Hindawi
Mathematical Problems in Engineering
Volume 2017, Article ID 7532174, 6 pages
https://doi.org/10.1155/2017/7532174

Research Article

Single-Machine Scheduling with Learning Effects and Maintenance: A Methodological Note on Some Polynomial-Time Solvable Cases

Kuo-Ching Ying,[1] Chung-Cheng Lu,[2] Shih-Wei Lin,[3,4,5] and Jie-Ning Chen[6]

[1] *Department of Industrial Engineering and Management, National Taipei University of Technology, Taipei, Taiwan*
[2] *Department of Transportation and Logistics Management, National Chiao Tung University, Taipei, Taiwan*
[3] *Department of Information Management, Chang Gung University, Taoyuan, Taiwan*
[4] *Department of Industrial Engineering and Management, Ming Chi University of Technology, Taipei, Taiwan*
[5] *Department of Neurology, Linkou Chang Gung Memorial Hospital, Taoyuan, Taiwan*
[6] *Wistron Corporation, Taipei, Taiwan*

Correspondence should be addressed to Shih-Wei Lin; swlin@mail.cgu.edu.tw

Received 11 March 2017; Revised 22 May 2017; Accepted 21 June 2017; Published 8 August 2017

Academic Editor: Josefa Mula

This work addresses four single-machine scheduling problems (SMSPs) with learning effects and variable maintenance activity. The processing times of the jobs are simultaneously determined by a decreasing function of their corresponding scheduled positions and the sum of the processing times of the already processed jobs. Maintenance activity must start before a deadline and its duration increases with the starting time of the maintenance activity. This work proposes a polynomial-time algorithm for optimally solving two SMSPs to minimize the total completion time and the total tardiness with a common due date.

1. Introduction

The single-machine scheduling problem (SMSP) is one of the most extensively studied classical scheduling problems owing to its wide range of applications in many realistic systems [1]. The excellent surveys of Abdul-Razaq et al. [2], Yen and Wan [3], Pinedo [4], and Adamu and Adewumi [5] have detailed the literature on the theory and applications of various SMSPs in the past several decades. Most SMSPs assume that job processing times are fixed and known throughout the process, and SMSPs with learning effects constitute a relatively new subfield in the area of scheduling.

Wright [6] scientifically examined the processing time of a single unit continuously decreasing with the processing of additional units in the aircraft industry; in the literature, this effect is known as the learning effect. Gawiejnowicz [7] was probably the first researcher who applied the learning effect to scheduling problems. Since learning effects are common in both manufacturing and service sectors, many studies have been devoted to investigating it following the pioneering work of Gawiejnowicz [7]. Depending on the production environment, learning effects can be classified into two categories [8]: *position-based* learning effects and the *sum-of-processing-times-based* learning effects. The former depend only on the number of jobs being processed; the latter depend on the sum of the normal processing times of the already processed jobs.

With respect to position-based learning effects, Biskup [9] modified the original power formula, $p_{j,[k]} = p_{[1]}k^a$, to formulate the basic position-based learning model, $p_{j,[k]} = p_j k^a$, where $p_{j,[k]}$ is the actual processing time of job j if it is scheduled in position k ($k = 2, \ldots, n$); $p_{[1]}$ denotes the processing time of the first unit; p_j represents the normal processing time of job j, and $a = \log_2 \mathrm{LR} \le 0$ is the learning index, which depends on the learning rate (LR). Biskup demonstrated that the SMSP with basic position-based learning and the goal of minimizing total completion times can be optimally solved in polynomial time by using the shortest processing time (SPT) rule. Since then, many studies of SMSPs with various performance measures have been performed using the basic position-based learning model of Biskup [9]. The most well-known ones include those of Mosheiov [10, 11], Lee et al. [12], Bachman and Janiak [13], Zhao et al. [14], Kuo and Yang [15], and Wu et al. [16]. Some extensions of the basic position-based learning model have been presented, including the consideration of job-dependent position-based learning effects [17–19], autonomous position-based and induced learning effects [20], position-based learning and deteriorating effects [21, 22], and position-based linear learning effects [13, 23, 24]. In all of the aforementioned position-based learning models, the actual processing time of a job decreases as its scheduled position k increases.

With respect to sum-of-processing-time-based learning effects, Kuo and Yang [25, 26] were among the pioneers that considered the processing times of jobs that had already been preprocessed and proposed two learning models, $p_{j,[k]} = p_j(1+\sum_{i=1}^{k-1} p_{[i]})^a$ and $p_{j,[k]} = p_j(\sum_{i=1}^{k-1} p_{[i]})^a$, for SMSPs with time-dependent learning. Yang and Kuo [27] later developed another model, $p_{j,[k]} = p_j(1+\sum_{i=1}^{k-1} p_{[i]}^A)^a$, that is based on the sum of actual processing times, $p_{[i]}^A$. Koulamas and Kyparisis [28] also proposed a sum-of-processing-time-based learning model for SMSPs: $p_{j,[k]} = p_j(1-\sum_{i=1}^{k-1} p_{[i]}/\sum_{i=1}^{n} p_{[i]})^b = p_j(\sum_{i=k}^{n} p_{[i]}/\sum_{i=1}^{n} p_{[i]})^b$, where $b \geq 1$ is the learning index. Based on this model, Biskup [8] indicated that the gains from learning are extremely high and suggested that $b \geq 0$ would be more appropriate. Wang [29] considered an SMSP with the learning model that was developed by Koulamas and Kyparisis and proved that an optimal schedule that minimizes the sum of the square completion times can be obtained using the SPT rule. All of the aforementioned sum-of-processing-time-based learning models have in common the fact that the actual processing time of a job cannot be calculated without data on the normal or actual processing times of its preceding jobs, so the analysis is more difficult than that based on position-based learning models. An excellent survey by Biskup presents more detailed discussions regarding SMSPs with learning effects [8].

Both position-based learning and sum-of-processing-times-based learning approaches are valid and numerous practical examples of each have been presented. Biskup [8] indicated that position-based learning usually involves operations that are independent of processing time, such as the setting up of machines, while sum-of-processing-times-based learning generally involves the experience that is gained by workers who are involved in the jobs. However, in some situations, position-based learning and sum-of-processing-times-based learning exist simultaneously, such as those that involve a robot with a neural network system, as is used in many assembly lines and has been discussed by Lee and Wu [30]. This fact has motivated recent studies [30–35] that consider both learning effects at once. SMSPs with position-based learning and/or the sum-of-processing-times-based learning have also been studied from different perspectives to develop models that are as realistic as possible [36–40].

Since the 1990s, the scheduling of jobs with fixed maintenance [41] or variable maintenance [42] in a manufacturing system has attracted much attention. For scheduling with fixed maintenance, the starting time and the maintenance duration are fixed and given in advance. In scheduling with variable maintenance, the starting time is set by the scheduler, and the maintenance duration is a nondecreasing function of the starting time [43]. For a comprehensive survey, refer to Ma et al. [43]. SMSPs with learning and maintenance commonly arise in realistic manufacturing systems. If learning and maintenance simultaneously occur in an SMSP, then they will critically and enormously influence scheduling. However, learning effects and maintenance activity have seldom been simultaneously studied in relation to SMSPs.

To incorporate more practically important factors into scheduling, this work proposes a polynomial-time algorithm for solving two SMSPs, which takes into account both learning effects and variable maintenance activity. The objectives are to minimize the total completion time and the total tardiness with a common due date. The remainder of this paper is organized as follows. Section 2 defines the four SMSPs that are considered herein. Section 3 elucidates in detail the proposed polynomial-time algorithm. Finally, Section 4 concludes by offering recommendations for future studies.

2. Problem Definition

The SMSPs of interest are characterized by a set, $J = \{J_1, J_2, \ldots, J_n\}$, of n independent jobs that are nonpreemptively processed on a single machine. All jobs are available for processing at the beginning of the planning horizon, which is time zero. Each job J_j ($j = 1, 2, \ldots, n$) has a normal processing time (without any learning effects), p_j, if the job is sequenced first in a schedule. The following learning model, proposed by Low and Lin [36], applies generally to practical SMSPs.

$$\begin{aligned} p_{j,[k]} &= p_j\left(1-\frac{\sum_{i=1}^{k-1} p_{[i]}}{\sum_{i=1}^{n} p_i}\right)^a b^{k-1} \\ &= p_j\left(\frac{\sum_{i=k}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i}\right)^a b^{k-1}, \quad j,k = 1,2,\ldots,n, \end{aligned} \tag{1}$$

where $p_{j,[k]}$ is the actual processing time of job J_j if it is sequenced in position k in a schedule; $a \geq 1$ and $0 < b < 1$ are the learning indices, and $p_{[k]}$ is the normal processing time of the job in position k. In (1), the actual processing time of a job is simultaneously influenced by the traditional sum-of-processing-time-based learning effect, $p_j(1-\sum_{i=1}^{k-1} p_{[i]}/\sum_{i=1}^{n} p_{[i]})^b$, and the general position-based learning effect, b^{k-1}. Equation (1) clearly reveals that position-based learning and sum-of-processing-times-based learning are special cases with $b = 1$ and $a = 1$, respectively.

The machine can handle only one job at a time and is continuously available during the planning horizon, except for the mandatory variable maintenance (VM) activity, which is performed at the beginning of the planning horizon or between two consecutive processed jobs; in that case, the starting time of the maintenance activity, s, must precede a specified deadline, s_d (i.e., $s \leq s_d$). Notably, the maintenance activity cannot be performed after the completion time of the final job (i.e., $s_d \leq \sum_{k=1}^{n-1} p_{j,[k]}$), because if it were so, the variable maintenance activity would effectively be neglected. The duration of maintenance for a given schedule π, t^{π}_{VM}, is a positive and nondecreasing function of its starting time s: $t^{\pi}_{\mathrm{VM}} = f(s)$ and $f(s_b) \geq f(s_a)$ for all $s_b > s_a$.

For a given schedule, let C_j be the completion time of job J_j, $j = 1, 2, \ldots, n$. The tardiness of job J_j ($j = 1, 2, \ldots, n$) is $T_j = \max\{0, C_j - d_j\}$, where d_j is the due date of job J_j. The objective is to determine the starting time of the maintenance activity and the sequence of all jobs that optimize the total completion time ($\sum C_j$) and the total tardiness ($\sum T_j$) with a common due date, respectively. According to the three-field classification scheme of Graham et al. [44], the two SMSPs of interest are represented as $1, \mathrm{VM}|\mathrm{LE}|\sum C_j$ and $1, \mathrm{VM}|\mathrm{LE}, d_j = d|\sum T_j$, respectively, where 1 denotes the SMSP; VM stands for the variable maintenance, and LE stands for learning effects.

It is important to note that the SMSP with the total flow time, $\sum F_j$, and identical release dates, $r_j = r$, is equivalent to the SMSP with the total completion time, $\sum C_j$; that is, $\sum F_j = \sum (C_j - r) = \sum C_j - nr$, so the $1, \mathrm{VM}|\mathrm{LE}, r_j = r|\sum F_j$ problem is equivalent to the $1, \mathrm{VM}|\mathrm{LE}|\sum C_j$ problem. Besides, the SMSP with the mean lateness, $(1/n)\sum L_j$, is also equivalent to the SMSP with the total completion time, $\sum C_j$; that is, $(1/n)\sum L_j = (1/n)\sum C_j - (1/n)D$ (where $D = \sum d_j$). Therefore, the $1, \mathrm{VM}|\mathrm{LE}|\overline{L}$ problem as well as the $1, \mathrm{VM}|\mathrm{LE}|\sum L_j$ problem is equivalent to the $1, \mathrm{VM}|\mathrm{LE}|\sum C_j$ problem. Hence, the proposed polynomial-time algorithm can also be used for optimally solving these equivalent SMSPs.

3. Polynomial-Time Algorithm

The following three theorems provide obvious properties for sequencing an optimal schedule in the two SMSPs under study.

Theorem 1. *In the $1, VM|LE|\sum C_j$ problem, there exists an optimal schedule that is obtained by sequencing jobs in the SPT order.*

Theorem 2. *In the $1, VM|LE, d_j = d|\sum T_j$ problem, there exists an optimal schedule that is obtained by sequencing jobs in the SPT order.*

Proof. The maintenance activity is carried out among a pair of consecutive jobs. The job interchange method can easily be used to reveal that the optimal schedules in the $1, \mathrm{VM}|\mathrm{LE}|\sum C_j$ and $1, \mathrm{VM}|\mathrm{LE}, d_j = d|\sum T_j$ problems are those consistent with the SPT order. Here, only two jobs, j' and j, which are scheduled before and after the maintenance activity, are considered. Their schedule is denoted as $\pi = (\ldots, j', \mathrm{VM}, j, \ldots)$. The positions of jobs j' and j can be swapped, keeping the other jobs in the same positions, to generate a new schedule, $\pi' = (\ldots, j, \mathrm{VM}, j', \ldots)$. If $p_{j'} > p_j$, then the duration of the maintenance activity is lower in π', $t^{\pi'}_{\mathrm{VM}} < t^{\pi}_{\mathrm{VM}}$, because the maintenance activity in π' will start before that in π. Therefore, the completion time of job j' in π' will be less than the completion time of job j in π, so $C^{\pi'}_{j'} < C^{\pi}_{j}$. Based on these results, the proofs of Theorems 1 and 2 are as follows. □

Proof of Theorem 1. Let $TC^{\pi}_{j\&j'}$ and $TC^{\pi'}_{j\&j'}$ denote the sums of the completion times of jobs j and j' in π and π', respectively. Now,

$$\begin{aligned} TC^{\pi}_{j\&j'} - TC^{\pi'}_{j\&j'} &= \left[\left(C^{\pi}_{j'} + C^{\pi}_{j}\right) - \left(C^{\pi'}_{j} + C^{\pi'}_{j'}\right)\right] \\ &= \left(p_{j'}\left(\frac{\sum_{i=k}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i}\right)^a b^{k-1} + t^{\pi}_{\mathrm{VM}}\right. \\ &\quad \left. + p_j\left(\frac{\sum_{i=k}^{n} p_{[i]} - p_{j'}}{\sum_{i=1}^{n} p_i}\right)^a b^k\right) \\ &\quad - \left(p_j\left(\frac{\sum_{i=k}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i}\right)^a b^{k-1} + t^{\pi'}_{\mathrm{VM}}\right. \\ &\quad \left. + p_{j'}\left(\frac{\sum_{i=k}^{n} p_{[i]} - p_j}{\sum_{i=1}^{n} p_i}\right)^a b^k\right) = \left(p_{j'} - p_j\right) \\ &\quad \cdot \left(\frac{\sum_{i=k}^{n} p_{[i]}}{\sum_{i=1}^{n} p_i}\right)^a b^{k-1} + \left(t^{\pi}_{\mathrm{VM}} - t^{\pi'}_{\mathrm{VM}}\right) \\ &\quad + p_j\left(\frac{\sum_{i=k}^{n} p_{[i]} - p_{j'}}{\sum_{i=1}^{n} p_i}\right)^a b^k - p_{j'}\left(\frac{\sum_{i=k}^{n} p_{[i]} - p_j}{\sum_{i=1}^{n} p_i}\right)^a \\ &\quad \cdot b^k. \end{aligned} \tag{2}$$

Substituting $x = p_j / \sum_{i=1}^{n} p_i$, $y = \sum_{i=k}^{n} p_{[i]} / \sum_{i=1}^{n} p_i$, and $\lambda = p_{j'}/p_j$ into (2) yields

$$\begin{aligned} &TC^{\pi}_{j\&j'} - TC^{\pi'}_{j\&j'} \\ &\quad = p_j y^a b^{k-1}\left[(1 - \lambda x)^a b - \lambda(1 - x)^a b + \lambda - 1\right] \\ &\qquad + \left(t^{\pi}_{\mathrm{VM}} - t^{\pi'}_{\mathrm{VM}}\right). \end{aligned} \tag{3}$$

Let $f(\lambda) = (1 - \lambda x)^a b - \lambda(1 - x)^a b + \lambda - 1$; then,

$$\begin{aligned} f'(\lambda) &= -ax(1 - \lambda x)^{a-1} b - (1 - x)^a b + 1, \\ f''(\lambda) &= a(a - 1)x^2(1 - \lambda x)^{a-2} b. \end{aligned} \tag{4}$$

Since $\lambda \geq 1$, $a \geq 1$, $0 < x \leq 1$, and $0 < b < 1$, $f''(\lambda) \geq 0$. Thus, $f'(\lambda)$ is an increasing function.

Let $g(x) = -ax(1 - x)^{a-1} b - (1 - x)^a b + 1$; now, $g'(x) = a(a - 1)x(1 - x)^{a-2} b + a(1 - x)^{a-1} b \geq 0$, for $a \geq 1$, $0 < x \leq 1$, and $0 < b < 1$.

Since $g(x) \geq g(0) = 1 - b > 0$, for $0 < b < 1$, $g(x)$ is increasing for $a \geq 1$, $0 < x \leq 1$, and $0 < b < 1$. Therefore, $g(x) = -ax(1-x)^{a-1}b - (1-x)^{a}b + 1 > 0$, and $f'(1) = -ax(1-x)^{a-1}b - (1-x)^{a}b + 1 \geq 0$.

Therefore, $f'(\lambda) \geq f'(1) \geq 0$ for $\lambda \geq 1$, $a \geq 1$, $0 < x \leq 1$, and $0 < b < 1$. Hence, $f(\lambda)$ is an increasing function for $\lambda \geq 1$, $a \geq 1$, $0 < x \leq 1$, and $0 < b < 1$. Since $f(\lambda) \geq f(1) = 0$, $(1 - \lambda x)^{a}b - \lambda(1-x)^{a}b + \lambda - 1 > 0$ for $\lambda \geq 1$, $a \geq 1$, $0 < x \leq 1$, and $0 < b < 1$, so the first term in (3) is nonnegative. Also, the second term in (3) $(t^{\pi}_{\mathrm{VM}} - t^{\pi'}_{\mathrm{VM}}) > 0$ if $p_{j'} > p_{j}$. Clearly, $TC^{\pi}_{j\&j'} - TC^{\pi'}_{j\&j'} > 0$ if $p_{j'} > p_{j}$, implying that the total completion time of π is larger than that of π', completing the proof of Theorem 1. □

Proof of Theorem 2. Let $\sum T^{\pi}_{j\&j'}$ and $\sum T^{\pi'}_{j\&j'}$ denote the average tardiness of jobs j and j' in π and π', respectively. Now,

$$\begin{aligned} &\sum T^{\pi}_{j\&j'} - \sum T^{\pi'}_{j\&j'} \\ &\quad = \left[\max\left\{0, \left(C^{\pi}_{j} - d\right)\right\} + \max\left\{0, \left(C^{\pi}_{j'} - d\right)\right\}\right] \\ &\qquad - \left[\max\left\{0, \left(C^{\pi'}_{j} - d\right)\right\} + \max\left\{0, \left(C^{\pi'}_{j'} - d\right)\right\}\right]. \end{aligned} \tag{5}$$

Obviously, $\sum T^{\pi}_{j\&j'} - \sum T^{\pi'}_{j\&j'} > 0$ because $(C^{\pi}_{j} + C^{\pi}_{j'}) > (C^{\pi'}_{j} + C^{\pi'}_{j'})$ if $p_{j'} > p_{j}$, implying that the total tardiness of π is larger than that of π', completing the proof of Theorem 2. □

Based on Theorems 1 and 2, the following simple algorithm (named YLLC algorithm) is proposed to determine the optimal starting time of the variable maintenance activity and the sequences of all jobs in the two SMSPs of interest.

Algorithm YLLC

Step 1. Sort all jobs in the SPT order based on normal processing times to generate a sequence $\pi = (J_{[1]}, J_{[2]}, \ldots, J_{[n]})$ with $p_{[1]} \leq p_{[2]} \leq \cdots \leq p_{[n]}$, where $p_{[k]}$ $(k = 1, 2, \ldots, n)$ is the normal processing time of the job with the kth processing priority.

Step 2. Use (1) to calculate the actual processing time, $p^{A}_{[k]}$ $(k = 1, 2, \ldots, n)$, of each job with the kth processing priority in π.

Step 3. For a given deadline s_d for the start of the maintenance activity, compute $\overline{k}$ that satisfies $\sum_{k=1}^{\overline{k}} p^{A}_{[k]} \leq s_d < \sum_{k=1}^{\overline{k}+1} p^{A}_{[k]}$.

Step 4. Construct the candidate schedules $\pi^{k} = (J_{[0]}, \ldots, J_{[k-1]}, \mathrm{VM}, J_{[k]}, \ldots, J_{[n]})$, $k = 1, 2, \ldots, \overline{k} + 1$, where $J_{[0]}$ is a dummy job. Select the optimal schedule among the candidate schedules, which has the lowest objective function value.

Notably, proof of Theorem 1 also reveals the existence of an optimal schedule to the $1, \mathrm{VM}|\mathrm{LE}|C_{\max}$ problem, which is obtained by sequencing jobs in the SPT order. Hence, the optimal solution to the $1, \mathrm{VM}|\mathrm{LE}|C_{\max}$ problem can also be obtained by using the proposed YLLC algorithm. However, the criteria of maximum lateness and the makespan with release dates are not considered in this study because the SMSP with anyone of these criteria are strongly NP-hard even with simple linear position learning models (without maintenance) [45].

Theorem 3. *Algorithm YLLC solves the* $1, VM|LE|\sum C_j$ *and* $1, VM|LE, d_j = d|\sum T_j$ *problems in* $O(n^2)$ *time.*

Proof. Sort the jobs using the SPT rule, requiring $O(n \log n)$ time. Given the SPT order and a starting time of the maintenance activity, the objective function value of a candidate schedules can be computed in $O(n)$ time. The maximum number of possible starting times for maintenance is n, so the objective function value is computed not more than n times. Hence, algorithm YLLC solves each of the four SMSPs in $O(n^2)$ time. □

4. Conclusions and Recommendations for Future Studies

Learning effects and maintenance activity are critical factors in many scheduling environments but have seldom been studied in relation to SMSPs. In an attempt to address real-world situations, this work considered two SMSPs with learning effects and variable maintenance activity. The objectives were to minimize the total completion time and the total tardiness. An algorithm for solving the two SMSPs was developed, and the SMSPs were thus proved to be solvable in polynomial time with the explicit consideration of learning and the variable maintenance. According to the results, the developed algorithm is useful to practitioners in guiding them to solve the relevant SMSPs.

The SMSP with learning effects and variable maintenance is not purely hypothetical but arises in real scheduling decisions. It will continue to attract the attention of researchers and practitioners. This study suggests various potential directions for further research. First, the proposed algorithm could be extended to solve SMSPs with more general functions of learning effects as well as with the aging/deteriorating effect. Second, different objective functions for the SMSP with learning effects and maintenance considerations should be considered in the light of the results herein. Third, an SMSP with learning effects and maintenance in which a biobjective function value is minimized would be an interesting topic of research. Fourth, an SMSP with learning effects and multiple variable maintenance activities during the planning horizon would also be an interesting topic of research. Finally, extending the problems to the jobs with different release dates or due dates would be complex but warrants further works.

Conflicts of Interest

There are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The first author of the work acknowledges the Ministry of Science and Technology of Taiwan for financially supporting this research, Grant MOST106-2221-E-027-085. The corresponding author of the work acknowledges the Ministry of Science and Technology of Taiwan and the Linkou Chang Gung Memorial Hospital for financially supporting this research, Grants MOST105-2410-H-182-009-MY2 and CMRPD3G0011, respectively.

References

[1] C.-C. Lu, K.-C. Ying, and S.-W. Lin, "Robust single machine scheduling for minimizing total flow time in the presence of uncertain processing times," *Computers and Industrial Engineering*, vol. 74, no. 1, pp. 102–110, 2014.

[2] T. S. Abdul-Razaq, C. N. Potts, and L. N. Van Wassenhove, "A survey of algorithms for the single machine total weighted tardiness scheduling problem," *Discrete Applied Mathematics. The Journal of Combinatorial Algorithms, Informatics and Computational Sciences*, vol. 26, no. 2-3, pp. 235–253, 1990.

[3] B. P.-C. Yen and G. Wan, "Single machine bicriteria scheduling: a survey," *International Journal of Industrial Engineering: Theory Applications and Practice*, vol. 10, no. 3, pp. 222–231, 2003.

[4] M. L. Pinedo, *Scheduling: Theory, Algorithms, and Systems*, Springer, New York, NY, USA, 2012.

[5] M. O. Adamu and A. O. Adewumi, "A survey of single machine scheduling to minimize weighted number of tardy jobs," *Journal of Industrial and Management Optimization*, vol. 10, no. 1, pp. 219–241, 2014.

[6] T. P. Wright, "Factors affecting the cost of airplanes," *Journal of the Aeronautical Sciences*, vol. 3, no. 4, pp. 122–128, 1936.

[7] S. a. Gawiejnowicz, "A note on scheduling on a single processor with speed dependent on a number of executed jobs," *Information Processing Letters*, vol. 57, no. 6, pp. 297–300, 1996.

[8] 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.

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

[10] G. Mosheiov, "Scheduling problems with a learning effect," *European Journal of Operational Research*, vol. 132, no. 3, pp. 687–693, 2001.

[11] G. Mosheiov, "Parallel machine scheduling with a learning effect," *Journal of the Operational Research Society*, vol. 52, no. 10, pp. 1165–1169, 2001.

[12] W.-C. Lee, C.-C. Wu, and H.-J. Sung, "A bi-criterion single-machine scheduling problem with learning considerations," *Acta Informatica*, vol. 40, no. 4, pp. 303–315, 2004.

[13] A. Bachman and A. Janiak, "Scheduling jobs with position-dependentprocessing times," *Journal of the Operational Research Society*, vol. 55, no. 3, pp. 257–264, 2004.

[14] C. L. Zhao, Q. L. Zhang, and H. Y. Tang, "Machine scheduling problems with a learning effect," *Dynamics of Continuous, Discrete and Impulsive Systems, Series A: Mathematical Analysis*, vol. 11, no. 5-6, pp. 741–750, 2004.

[15] W.-H. Kuo and D.-L. Yang, "Single machine scheduling with past-sequence-dependent setup times and learning effects," *Information Processing Letters*, vol. 102, no. 1, pp. 22–26, 2007.

[16] C. Wu, W. Lee, and T. Chen, "Heuristic algorithms for solving the maximum lateness scheduling problem with learning considerations," *Computers and Industrial Engineering*, vol. 52, no. 1, pp. 124–132, 2007.

[17] G. Mosheiov and J. B. Sidney, "Scheduling with general job-dependent learning curves," *European Journal of Operational Research*, vol. 147, no. 3, pp. 665–670, 2003.

[18] G. Mosheiov and J. B. Sidney, "Note on scheduling with general learning curves to minimize the number of tardy jobs," *Journal of the Operational Research Society*, vol. 56, no. 1, pp. 110–112, 2005.

[19] B. M. T. Lin, "Complexity results for single-machine scheduling with positional learning effects," *Journal of the Operational Research Society*, vol. 58, no. 8, pp. 1099–1102, 2007.

[20] D. Biskup and D. Simons, "Common due date scheduling with autonomous and induced learning," *European Journal of Operational Research*, vol. 159, no. 3, pp. 606–616, 2004.

[21] J.-B. Wang, "A note on scheduling problems with learning effect and deteriorating jobs," *International Journal of Systems Science. Principles and Applications of Systems and Integration*, vol. 37, no. 12, pp. 827–833, 2006.

[22] J. Wang, "Single-machine scheduling problems with the effects of learning and deterioration," *Omega*, vol. 35, no. 4, pp. 397–402, 2007.

[23] T. C. Cheng and G. Wang, "Single machine scheduling with learning effect considerations," *Annals of Operations Research*, vol. 98, no. 1, pp. 273–290, 2000.

[24] J.-B. Wang and Z.-Q. Xia, "Flow-shop scheduling with a learning effect," *Journal of the Operational Research Society*, vol. 56, no. 11, pp. 1325–1330, 2005.

[25] W.-H. Kuo and 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*, vol. 174, no. 2, pp. 1184–1190, 2006.

[26] W.-H. Kuo and D.-L. Yang, "Minimizing the makespan in a single machine scheduling problem with a time-based learning effect," *Information Processing Letters*, vol. 97, no. 2, pp. 64–67, 2006.

[27] D.-L. Yang and W.-H. Kuo, "Single-machine scheduling with an actual time-dependent learning effect," *Journal of the Operational Research Society*, vol. 58, no. 10, pp. 1348–1353, 2007.

[28] C. Koulamas and G. J. Kyparisis, "Single-machine and two-machine flowshop scheduling with general learning functions," *European Journal of Operational Research*, vol. 178, no. 2, pp. 402–407, 2007.

[29] J.-B. Wang, "Single-machine scheduling with general learning functions," *Computers & Mathematics with Applications. An International Journal*, vol. 56, no. 8, pp. 1941–1947, 2008.

[30] W.-C. Lee and C.-C. Wu, "Some single-machine and m-machine flowshop scheduling problems with learning considerations," *Information Sciences. An International Journal*, vol. 179, no. 22, pp. 3885–3892, 2009.

[31] T. C. Cheng, C.-C. Wu, and W.-C. Lee, "Some scheduling problems with sum-of-processing-times-based and job-position-based learning effects," *Information Sciences. An International Journal*, vol. 178, no. 11, pp. 2476–2487, 2008.

[32] C. C. Wu and W. C. Lee, "Single-machine and flowshop scheduling with a general learning effect model," *Computers & Industrial Engineering*, vol. 56, no. 4, pp. 1553–1558, 2009.

[33] Y. Yin, D. Xu, K. Sun, and H. Li, "Some scheduling problems with general position-dependent and time-dependent learning

effects," *Information Sciences. An International Journal*, vol. 179, no. 14, pp. 2416–2425, 2009.

[34] X. Zhang and G. Yan, "Machine scheduling problems with a general learning effect," *Mathematical and Computer Modelling*, vol. 51, no. 1-2, pp. 84–90, 2010.

[35] P.-J. Lai and W.-C. Lee, "Single-machine scheduling with general sum-of-processing-time-based and position-based learning effects," *Omega*, vol. 39, no. 5, pp. 467–471, 2011.

[36] C. Low and W.-Y. Lin, "Some scheduling problems with time-dependent learning effect and deteriorating jobs," *Applied Mathematical Modelling. Simulation and Computation for Engineering and Environmental Systems*, vol. 37, no. 20-21, pp. 8865–8875, 2013.

[37] C.-H. Wu, P.-J. Lai, and W.-C. Lee, "A note on single-machine scheduling with sum-of-processing-time-based learning and forgetting effects," *Applied Mathematical Modelling. Simulation and Computation for Engineering and Environmental Systems*, vol. 39, no. 2, pp. 415–424, 2015.

[38] S.-C. Liu, W.-L. Hung, and C.-C. Wu, "Note on a single-machine scheduling problem with sum of processing times based learning and ready times," *Mathematical Problems in Engineering*, vol. 2015, Article ID 452602, 9 pages, 2015.

[39] Y.-B. Wu and J.-J. Wang, "Single-machine scheduling with truncated sum-of-processing-times-based learning effect including proportional delivery times," *Neural Computing and Applications*, vol. 27, no. 4, pp. 937–943, 2016.

[40] A. Azadeh, H. Habibnejad-Ledari, S. Abdolhossein Zadeh, and M. Hosseinabadi Farahani, "A single-machine scheduling problem with learning effect, deterioration and non-monotonic time-dependent processing times," *International Journal of Computer Integrated Manufacturing*, vol. 30, no. 2-3, pp. 292–304, 2016.

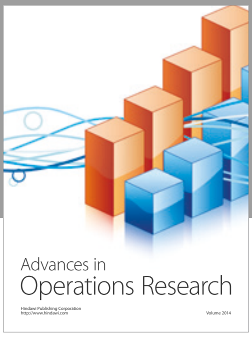
[41] C.-Y. Lee and S. D. Liman, "Single machine flow-time scheduling with scheduled maintenance," *Acta Informatica*, vol. 29, no. 4, pp. 375–382, 1992.

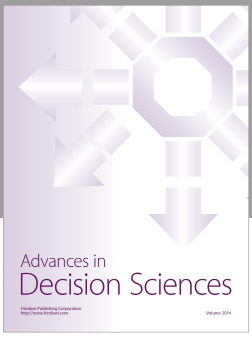
[42] M. A. Kubzin and V. A. Strusevich, "Planning machine maintenance in two-machine shop scheduling," *Operations Research*, vol. 54, no. 4, pp. 789–800, 2006.

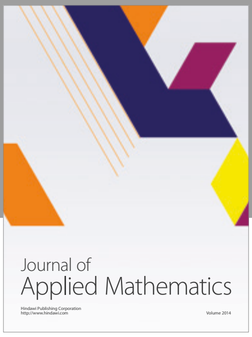
[43] Y. Ma, C. B. Chu, and C. R. Zuo, "A survey of scheduling with deterministic machine availability constraints," *Computers and Industrial Engineering*, vol. 58, no. 2, pp. 199–211, 2010.

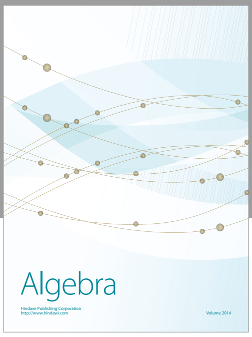
[44] R. L. Graham, E. L. Lawler, J. K. Lenstra, and A. H. Rinnooy Kan, "Optimization and approximation in deterministic sequencing and scheduling: a survey," *Annals of Discrete Mathematics*, vol. 5, pp. 287–326, 1979.

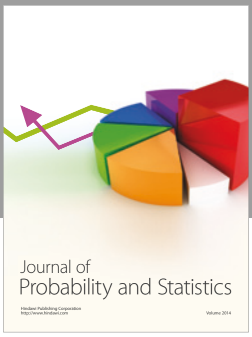
[45] R. Rudek, "Scheduling problems with position dependent job processing times: computational complexity results," *Annals of Operations Research*, vol. 196, pp. 491–516, 2012.


Advances in
Operations Research

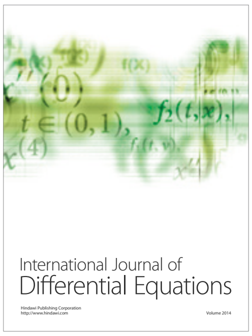

Advances in
Decision Sciences

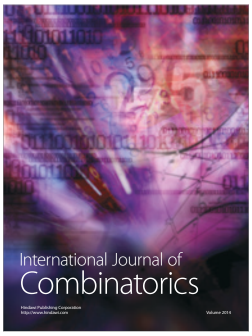

Journal of
Applied Mathematics

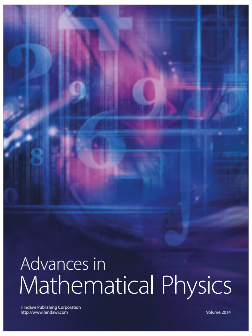

Algebra

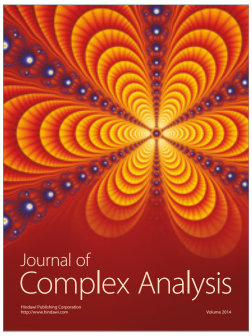

Journal of
Probability and Statistics

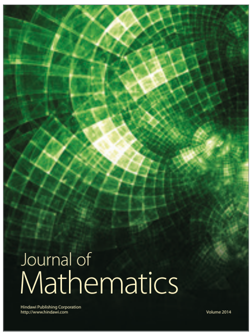
The Scientific
World Journal

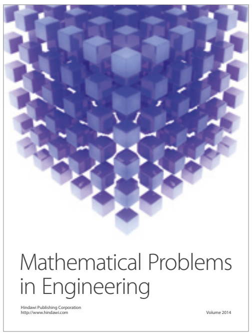

International Journal of
Differential Equations

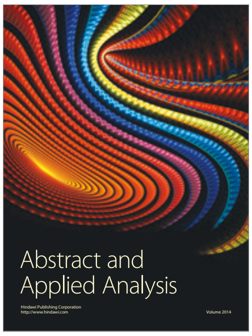

International Journal of
Combinatorics

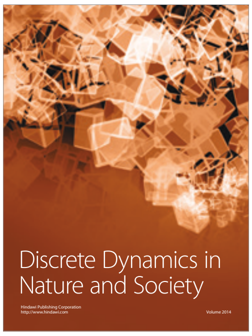

Advances in
Mathematical Physics

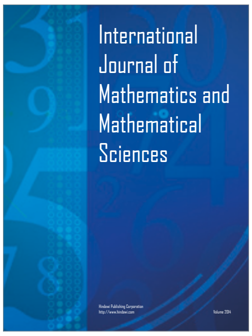

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

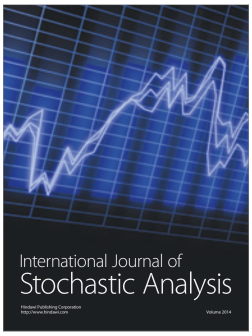

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization