Gazi University Journal of Science
GU J Sci
26(3):389-397 (2013)

Minimizing Completion Time Variance in a Flowshop Scheduling Problem with a Learning Effect

Tamer EREN[1,♠]

[1]*Kırıkkale University Faculty of Engineering Department of Industrial Engineering 71451, Kırıkkale*

Received: 11.07.2012 Accepted: 12.12.2012

ABSTRACT

In this paper, flowshop scheduling problem with a learning effect is considered. The objective function of the problem is minimizing completion times variance. A non-linear programming model is developed for the problem. Also the model is tested on an example. Results of computational tests show that the proposed model is effective in solving problems with up to 30 jobs. The overall average solution error of the heuristic algorithm is 2 %. Processing of the 30 jobs case requires only 0.1 s on average to obtain an ultimate or even optimal solution. To solve the large sizes problems up to 500 jobs, heuristics methods were used. The performances of heuristics about the solution error were evaluated with the non-linear programming model results for small size problems and each other for large size problems. According to results, the special heuristic for all number of jobs was the more effective than others. The heuristic scheduling algorithm is more practical to solve real world applications than the non-linear programming model.

Key words: flowshop scheduling, learning effect, completion time variance, non-linear programming model, heuristic methods

1. INTRODUCTION

The completion time variance problem was first proposed by Merten and Muller [l], motivated by the file organization problem in computing systems, where it is desirable to provide uniform response times to users' requests to retrieve data files. It may be shown that other scheduling problems may fit into the same mathematical model. A typical example is the just-in-time production in advanced manufacturing systems [2]. Most of these studies done in the area concern a single machine case. For examples: Schrage [3], Kubiak [4], Eilon and Chowdhury [5], Bagchi et al. [6], Gupta et al. [7], Mittenthal et al. [8], Gupta et al. [9], Ventura and Weng [10], Manna and Prasad [11], De et al. [12], Prasad et al. [13], Federgruen and Mosheiov [14] and Gajpal and Rajendran [15].

Another area of research in the scheduling literature involves the learning effect problem. In traditional machine scheduling problems, job processing times are assumed to be constant for all jobs. However, recent empirical studies in several industries have verified that unit costs decline as firms produce more of a product and gain knowledge or experience. For instance, repeated processing of similar tasks improves worker skills; workers are able to perform setup, to deal with machine operations and software, or to handle raw materials and components at a greater pace [16]. This phenomenon is known as the "learning effect." Biskup [16] was the first to investigate the learning effect in scheduling problems. He assumed a learning process that reflects a decrease in the process time as a function of the number of repetitions i.e. as a function of the job position in the sequence. Biskup [16] showed that

♠Corresponding author, e-mail: teren@kku.edu.tr

single-machine scheduling problems with a learning effect still remain polynomially solvable if the objective is to minimize the deviation from a common due date or to minimize the sum of flow times. Later, Mosheiov [17] continued to study Biskup's [16] model and introduced a polynomial solution to the single-machine makespan minimization problem. In addition, he showed that the classical solutions in some other scheduling problems do not hold when the learning effect is taken into consideration. Mosheiov[18] further considered the scheduling problem of minimizing flow time on identical parallel machines. He showed that the problem has a solution which is polynomial in the number of jobs. Eren and Güner [19] analyzed a scheduling problem with job-dependent learning effect in a two-machine flowshop with makespan by performance measure. They showed that Johnson [20] algorithm cannot guarantee the best results in the situation with job-dependent learning effect. They also proposed a mixed integer programming model for this problem. In addition, studies about a learning effect in a two-machine flowshop scheduling with total completion times has been considered by Lee and Wu [21] and Eren and Güner [22]. Lee et al. [23] analyzed the single-machine bicriteria problem with a learning effect. The objective was to find a sequence that minimizes a linear combination of the total completion time and the maximum tardiness. Eren and Güner [24] worked on the well-known single-machine total tardiness problem with learning effects. Eren and Güner [25] worked on a different bicriteria single-machine scheduling problem, namely jointly minimizing the sum of completion times and the total tardiness, $1/LE/\alpha\sum C+\beta\sum T$. The authors developed a mathematical programming and heuristics to solve this problem. Eren and Güner [26] analyzed the bicriteria flowshop problem with a learning effect. The objective was to find a sequence that minimizes a linear combination of the total completion time and the makespan.

Xu et al. [27] considered the same problem of Wang [28]. The objectives were to respectively minimize the following objectives: the sum of weighted completion times, the discounted total weighted completion times, and the sum of the quadratic job completion times. They proposed heuristic algorithms by applying the optimal permutations of the corresponding single machine scheduling problems to the flowshop scheduling problems. They also gave the worst-case bound of the heuristic algorithms. Wu and Lee [29] consider a permutation flowshop scheduling problem with learning effects to minimize the sum of completion times or flow times. Wang and Liu [30] consider a two-machine flow shop scheduling problem with both deterioration and learning effects. Cheng et al. [31] consider two-machine flowshop scheduling problem with a truncated learning function in which the actual processing time of a job is a function of the job's position in a schedule and the learning truncation parameter. The objective is to minimize the makespan. We propose a branch-andbound and three crossover-based genetic algorithms (GAs) to find the optimal and approximate solutions, respectively, for the problem. Kuo et al. [32] consider some flowshop scheduling problems with the time-dependent learning effect. The following objective functions are respectively investigated: the makespan, the total flowtime, the sum of weighted completion times, the sum of the kth power of completion times and the maximum lateness.

In this paper, a scheduling problem with a learning effect on two-machine flowshop is considered. The objective function of the problem is minimization of the completion time variance. A non-linear programming model is developed for the problem. Heuristic methods are also used to solve large size problems.

In the classical scheduling notation, this problem is denoted by $F_2/LE/\sum_{j=1}^{n}(C_j-\overline{C})^2$. It is strongly NP-hard, even if single machine and without a learning effect[33].

The rest of the study is organized as follows: In Sections 2 and 3, the problem and the proposed a non-linear programming model are described. Heuristic methods that are used to solve large size problems are presented in Section 4. The experimental results are given in Section 5. Finally, Section 6 provides conclusions and evaluations of the study.

2. PROBLEM DESCRIPTION

Let $n=\{J_1,J_2,...,J_n\}$ be the set of jobs to be scheduled and $M=\{M_1,M_2\}$ be the two-machines. The normal processing times for job *j* on M_1 and M_2 are denoted p_{j1} and p_{j2}, respectively. Furthermore, we assume that both machines have the different learning effects. That is if p_{j1r} and p_{j2r} are the actual processing times of job *j* scheduled in position *r* in a sequence, then $p_{j1r}=p_{j1}r^a$ and $p_{j2r}=p_{j2}r^a$ (where $a\le 0$ is learning effect, given is the logarithm to the base 2 of the learning rate) the objective is to find a schedule that minimizing completion times variance in a two-machine flowshop. Let $\left(C_j-\overline{C}\right)^2$ is completion time variance of the job *j*. The minimize completion time variance with a learning effect in two-machine flowshop scheduling problem is shown as $F_2/LE/\sum_{j=1}^{n}(C_j-\overline{C})^2$.

3. A NON-LINEAR PROGRAMMING MODEL

In this proposed model, there are $\left(n^2+7n+1\right)$ variables and $\left(9n+1\right)$ constraints, where *n* denotes the number of jobs. The base of this model is structured on Chou and Lee's [34], Eren and Güner [35] and Eren [36-38] integer programming models. The parameters and variables in the model are described below and then the proposed model is given.

Assumptions made in this paper are:

- ✓ Setup time is known and is included in the processing time.
- ✓ Machine preemption is not allowed, each operation, once started, must be completed before another operation may be started on the same machines.
- ✓ Machines are stable and remain available throughout the scheduling period.
- ✓ No job may be processed on more than one machine simultaneously.
- ✓ A machine may only process one job at a time.

Parameters:

j	job index		$j = 1,2,...,n.$
i	machine index		$i = 1,2.$
p_{ji}	the processing time of job j on the i^{th} machine	$i = 1,2.$	$j = 1,2,...,n.$
a	learning index		$a \leq 0$

Decision variables:

Z_{jr} If job j is scheduled at the r^{th} position to be processed, 1 otherwise 0, $r = 1,2,...,n.$ $j = 1,2,...,n.$

X_r the idle time on the second machine between the starting of the r^{th} position job and the completion of the $(r\text{-}1)^{th}$ position job, $r = 1,2,...,n.$

Y_r the time between its completion at the first machine and its begin processing at the second machine for the r^{th} position job, $r = 1,2,...,n.$

S_r the starting time for the r^{th} position job at the first machine $r = 1,2,...,n.$

Objective function:

$$\min \sum_{r=1}^{n} CD_r^2$$

Constraints:

$$\sum_{j=1}^{n} Z_{jr} = 1 \qquad r = 1,2,...,n. \qquad (1)$$

$$\sum_{r=1}^{n} Z_{jr} = 1 \qquad j = 1,2,...,n. \qquad (2)$$

$$p_{[ri]} = \sum_{j=1}^{r} Z_{jr} p_{ji} r^{a} \qquad i = 1,2. \qquad r = 1,2,...,n. \qquad (3)$$

$$S_r \geq S_{r-1} + p_{[r1]} \qquad r = 1,2,...,n. \qquad (4)$$

$$C_r = C_{r-1} + X_r + p_{[r2]} \qquad r = 1,2,...,n. \qquad (5)$$

$$X_r = S_r + p_{[r1]} - C_{r-1} \qquad r = 1,2,...,n. \qquad (6)$$

$$\overline{C} = \sum_{r=1}^{n} C_r / n \qquad r = 1,2,...,n. \qquad (7)$$

$$CD_r \geq C_r - \overline{C} \qquad r = 1,2,...,n. \qquad (8)$$

$$CD_r \geq \overline{C} - C_r \qquad r = 1,2,...,n. \qquad (9)$$

$$C_0 = S_0 = 0,$$

All variables should be greater than or equal to zero and Z_{jr} is a binary integer.

Constraint (1) specifies that only one job be scheduled at the r^{th} job priority. Constraint (2) defines that each job be scheduled only once. Constraint (3) denotes the r^{th} of i^{th} machine position jobs processing time. Constraint (4) represents that the beginning processing time of the r^{th} position ranked job be greater than or equal to the previous jobs completion time at the first machine. Constraint (5) represents that the beginning processing time of the r^{th} position job be greater than or equal to the previous jobs completion time at the machine. Constraints (6) indicates that the idle time on the second machine to process the r^{th} position job (X_r) equals the starting time for the r^{th} position job on the first machine (S_r) plus its processing time on the first machine ($p_{[r,1]}$), plus the time between its completion on the first machine and begin processing time on the second machine (Y_r) minus the completion time for the ($r-1$)th position job at the second machine (C_{r-1}). Constraint (7) represents the mean completion time. Constraint (8-9) denotes the r^{th} position completion time deviation (CD).

Illustrative example:

Ten jobs, two-machine flowshop scheduling with a learning effect example is used to illustrate a typical solution CPLEX 10 software package. Table 1 shows data for ten jobs, two-machine flowshop scheduling problem with a learning effect.

The problem is formulated as the non-linear programming model and then solved by the CPLEX 10 software package for minimizing completion time variance. Optimal sequence is 2-9-4-10-6-5-1-3-7-8, and the optimal objective value is 55562.07 s^2.

4. HEURISTIC METHODS

The considered two-machine flowshop scheduling with a learning effect can be solved optimally for small size problems with up to 30 jobs by the proposed non-linear programming model. Four heuristics methods are developed for solving large size problems. Steps of heuristic are given below.
The methodology of heuristics are similar to NEH [39] procedure. Steps of Heuristics are given below.

An improvement algorithm

Step 1. Obtain an initial sequence.

Step 2. Set $k = 2$. Pick the first two jobs from the rearranged jobs list and schedule them in order to minimization completion times variance as if there are only two jobs. Set the better one as the current solution.

Step 3. Increment k by 1. Generate k candidate sequences by inserting the first job in the remaining job list into each slot of the current solution. Among these candidates, select the best one with the least partial minimization completion times variance. Update the selected partial solution as the new current solution.

Step 4. If $k = n$, a schedule (the current solution) has been found and stop. Otherwise, go to step 3.

Table 1. Data for the example problem (s)

j	1	2	3	4	5	6	7	8	9	10
p_{j1}	72	90	9	18	27	31	21	18	2	69
p_{j2}	40	93	45	43	36	18	81	91	62	38

Heuristic 1, 2, 3 and 4 (H1, H2, H3 and H4) are obtained by using Johnson Algorithm [25], SPT (**S**hortest **P**rocessing **T**ime: $\min\{p_{j1}+p_{j2}\}$), SPT1 (**S**hortest **P**rocessing **T**ime on Machine **1**: $\min\{p_{j1}\}$) and SPT2 (**S**hortest **P**rocessing **T**ime on Machine **2**: $\min\{p_{j2}\}$) sequence respectively, in Step 1.

5. EXPERIMENTAL RESULS

In this study, all experimental tests were conducted on a personal computer with Pentium IV/2 1 GB Ram. The integer programming model is used to find the optimal solutions of the considered problem using CPLEX 10 software package. Normal processing times are drawn from the uniform distribution on the interval (1, 100). Five problem sizes are tested, namely, n=10, 15, 20, 25 and 30. For each problem size, 30 simulations are randomly generated. The learning index is taken to be $a=-0.322$ (Learning rate 80 %). The experimental set is given in Table 2. As seen from Table 2, totally 450 problems are solved.

Averagely CPU times of problem sets are shown in Figure 1. As can be seen in Figure 1 the considered problem with a learning effect can be solved up to 30 jobs by the proposed non-linear programming model. In Figure 1, the computational time grows sharply as job size *n* increases.

The optimal solutions of the considered problem can be found up to 30 machines. To solve large size problems in a short time and to find the optimal or near optimal solutions, heuristics should be used. For the problem considered, four heuristic methods were used. Heuristic error is calculated as follows.

$$\text{Error} = \frac{\text{Heuristic solution - Optimal solution}}{\text{Optimal solution}}$$

Results of the heuristic methods are compared with results of the optimal solution obtained by the integer model, and errors of these heuristics are given in Figure 2. The performance of the H1, H2, H3 and H4 heuristics are very good at all. The mean error percentages are 1.99 %, 1.61 %, 2.19 % and 1.77 % respectively.

The experimental set of large size problems is given in Table 3. As seen in Table 3, totally 630 problems are solved. Results of large size problems are also computed using heuristic methods. Since optimal solutions of these problems have not been known, results of the heuristics are compared with the best result in order to define their performances. In this comparison, the error, formulated below, is used as a performance measure:

$$\text{Error} = \frac{\text{Heuristic solution - Best heuristic solution}}{\text{Best heuristic solution}}$$

The best solutions of the considered problem can be found up to 500 jobs. Figure 3 presents solution errors the heuristics according to number of jobs. The H2 gives the best solutions of the problem. The performance of the H2 heuristic is very good at all. The mean error percentages are 2.02 %.

Table2. The parameters of problem for small size problem

Parameter	Alternatives	Values
p_{ji}	1	~U(1,100).
Number of jobs	5	10,15,20,25,30
Machines	1	2
Learning index (*a*)	3	-0.322
Number of solution problem	30	
Total problem		1×5×1×3×30=450

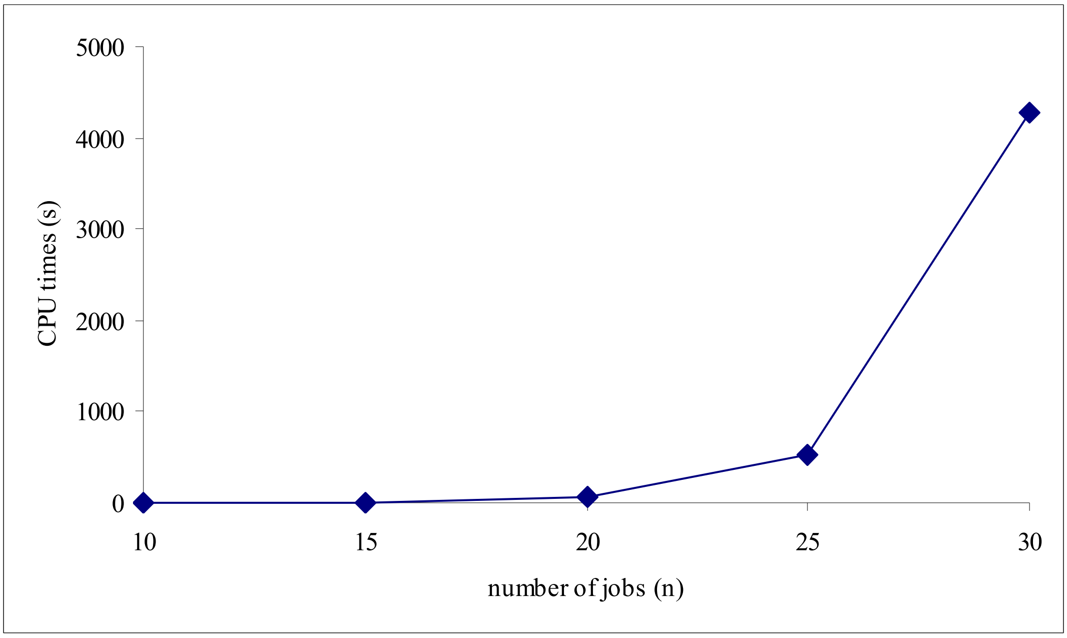

Figure 1. CPU times (seconds) of the non-linear programming model

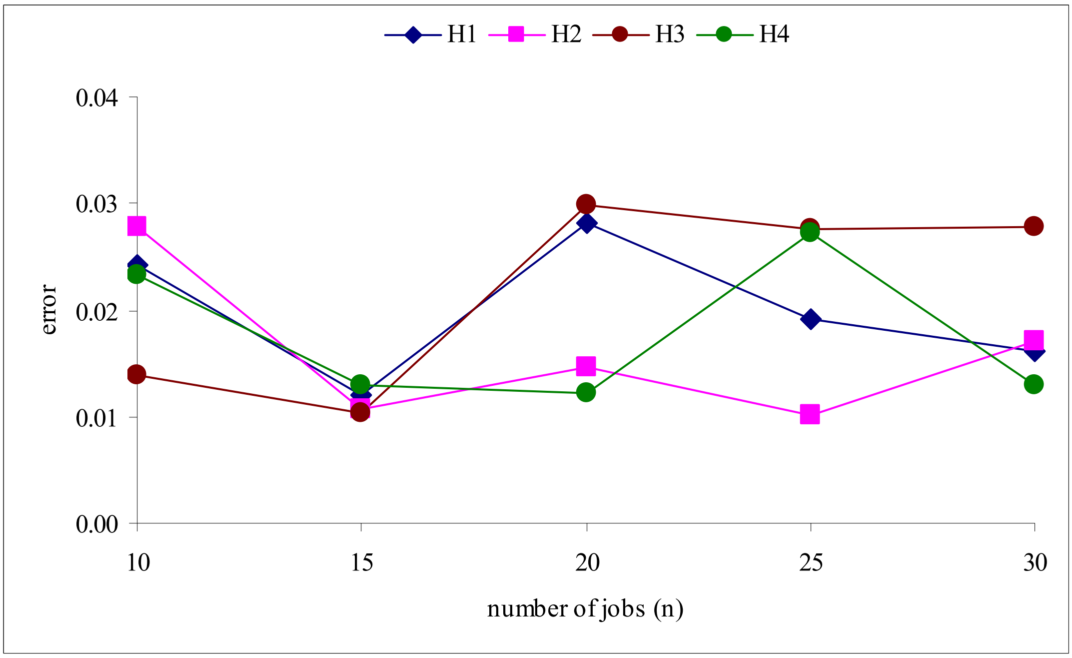

Figure 2. Heuristic errors of small size problems

Table 3. The parameters of problem for large size problem

Parameter	Alternatives	Values
p_j	1	~U(1,100).
Number of jobs	7	40,50,100,200,300,400,500
Machines	1	2
Learning index (a)	3	-0.322
Number of solution problem	30	
Total problem		1×7×1×3×30=630

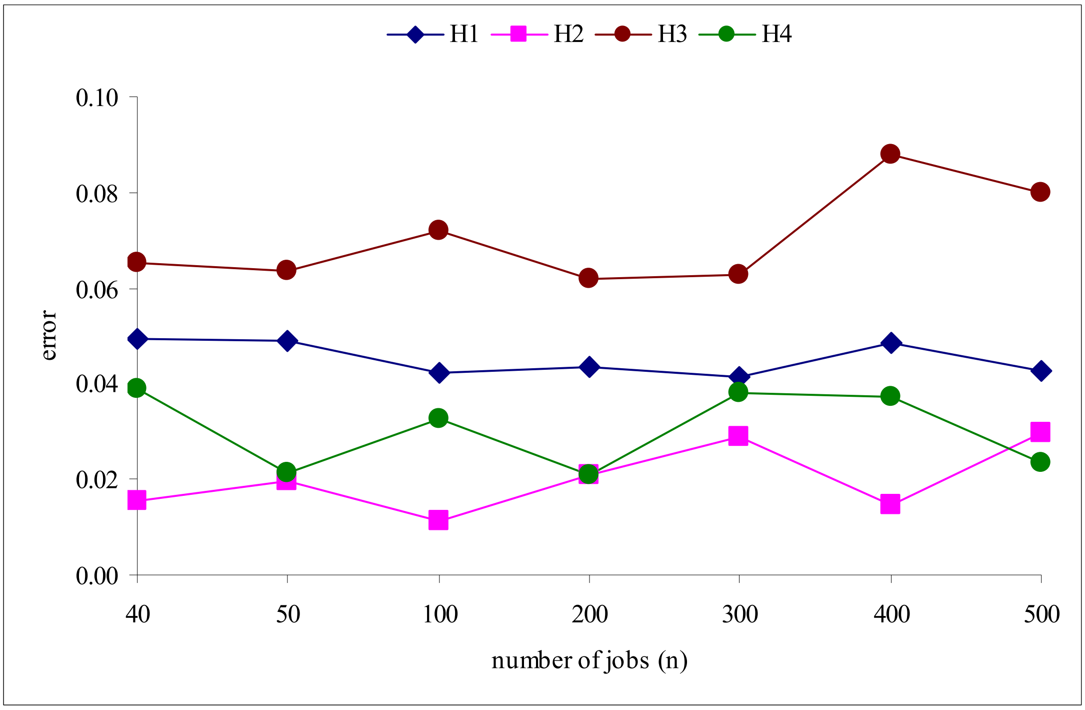

Figure 3. Heuristic errors of large size problems

6. CONCLUSIONS

In this paper, two-machine flowshop scheduling problem with a learning effect is considered. The objective function of the problem is minimization of completion time variance. A non-linear programming model is developed for the problem which belongs to NP-hard class. Results of computational tests show that the proposed model is effective in solving problems with up to 30 jobs.

To solve the large sizes problems up to 500 jobs, heuristics methods were used. The performances of heuristics about the solution error were evaluated with the non-linear programming model results for small size problems and each other for large size problems. According to results, the special heuristic for all number of jobs was the more effective than others. The overall average solution error of the heuristic algorithm is 2 %. Processing of the 30 jobs case requires only 0.1 s on average to obtain an ultimate or even optimal solution. The heuristic scheduling algorithm is more practical to solve real world applications than the non-linear programming model.

Other performance criteria with a learning effect in flowshop scheduling can be also considered for future studies.

REFERENCES

[1] Merten, A.G., Muller, M.E., "Variance minimization in single machine sequencing problems", ***Management Science,*** 18: 518–528, (1972).

[2] Viswanathkumar, G., Srinivasan, G., "A branch and bound algorithm to minimize completion time variance on a single processor", ***Computers & Operations Research,*** 30: 1135–1150, (2003).

[3] Schrage, L., "Minimizing the time-in-system variance for a finite jobset", ***Management Science,*** 21: 540–543, (1975).

[4] Hall, N.G., Kubiak, W.. "Proof of a conjecture of Schrage about the completion time variance problem", ***Operations Research Letters,*** 14: 467–472, (1991)

[5] Eilon, S., Chowdhury, I.C. "Minimizing the waiting time variance in the single machine problem", **Management Science,** 23: 567–575, (1977).

[6] Bagchi, U., Chang, Y.L., Sullivan, R.S., "Minimizing absolute and squared deviation of completion times with different earliness and tardiness penalties and a common due date", ***Naval Research Logistics***, 34: 739–751, (1987).

[7] Gupta, M.C., Gupta, Y.P., Bector, C.R., "Minimizing the flow-time variance in single-machine systems", **Journal of Operational Research Society,** 41: 767–779, (1990).

[8] Mittenthal, J., Raghavachari, M., Rana, A.I., "A hybrid simulated annealing approach for single machine scheduling problems with non-regular penalty functions", ***Computers & Operations Research,*** 20: 103–111, (1993).

[9] Gupta, M.C., Gupta, Y.P., Kumar, A., "Minimizing flowtime time variance in single machine systems using genetic algorithms", ***European Journal of Operational Research,*** 70: 289–303, (1993).

[10] Ventura, J., Weng, M.X., "Minimizing single machine completion time variance", ***Management Science,*** 41: 1448–1455, (1995).

[11] Manna, D.K., Prasad, V.R., "Bounds for the position of the smallest job in completion time variance minimization", ***European Journal of Operational Research,*** 114: 411–419, (1999).

[12] De, P., Ghosh, J.B., Wells, C.E., "On the minimization of completion time variance with a bicriteria extension", ***Operations Research***, 40: 1148–1155, (1992).

[13] Prasad, V.R., Manna, D.K., Arthanari, T.S., "A note on the lower bound for completion time variance in single machine scheduling", ***Operations Research,*** 34: 277–282, (1997).

[14] Federgruen, A., Mosheiov, G., "Heuristics for multi-machine scheduling problems with earliness and tardiness costs", ***Management Science,*** 42: 1544–1555, (1996).

[15] Gajpal, Y., Rajendran, C., "An ant-colony optimization algorithm for minimizing the completion-time variance of jobs in flowshops", ***International Journal of Production Economics,*** 101: 259–272, (2006).

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

[17] Mosheiov, G., "Scheduling problems with learning effect", ***European Journal of Operational Research,*** 132: 687–693, (2001).

[18] Mosheiov, G., "Parallel machine scheduling with learning effect", ***Journal of The Operational Research Society,*** 52: 1165-1169, (2001).

[19] Eren, T., Güner, E., "Flowshop scheduling with general job-dependent learning effect", ***K.H.O. Journal of Defense Science,*** 2: 1-11, (2003).

[20] Johnson, S.M., "Optimal two-and three-stage production schedules with setup times included", ***Naval Research Logistics,*** 1: 61–68, (1954).

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

[22] Eren, T., Güner, E., "Minimizing mean flow time in a flowshop scheduling with learning effect", ***Journal of the Faculty of Engineering and Architecture of Gazi University,*** 19: 119-124, (2004).

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

[24] Eren, T., Güner, E., "Minimizing total tardiness in a scheduling problem with a learning effect", ***Applied Mathematical Modelling,*** 31(7): 1351-1361, (2007).

[25] Eren, T., Güner, E., "A bicriteria scheduling with a learning effect: total completion time and total

tardiness", ***INFOR: Information Systems and Operational Research,*** 45(2): 75-81, (2007).

[26] Eren, T., Güner, E., "A bicriteria flowshop scheduling with a learning effect", ***Applied Mathematical Modelling,*** 32(9): 1719-1733, (2008).

[27] Xu, Z., Sun, L., Gong, J., "Worst-case analysis for flow shop scheduling with a learning effect", ***International Journal of Production Economics,*** 113: 748–753, (2008).

[28] Wang, J.-B., "Flow shop scheduling jobs with position dependent processing times", ***Journal of Applied Mathematics and Computing,*** 18: 383–391, (2005).

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

[30] Wang, J.B., Liu, L.L., "Two-machine flow shop problem with effects of deterioration and learning", ***Computers & Industrial Engineering,*** 57: 1114–1121. (2009).

[31] Cheng, T.C.E., Wu, C.C., Chen, J.C., Wu, W.H., Cheng, S.R., "Two-machine flowshop scheduling with a truncated learning function to minimize the makespan", ***International Journal of Production Economics***,141 (1): 79-86, (2013).

[32] Kuo, W.H., Hsu, C.J., Yang, D.L., "Worst-case and numerical analysis of heuristic algorithms for flowshop scheduling problems with a time-dependent learning effect", ***Information Sciences,*** 184: 282–297, (2012).

[33] Kubiak, W., "Completion time variance minimization on a single machine is difficult", ***Operations Research Letters***, 14: 49–59, (1993).

[34] Chou, F.D., Lee, C.E., "Two-machine flowshop scheduling with bicriteria problem", ***Computers & Industrial Engineering***, 36: 549–564, (1999).

[35] Eren T., Güner E., Setup times with a learning effect in flowshop scheduling problem", ***Journal of the Faculty of Engineering and Architecture of Gazi University,*** 22 (2): 353-362, (2007).

[36] Eren, T., "A note on minimizing maximum lateness in an m-machine scheduling problem with a learning effect", ***Applied Mathematics and Computation,*** 209 (2): 186-190, (2009).

[37] Eren, T., "Minimizing the total weighted completion time on a single machine scheduling with release dates and a learning effect", ***Applied Mathematics and Computation,*** 208 (2): 355-358, (2009).

[38] Eren T., "A bicriteria parallel machine scheduling with a learning effect of setup and removal times" ***Applied Mathematical Modelling,*** 33 (2): 1141-1150, (2009).

[39] Nawaz, M., Enscore, E.E., Ham, I., "A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem", ***Omega,*** 11: 91–95, (1983).