{\max}^*(k, 0)$. Consider

$$C_{\max}(k, t) - C_{\max}^*(k, 0) \\ = t - \sum_{r=k+1}^{n} p_{[r]} [r^a - (r - k)^a] \frac{t}{T} < 0. \tag{5}$$

This implies $T < \sum_{r=k+1}^{n} p_{[r]} [r^a - (r - k)^a]$. □

Once the duration of maintenance activity (t) is known, by Theorem 3, we can obtain the optimal solution via Algorithm 4.

Algorithm 4. Consider the following.

Step 1. Initialize $0 < t \leq T$, $C^*_{\max} = \infty$, and $k^* = 0$.

Step 2. Order the jobs in LPT rule; that is, $p_{[1]} \geq p_{[2]} \geq \cdots \geq p_{[n]}$.

Step 3. For k from 1 to $\lfloor n/2 \rfloor$ do the following.

Step 3.1. Calculate the positional weights w_j ($j = 1, 2, \ldots, n$) according to (4).

Step 3.2. Assign the job with the largest basic processing time to the position with the smallest value of positional weight w_j, the job with the next largest basic processing time to the position with the next smallest value of positional weight w_j, and so forth. If there is a tie, break it arbitrarily.

Step 3.3. Calculate the function $C_{\max}(k, t)$ in (4).

Step 3.4. If $C_{\max}(k, t) < C^*_{\max}$, then $C^*_{\max} := C_{\max}(k, t)$ and $k^* := k$.

Step 4. The optimal position of OMA and makespan are k^* and $C^*_{\max}$, respectively.

The time complexity of Step 1 is $O(1)$ and the time complexity of Step 2 is $O(n \log n)$. Since the symmetry, the OMA can be scheduled immediately after any job that before half in the job sequence; $\lfloor n/2 \rfloor$ different positions must be evaluated to obtain the global optimal solution. The time complexity of Step 4 is $O(1)$. Thus, the computational complexity of the problem does not exceed $O(n^2)$ time. We conclude that the following theorem holds.

Theorem 5. *For the $1|p_{ir}(k, t), OMA|C_{\max}$ problem, if an OMA is scheduled into the job sequence and the duration of maintenance activity is known, $0 < t \leq T$, then the computational complexity of the problem does not exceed $O(n^2)$ time.*

Next, one derives the optimal policy for assigning an adequate length of t in a suitable position k when the job processing time is represented as (3) and the objective of minimizing the makespan.

Theorem 6. *If the OMA is scheduled in position k, then the optimal policy is to perform the maintenance activity completely; that is, $t = T$.*

Proof. Clearly, (4) can be rewritten as

$$
\begin{aligned}
C_{\max}(k, t) &= \sum_{r=1}^{n} p_{[r]} r^a + t + \sum_{r=k+1}^{n} p_{[r]} \left[(r-k)^a - r^a\right] \frac{t}{T} \\
&= \sum_{r=1}^{n} p_{[r]} r^a + t \left\{ 1 - \frac{1}{T} \sum_{r=k+1}^{n} p_{[r]} \left[r^a - (r-k)^a\right] \right\}.
\end{aligned}
\tag{6}
$$

By Theorem 3, the last item in (6) is less than zero. That is, $\{1 - (1/T) \sum_{r=k+1}^{n} p_{[r]} [r^a - (r-k)^a]\} < 0$. For any job sequence, to minimize the value of (6) is to maximize the value of t. Therefore, if the OMA is scheduled, then the optimal policy is $t = T$. □

Comment. Theorem 6 reveals that to schedule a partial maintenance activity into job sequence seems meaninglessly. However, when the unit cost between maintenance activity and job processing time is differently, the partial recovery model may be meaningful.

Theorem 7. *If the length of the maintenance activity is $t = T$ then the optimal policy is to schedule the maintenance activity in the middle of the task sequence; that is, $k = \lfloor n/2 \rfloor$.*

Proof. Let $W_k = [1^a, 2^a, \ldots, k^a, 1^a, 2^a, \ldots, (n-k)^a]$ denote the positional weight vector of the maintenance activity which is scheduled at position k. If the optimal position of OMA is scheduled at position k and applied to Lemma 1, then $C^*_{\max}(k, T) \leq C^*_{\max}(k-1, T)$ and $C^*_{\max}(k, T) \leq C^*_{\max}(k+1, T)$. This implies that $k^a \leq (n-k+1)^a$ and $(n-k)^a \leq (k+1)^a$. Because $a > 0$, thus $k \leq n - k + 1$ and $n - k \leq k + 1$. We have $(n-1)/2 \leq k \leq (n+1)/2$. Since k is an integer, it follows that k is the largest integer less than or equal to $n/2$; that is, $k = \lfloor n/2 \rfloor$. □

Applying to the weight-matching approach, the following theorem holds immediately.

Theorem 8. *Problem $1|p_{ir}(k, t), t = T, OMA|C_{\max}$ can be solved in $O(n \log n)$ time.*

Assume $p_1 = p_{[1]} \geq p_2 = p_{[2]} \geq \cdots \geq p_n = p_{[n]}$. Then, the optimal solution of the problem $1|p_{ir}(k, t), t = T, OMA|C_{\max}$ is calculated by

$$
C^{*(T,1)}_{\max} = \begin{cases} T + \sum_{i=1}^{k} (p_{2i-1} + p_{2i}) i^a & n \text{ is even} \\[2ex] T + p_n (k+1)^a + \sum_{i=1}^{k} (p_{2i-1} + p_{2i}) i^a & n \text{ is odd}. \end{cases}
\tag{7}
$$

Example 9. For the convenience, we assume that there are 10 independent jobs to be processed on a single-machine scheduling, with the aging rate $a = 0.1$, the standard duration of maintenance activity $T = 10$, and the basic processing time of jobs $p_i = 5 \times (11 - i)$, $i = 1, 2, \ldots, 10$. Clearly, the job sequence is a nonincreasing order; that is, $p_1 = p_{[1]} \geq p_2 = p_{[2]} \geq \cdots \geq p_{10} = p_{[10]}$. If the maintenance activity is not considered (i.e., $t = 0$) to schedule into the job sequence, by Theorem 2, then it is optimally to sequence jobs in non-increasing order of their basic processing time; that is, $C^*_{\max}(k, 0) = 309.74$. Applying to the package of the Microsoft Office Excel, we can find an integer k ($k = 2$) so that $T < \sum_{r=k+1}^{n} p_{[r]} [r^a - (r-k)^a]$ that is, $10 < \sum_{r=3}^{10} p_{[r]} [r^{0.1} - (r-2)^{0.1}] = 12.06$. By Theorem 3, this result reveals that to schedule a maintenance activity into job sequence is beneficial to

the objective of makespan. Therefore, if $t = T$, by (7), the optimal solution equals 304.59; that is, $C_{\max}^{*(T,1)} = 304.59$. If $t = 6$, by Algorithm 4, the optimal solution equals 307.42; that is, $C_{\max}^{*}(4, 6) = 307.42$.

Finally, one explores the case of m times maintenance activities completely performed, that is, $t = T$, in the planning horizon. This is an extension of the problem $1|p_{ir}(k, t), t = T, \text{OMA}|C_{\max}$. We denote it by $1|p_{ir}(k, t), t = T, \text{OMA} = m|C_{\max}$, where $t = T$ and $\text{OMA} = m$ in the second field means there are m times maintenance activities be completely performed in the planning horizon.

Let $\pi = (\pi_1, \text{OMA}_1, \pi_2, \text{OMA}_2, \ldots, \pi_m, \text{OMA}_m, \pi_{m+1})$, where π_i denotes the subsequence of n_i jobs; that is, $\sum_{i=1}^{m+1} n_i = n$. The positional weight in π_i is an increasing function according to the job order; that is, $(1^a, 2^a, \ldots, n_i^a)$. To minimize the makespan, one can easily proof that the difference of the number of jobs between π_i and π_j $(i \neq j)$ is at most 1; that is, $n_i - n_j = 0$ or $|n_i - n_j| = 1$. Therefore, the scheduling problem of $1|p_{ir}(k, t), t = T, \text{OMA} = m|C_{\max}$ can be solved in $O(n \log n)$ time by Algorithm 10 as follows.

Algorithm 10. Consider the following.

Step 1. Consider $l = \lfloor n/(m + 1) \rfloor$, and $c = n - l \times (m + 1)$.

Step 2. Order the jobs in LPT rule; that is, $p_{[1]} \geq p_{[2]} \geq \cdots \geq p_{[n]}$.

Step 3. Assign the first $m + 1$ jobs to the first position of $\pi_1, \pi_2, \ldots$, and π_{m+1} accordingly, assign the second $m + 1$ jobs to the second position of $\pi_1, \pi_2, \ldots$, and π_{m+1} accordingly,..., assign the lth $m + 1$ jobs to the lth position of $\pi_1, \pi_2, \ldots$, and π_{m+1} accordingly.

Step 4. If $c \neq 0$, then assign the last c jobs to the $(l+1)$-th position of $\pi_1, \pi_2, \ldots$, and π_c accordingly.

Step 5. The optimal solution is calculated by

$$
C_{\max}^{*(T,l)} = \begin{cases} T \times l + \sum_{i=1}^{l} \left(\sum_{j=1}^{m+1} p_{[(i-1)(m+1)+j]} \right) i^a & c = 0 \\ T \times l + \sum_{i=1}^{l} \left(\sum_{j=1}^{m+1} p_{[(i-1)(m+1)+j]} \right) i^a & \\ \quad + \sum_{j=1}^{c} p_{[l(m+1)+j]} (l + 1)^a & c \neq 0. \end{cases} \tag{8}
$$

Comparing at most $(l$-1) times of (8), one can derived the optimal times of maintenance activities.

Example 11. Assume that there are 30 independent jobs to be processed on a single-machine scheduling, with the aging rate $a = 0.1$, the standard duration of maintenance activity $T = 80$, and the basic processing time of jobs $p_i = 5 \times (31 - i)$, $i = 1, 2, \ldots, 30$. Assume that there are 3 times of maintenance activities which are completely performed in the planning horizon at most. If the maintenance activity is not considered to schedule into the job sequence, by Theorem 2, then it is

optimally to sequence jobs in nonincreasing order of their basic processing time; that is, $C_{\max}^{*}(k, 0) = 2867.44$. Applying to the package of the Microsoft Office Excel, one can find an integer k ($k = 3$) so that $T < \sum_{r=k+1}^{n} p_{[r]}[r^a - (r - k)^a]$; that is, $80 < \sum_{r=4}^{30} p_{[r]}[r^{0.1} - (r - 2)^{0.1}] = 96.39$. By Theorems 3, 6, and 7 and by (8), one has $C_{\max}^{*(T,1)} = 2779.69$, $C_{\max}^{*(T,2)} = 2772.78$, and $C_{\max}^{*(T,3)} = 2796.66$. These results reveal that scheduled two times of maintenance activities are completely performed into job sequence which is beneficial to the objective of makespan.

4. Conclusions

This paper examined a recovery function for the position-dependent aging with an optional maintenance activity. Once the duration of maintenance activity is known, one showed that the proposed problems are polynomial time solvable. Furthermore, if the maintenance activity is completely performed, one concluded that the optimal policy is to schedule the maintenance activity in the middle of the task sequence and it may optimally be solved by lower order algorithm. Next, one investigated the case of multiple maintenance activities which are completely performed in the planning horizon. Result showed that the problem is also polynomial time solvable. To sum up, by applying the optimal solution function, the decision maker may easily find the optimal times of maintenance activities.

It is worthwhile for future research to consider the problem with other regular performance measures, due-window related topics, or multimachine settings. Extensions the recovery function to the job-dependent and/or time-dependent processing time models are worth investigating, not only in the context of our problem but for all traditional objective functions.

References

[1] J. N. D. Gupta and S. K. Gupta, "Single facility scheduling with nonlinear processing times," *Computers & Industrial Engineering*, vol. 14, no. 4, pp. 387–393, 1988.

[2] A. S. Kunnathur and S. K. Gupta, "Minimizing the makespan with late start penalties added to processing times in a single facility scheduling problem," *European Journal of Operational Research*, vol. 47, no. 1, pp. 56–64, 1990.

[3] T. C. E. Cheng, Q. Ding, and B. M. T. Lin, "A concise survey of scheduling with time-dependent processing times," *European Journal of Operational Research*, vol. 152, no. 1, pp. 1–13, 2004.

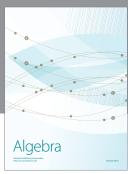[4] G. Mosheiov, "Multi-machine scheduling with linear deterioration," *INFOR*, vol. 36, no. 4, pp. 205–214, 1998.

[5] M. A. Kubzin and V. A. Strusevich, "Two-machine flow shop no-wait scheduling with machine maintenance," *4OR*, vol. 3, no. 4, pp. 303–313, 2005.

[6] E. J. Lodree Jr. and C. D. Geiger, "A note on the optimal sequence position for a rate-modifying activity under simple linear deterioration," *European Journal of Operational Research*, vol. 201, no. 2, pp. 644–648, 2010.

[7] S. K. Gupta, A. S. Kunnathur, and K. Dandapani, "Optimal repayment policies for multiple loans," *Omega*, vol. 15, no. 4, pp. 323–330, 1987.

[8] S. Browne and U. Yechiali, "Scheduling deteriorating jobs on a single processor," *Operations Research*, vol. 38, no. 3, pp. 495–498, 1990.

[9] D. Biskup, "A state-of-the-art review on scheduling with learning effects," *European Journal of Operational Research*, vol. 188, no. 2, pp. 315–329, 2008.

[10] S. Gawiejnowicz, *Time-Dependent Scheduling*, Springer, New York, NY, USA, 2008.

[11] C.-C. Wu and W.-C. Lee, "Scheduling linear deteriorating jobs to minimize makespan with an availability constraint on a single machine," *Information Processing Letters*, vol. 87, no. 2, pp. 89–93, 2003.

[12] M. Ji, Y. He, and T. C. E. Cheng, "Scheduling linear deteriorating jobs with an availability constraint on a single machine," *Theoretical Computer Science*, vol. 362, no. 1–3, pp. 115–126, 2006.

[13] C. Low, C.-J. Hsu, and C.-T. Su, "Minimizing the makespan with an availability constraint on a single machine under simple linear deterioration," *Computers & Mathematics with Applications*, vol. 56, no. 1, pp. 257–265, 2008.

[14] C.-J. Hsu, C. Low, and C.-T. Su, "Single-machine scheduling problem with an availability constraint under simple linear deterioration," *Journal of the Chinese Institute of Industrial Engineers*, vol. 27, no. 3, pp. 189–198, 2010.

[15] S.-J. Yang and D.-L. Yang, "Single-machine scheduling problems with aging/deteriorating effect under an optional maintenance activity consideration," *INFOR*, vol. 48, no. 3, pp. 171–179, 2010.

[16] A. Levin, G. Mosheiov, and A. Sarig, "Scheduling a maintenance activity on parallel identical machines," *Naval Research Logistics*, vol. 56, no. 1, pp. 33–41, 2009.

[17] C.-L. Zhao and H.-Y. Tang, "Single machine scheduling with general job-dependent aging effect and maintenance activities to minimize makespan," *Applied Mathematical Modelling*, vol. 34, no. 3, pp. 837–841, 2010.

[18] S.-J. Yang and D.-L. Yang, "Minimizing the makespan on single-machine scheduling with aging effect and variable maintenance activities," *Omega*, vol. 38, no. 6, pp. 528–533, 2010.

[19] S.-J. Yang and D.-L. Yang, "Minimizing the total completion time in single-machine scheduling with aging/deteriorating effects and deteriorating maintenance activities," *Computers & Mathematics with Applications*, vol. 60, no. 7, pp. 2161–2169, 2010.

[20] J.-B. Wang and C.-M. Wei, "Parallel machine scheduling with a deteriorating maintenance activity and total absolute differences penalties," *Applied Mathematics and Computation*, vol. 217, no. 20, pp. 8093–8099, 2011.

[21] B. Mor and G. Mosheiov, "Scheduling a maintenance activity and due-window assignment based on common flow allowance," *International Journal of Production Economics*, vol. 135, no. 1, pp. 222–230, 2012.
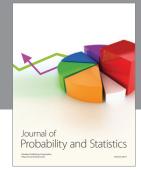
[22] Y. Yin, T. C. E. Cheng, D. Xu, and C.-C. Wu, "Common due date assignment and scheduling with a rate-modifying activity to minimize the due date, earliness, tardiness, holding, and batch delivery cost," *Computers & Industrial Engineering*, vol. 63, no. 1, pp. 223–234, 2012.

[23] G. Schmidt, "Scheduling with limited machine availability," *European Journal of Operational Research*, vol. 121, no. 1, pp. 1–15, 2000.

[24] H. Wang, "A survey of maintenance policies of deteriorating systems," *European Journal of Operational Research*, vol. 139, no. 3, pp. 469–489, 2002.

[25] Y. Ma, C. Chu, and C. Zuo, "A survey of scheduling with deterministic machine availability constraints," *Computers & Industrial Engineering*, vol. 58, no. 2, pp. 199–211, 2010.

[26] C.-Y. Lee and V. J. Leon, "Machine scheduling with a rate-modifying activity," *European Journal of Operational Research*, vol. 128, no. 1, pp. 119–128, 2001.

[27] D. Biskup, "Single-machine scheduling with learning considerations," *European Journal of Operational Research*, vol. 115, no. 1, pp. 173–178, 1999.

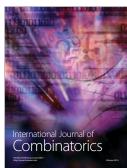[28] G. H. Hardy, J. E. Littlewood, and G. Polya, *Inequalities*, Cambridge University Press, London, UK, 1967.