Bi-criteria SDST Flow Shop Scheduling using Simulated Annealing

Naveen Hooda[1]*, Ashwani K. Dhingra[1]

1Department of Mechanical Engineering, University Institute of Engineering & Technology Maharshi Dayanand University Rohtak 124 001 Haryana INDIA

* Corresponding author. Tel: +** (0)01262-393291,

E-mail: naveen_huda@yahoo.com

Abstract

The flow shop scheduling problem (FSSP) with sequence dependent setup times (SDST) is one of the most complex class of scheduling problems. Efficient supervision of sequence dependent setup times (SDSTs) is one of the significant features to enhance the performance of manufacturing system. In the present work, two different SAs named SA (NEH) and SA (NEH_EDD) have been proposed for bi-criteria SDST flow shop scheduling problem. The performance among both heuristics based simulated annealing approaches have been compared with the help of a defined performance index known as Average Relative percentage deviation (ARPD) up to 200 jobs and 20 machines benchmark problems of Talliard (1993). The objective function considers the minimizing makespan and number of tardy jobs. From the comparison, it has been found that performance of proposed SAs vary with the size of job and machines as SA(NEH_EDD) upto 100 jobs for 5 and 10 machine problems and further increase in job size and machine size, SA(NEH) shows better results.

***Keywords**: Flow shop scheduling, Sequence dependent set up time, Hybrid Simulated Annealing, makespan, number of tardy jobs.*

1. Introduction

Scheduling is concerned with allocating limited resources over tasks to optimize certain objective functions. In scheduling, the nature and quantity of each resource should be well-known so that accomplishment of different tasks can be possibly determined. In flow shop production (FSP) system, jobs flow from an initial machine, through several intermediate machines, and at last to a final machine before completion. The flow of work is always unidirectional in a flow shop. If flow shop problem is with jobs having sequence dependent setup times on machines then it would definitely be a more alluring and realistic variation (R'ios-mercado et. al., 1999; Cheng T.C.E. et. al., 2000; Ruiz R. et. al., 2005; Allahverdi A. et. al., 2008). Setup times are defined to be the work to arrange the resource (machine), process, or tasks (products). Sequence-dependent setup times are typically found in the conditions where the facility is a flexible machine. Setup times involve operations that have to be performed on machines and that are not part of the processing times. This includes cleaning, fixing or releasing parts to machines, adjustments to machines, setting the required jigs and fixtures, and inspecting material etc. The awareness in scheduling problems that considers set up times as separate began in mid-1960s. Until now only hypothetical analysis has been done because of the complexity of SDST problems and there exists neither a universally acknowledged method nor an analytical algorithm for solving such problems. Lockett and Muhlemann (1972) proposed a branch and bound algorithm for the scheduling of the jobs with SDSTs on a single machine to minimize the total number of tool changes. Scheduling problems involving SDSTs are strongly NP-hard (Baker, 1974). In an another earlier work Rinnooy Kan et. Al. (1975) presented a branch and bound algorithm for sequencing jobs on a single processor to minimize the weighted sum of tardiness of the jobs. Srikar, B. N. and Ghosh, S. (1986) proposed a mix integrated linear programme (MILP) to minimize an n jobs and m machines flow shop with SDSTs. Their formulation required a less number of integer binary variables than other formulations. Parthasarathy S. and Rajendran C. (1997) proposed a heuristic algorithm to minimize mean weighted tardiness in a flow shop with sequence dependent set up times. Tan K. and Narasimhan R. (1997) proposed an algorithms based on subsets of those problems used by Ragatz & Rubin (1995) and Ragatz (1993). The algorithm is invaluable for 'on-line' production scheduling and 'last-minute' changes to production schedule. Some earlier research papers have reviewed research associated to setup times e.g. (Allahverdi et al. 1999; Yang and Liao, 1999 and Zhu X. et al., 2006). Allahverdi et al. (1999) and Zhu X. et. al. (2006) particularly mentioned nearly 200 and 100 references that deal with setup issues. Cheng T. C. E. et al. (2000) reviewed research on flow shop scheduling problems with setup times. They presented a complexity hierarchy and classified research into four categories that involved sequence independence and dependence relative to both job and family setup times. Zhao C. et. al. (2010) proposed polynomial time algorithms to solve single machine scheduling problems with setup times and deteriorating jobs.

One of the most complications related to flow shop scheduling environment is considering sequence dependent set up times (SDSTs) separately than processing time. A large amount of the work in flow shop scheduling problems considered SDST included in the processing time. But for advance manufacturing system performance, a separate managing of SDST has to be considered. Also, very limited study on flow shop scheduling under SDST environment for due date related performance measure have been done so far. Hence both these situations have attracted increased concentration from the managers and researchers.

The scheduling literature reveals that research on bi-criteria is also focused by many researchers. Chakravarthy K. and Rajendran C. (1999) developed a heuristic for scheduling in a flow shop with the objective of minimizing the makespan and maximum tardiness of a job. Eren T. (2010) developed an integer programming model for the problem which belongs to NP-hard class and the objective function of the problem is minimization of the weighted sum of total completion time and makespan. Results of computational tests show that the proposed model is effective in solving problems with up to 18 jobs and 6 machines. Gholami M. et al. (2009) describes how we can incorporate simulation into genetic algorithm approach to the scheduling of a SDST hybrid flow shop with machines that suffer stochastic breakdown. Mansouri S. A. (2006) proposed a multi-objective simulated annealing (MOSA) solution approach to a bi-criteria sequencing problem to coordinate required set-ups between two successive stages of a supply chain in a flow shop pattern. The two objectives considered are minimizing total set-ups and minimizing the maximum number of set-ups between the two stages that are both NP-hard problems. Safari E. and Sadjadi S. J. (2011) proposed a hybrid method for flowshops scheduling with condition-based maintenance constraint and machines breakdown. The proposed algorithm is designed for non-resumable flowshop state where the processing of jobs after preventive maintenance is restarted from the beginning. Naderi B. et. al. (2008) introduced a novel simulated annealing (SA) with a new concept, called "Migration mechanism", and a new operator, called "Giant leap", to strengthen the competitive performance of SA through striking a compromise between the lengths of neighborhood search structures. Fakhrzad M.B. and Heydari M. (2008) presented an efficient solution approach for hybrid flow-shop scheduling problem with identical parallel machines at each stage (machine center) for minimizing the sum of the earliness and tardiness costs. Riezebos J. et. al. (1995) proposed some heuristic procedures for a special type of flow shop which consider time lags between multiple operations of a job. Alcaide D. (2002) proposed a general procedure that tries to solve the minimum expected makespan flow shop problem subjected to breakdowns. Random variables have been considered, which measure the length of availability periods and repair times, to study availability intervals of machines. Partial feasible schedules have been proposed in these intervals and combine them to offer a final global solution to optimize the expected makespan.

The makespan criteria is very fundamental as well as crucial in flow shop scheduling in order to strengthen the productivity and maximum deployment of assets. Another decisive feature towards fulfilling consumer pleasure is on-time delivery. Also a special attention in the recent years is acknowledged by the problems with due date related objectives due to the introduction of new techniques of inventory management such as just-in-time (JIT) concepts. Jobs are to be completed neither too early nor too late in JIT systems which leads to the scheduling problems with both earliness and tardiness costs. Hence in practical circumstances of today's world, current work deals with bi-criteria flow shop scheduling under SDST with minimization of makespan and number of tardy jobs simultaneously. Equal weight age is provided to both the objectives in the said work and different combinations of machines & jobs are taken to analyze the results to a best level. In the past a number of heuristics have been proposed and various researchers have modified the original NEH (1983) to solve the NP hard problems and to get the optimum results. Low C. et. al. (2004), Chakravarthy K. & Rajendran C. (1999), Nearchou A. C. (2004) and Laha D. & Chakraborty U. K. (2008) have modified the original NEH (1983) and have got better results. Thus, in the present work, two modified heuristics have been proposed for initial feasible sequence. Sequence obtained from the modified heuristics is combined with initial seed sequence of simulated annealing and called as Hybrid Simulated Annealing (HSA). Hence two different SAs named SA (NEH) and SA (NEH_EDD) have been proposed for bi-criteria SDST flow shop scheduling problem. The performance among both heuristics based simulated annealing approaches have been compared with the help of a defined performance index up to 200 jobs and 20 machines benchmark problems of Talliard (1993).

2. Problem Formulation

$n \times m$ SDST flow shop scheduling problems have been considered in the present study. The various assumptions, parameters and fitness function have been described below.

2.1 Assumptions

- All jobs are available for processing at time zero.
- Each job must be completed if it is started.
- Each and every processing time of each job on every machine must be known in advance; whatever may be the sequence of the jobs to be processed.
- Processing times are independent of the schedule.
- Machines may be idle or at rest.
- If the next machine on the sequence needed by a job is not available, the job can wait and joins the queue at that machine i.e. in-process inventory is allowed.
- Setup times are known and they are not included in processing times.
- Machines never breakdown and are available throughout the scheduling period.

- No machine can process more than one operation at the same time.

2.2 Parameters

Table 1: Parameters for proposed SAs

Parameter	Value
Initial Temperature	n
Annealing Function	Randomly swapping
Temperature Function	Exponentially
Acceptance Function	Boltzmann probability
Re-anneal interval	*nx8*
Stopping Criteria (time in seconds)	*n xmx0.5*

2.3 Fitness function

Bi-criteria fitness function considered the minimizing the weighted sum of makespan and number of tardy jobs which has been described below:

First performance measures for scheduling is makespan (C_{max}), which has been used for maximum deployment of resources to enhance productivity and acknowledged as maximum completion time of last job to way out from the system.

$$C_{max} = \text{Max}\,(C_1 \ldots C_n)$$

The second criterion for scheduling is the minimizing the number of tardy jobs. Associated with each job *j* is a due date $d_j > 0$. Let $U_j = 1$ if due date for job j is smaller than the completion time C_j of job j, otherwise $U_j = 0$. The total number of tardy jobs (N_t) is defined as :

$$N_t = \sum_{j=1}^{n} U_j$$

Therefore, bi-criteria fitness function is obtained by combining both objectives into single scalar function which has been framed as:

$$\text{Min}\,[\alpha\, C_{max} + \beta\, (N_t)]$$

Where α and β are the weight values for the considered objective functions having constraints:

$$\alpha \geq 0 \quad \text{and} \quad \beta \geq 0$$

$$\alpha + \beta = 1$$

Only one set of equal weighted values have been considered to give relative significance to both the objective functions. All experiment tests are conducted on benchmarks problems of Taillard (1993) with stopping time of the algorithm has been fixed to computational time limit based criteria for fair comparison upto 200 jobs and 20 machines.

3. Simulated Annealing

3.1 Proposed Hybrid Simulated Annealing (SA)

Proposed SA in this study has been explained as follows:

Step 1: *Initialization:* The algorithm begins with initial seed sequence obtained from table 2 and described in section 3.1 and allots the seed sequence to {S} and best. Get the bi-criteria SDST flow shop function for {S} and initialize the initial temperature.

Step 2: *Modify the schedule:* The annealing function will now modify this schedule and return to a new schedule that has been changed by an amount proportional to the temperature.

Step 3: *Access the solution:* The algorithm determines whether the new schedule{S'} is superior or inferior than the current{S}. If {S'} is superior than the {S}, it becomes the next schedule{S=S'}. If the {S'} is inferior than the {S}, the algorithm may still make it the next schedule based on an acceptance probability.

Step 4: *Adjust temperature:* The algorithm lowers the temperature proportional to cooling rate and store the best point found so far.

Step 5: *Re-annealing:* Re-annealing is carried out after a certain number of schedules are accepted by SA. Re-annealing raises the temperature and the search is resumed with the new temperature values.

Step 6: The algorithm stops when maximum time limit exceeds $n \times m \times 0.5$ *seconds*.

3.2 Proposed Heuristics

In the present work, a modified NEH has been proposed for initial seed sequence in simulated annealing for comparative analysis with original NEH (Nawaz et. al., 1983). A NEH based heuristic of Nawaz et. al. (1983) for makespan minimization has been modified for bi-criteria fitness function of minimizing makespan and number of tardy jobs for SDST flow shop scheduling. The description of proposed modified NEH is as follows:

Step 1: Generate the initial sequence.

Step 2: Fix k = 2. Select the earliest two jobs from the rearranged jobs list and schedule them in order to minimizing weighted sum of makespan and number of tardy jobs as if there are only two jobs. Set the better one as the current solution.

Step 3: Increase k by 1. Create k candidate sequences by putting the first job in the left over job list into each slot of the current solution. Among these candidates, select the best one with the least partial minimization of weighted sum of makespan and number of tardy jobs. Update the selected partial solution as the new current solution.

Step 4: If k = n, a schedule (the current solution) has been found and stops. If not, move to step 3.

Initial sequence in step 1 of proposed heuristic is being generated by the following rules as described below:-

1. Schedule the jobs initially in descending order of $\sum_{i=1}^{m} P_{ij}$ as in original NEH (Nawaz et. al., 1983).
2. Organize the jobs according to the earliest due date of jobs i.e. EDD (Kim, 1993).

Thus the final sequence has been obtained by considering the initial sequence as above in step 1 of proposed heuristic and designated as NEH_EDD.Thus a modified heuristics has been developed and final sequence from these heuristic has been taken as initial sequence in the SA procedure resulting into two modified heuristics based Simulated Annealing (SA).

Table 2: Different heuristics for comparative analysis of proposed SAs

S. No.	Heuristic	Description
1	NEH	Initial sequence is developed as proposed by Nawaz et al. (1983).
2	NEH_EDD	Initially jobs are ordered according to least value of due date of jobs, then final sequence is obtained by NEH procedure as in section 3.2

4. Results and Discussion

In the present work, all the experimental tests have been conducted on a personal computer having core 2 duo processor with 2 GB RAM. Stopping limit of both SAs has been fixed to computational time limit of $n \times m \times 0.5$ sec*onds***.** The relative percentage deviation (*RPD*) (Naderi et. al., 2009) has been used to compare all the algorithms which have been calculated as:-

$$\text{Relative Percentage Deviation (RPD)} = \frac{(Mean_{sol} - Best_{sol})}{Best_{sol}} \times 100$$

Where $Mean_{sol}$ is the solution obtained by Algorithm and $Best_{sol}$ is the best (least) solution obtained among both the SAs for particular problem. $Best_{sol}$ can be found between the results obtained by running particular algorithm five times for a particular problem and mean solution is the final average solution given by the algorithm for all the five runs. RPD nearer to zero gives the best results. A total of 110 benchmark problems proposed by Taillard (1993) have been solved for analyzing the results, which includes different combinations of 20, 50, 100 and 200 jobs being processed on 5, 10 and 20 machines. Each combination of different machines and jobs has 10 different set of problems i.e. a particular combination suppose 20 jobs and 5 machines has 10 different set of such problems. After finding the RPD for 10 different set of problems of a particular combination their mean is also calculated called average relative percentage deviation (ARPD). In the present work both the performance measures i.e. makespan and numbers of tardy jobs have been given equal importance in the proposed fitness function.

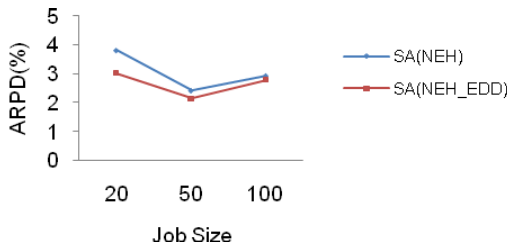

Fig. 1 ARPD for both SAs for 5 machine problems

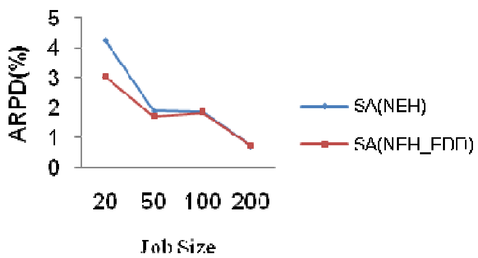

Fig. 2 ARPD for both SAs for 10 machine problems

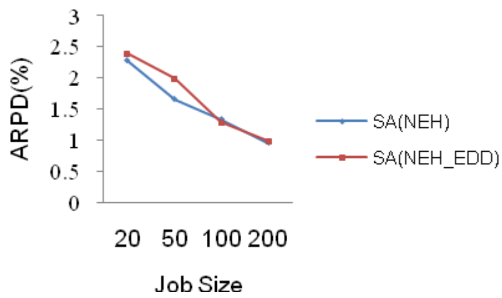

Fig.3 ARPD for both SAs for 20 machine problems

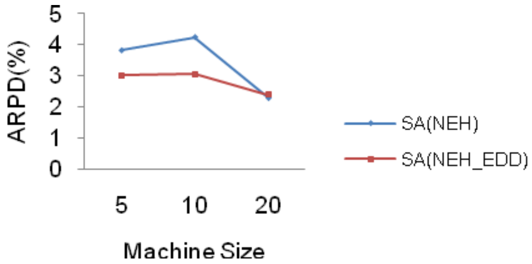

Fig. 4 ARPD for both SAs for 20 job problems

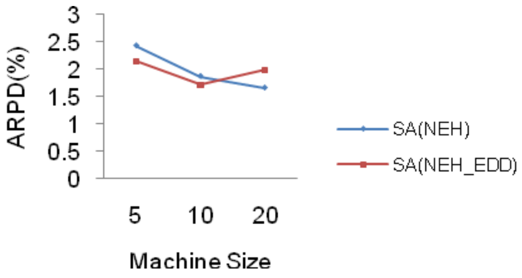

Fig. 5 ARPD for both SAs for 50 job problems

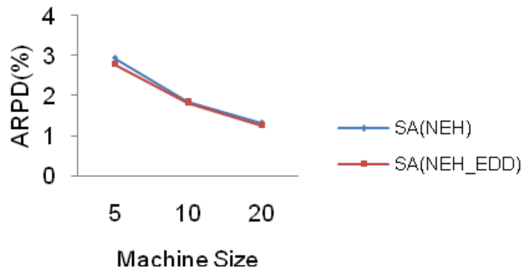

Fig. 6 ARPD for both SAs for 100 job problems

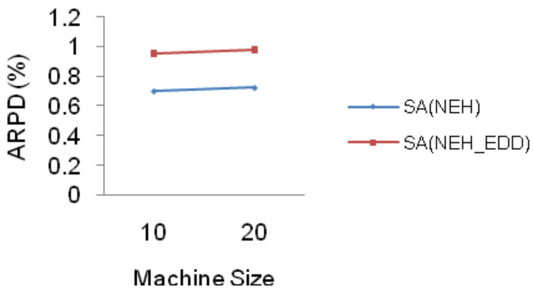

Fig. 7 ARPD for both SAs for 200 job problems

Fig. 1 to Fig. 3 shows ARPD for 5, 10 and 20 machine problems. The results have also been compiled in tabular form as shown in table 3 & 4. It can be seen from table 3 that for 5 machines, SA (NEH_EDD) presents improved results for 20, 50 and 100 job problems over SA (NEH) with ARPD of 3.02%, 2.15% and 2.79% respectively.
For 10 machines, SA (NEH_EDD) with ARPD of 3.04%, 1.72% and 1.84% shows better results over SA (NEH) for 20, 50 and 100 job problems respectively. For 200 job problem, SA (NEH) with ARPD of 0.70% shows improved performance. Also for 20 machines, SA(NEH) presents better results over SA(NEH_EDD) for 20, 50 and 200 job problems with ARPD of 2.28%, 1.66% and 0.95% respectively. SA (NEH_EDD) with ARPD of 1.27% shows improved results.

Table 3 ARPD for different SAs for different machine problems

Machines	Jobs	SA (NEH)	SA (NEH_EDD)
5	20	3.81	**3.02**
	50	2.43	**2.15**
	100	2.94	**2.79**
10	20	4.23	**3.04**
	50	1.87	**1.72**
	100	1.85	**1.84**
	200	**0.70**	0.72
20	20	**2.28**	2.38
	50	**1.66**	1.98
	100	1.33	**1.27**
	200	**0.95**	0.98

Hence from the analysis it has been concluded that performance of proposed SAs vary with the size of machines and jobs as SA(NEH_EDD) for 5 machines shows better performance with 20 and 50 job problems over SA(NEH) and also for 10 machines with job size of 20, 50 and 100. Whether SA(NEH) shows better performance over SA(NEH_EDD) for 20 machines with 20, 50 and 200 job problems.
Fig. 4 to Fig. 7 shows ARPD for 20, 50, 100 and 200 job problems. It can be seen from table 4 that for 20 jobs, SA(NEH_EDD) presents better results for 5 and 10 machine problems with ARPD of 3.02% and 3.04% respectively. For 20 machine problem SA (NEH) presents improved results with ARPD of 2.28%.Also for 50 jobs, SA (NEH_EDD) presents better results with ARPD of 2.15% and 1.72% for 5 and 10 machine problems respectively. Whether for 10 machines SA (NEH) shows improved results with ARPD of 1.66%. And for 100 jobs also, SA (NEH_EDD) shows improved results over SA (NEH) with ARPD of 2.79%, 1.84% and 1.27% for 5, 10 and 20 machine problems respectively. Also for 200 jobs, SA (NEH) with ARPD of 0.70% and 0.95% for 10 and 20 machine problems respectively presents improved results over SA (NEH_EDD) for any machine size.

Table 4 ARPD for different SAs for different job problems

Jobs	Machines	SA (NEH)	SA (NEH_EDD)
20	5	3.81	**3.02**
	10	4.23	**3.04**
	20	**2.28**	2.38
50	5	2.43	**2.15**
	10	1.87	**1.72**
	20	**1.66**	1.98
100	5	2.94	**2.79**
	10	1.85	**1.84**
	20	1.33	**1.27**
200	10	**0.70**	0.72
	20	**0.95**	0.98

5. Conclusions

Bi-criteria fitness function proposed for scheduling the jobs in SDST flow shop proves to be an efficient and broad measure, in today's competitive and fast growing environment. From comparative analysis among SAs, it has been concluded that their performance varies with size of job and machines. SA (NEH_EDD) upto 100 jobs for 5 and 10 machine problems and SA (NEH) for 200 jobs and 20 machine problems shows better performance.

References

[1] A. Allahverdi, C. T. Ng, T. C. E. Cheng, and M. Kovalyov, "A survey of scheduling problem with setup times or costs", European Journal of Operational Research, Vol. 187, No. 3, 2008, pp. 985-1032.
[2] A. Allahverdi, J. N. D. Gupta, and T. Aldowaisan, "A review of scheduling research involving setup considerations", Omega the International Journal of Management Sciences, Vol. 27, No. 2, 1999, pp. 219–239.
[3] A. C. Nearchou, "A novel metaheuristic approach for the flow shop scheduling problem", Engineering Applications of Artificial Intelligence, Vol. 17, No. 3, 2004, pp. 289–300.
[4] A. G. Lockett, and A. P. Muhlemann, "Technical notes: a scheduling problem involving sequence dependent changeover times", Operation research, Vol. 20, No. 5, 1972, pp. 895-920.
[5] A. H. G. Rinnooy Kan, B. J. Lageweg, and J. K. Lenstra, "Minimizing total costs in one machine scheduling", Operations Researches, Vol. 23, No. 5, 1975, pp. 908-927.
[6] B. N. Srikar, and S. Ghosh, "A MILP model for the n-job,M-stage flow shop with sequence dependent set up times", International Journal of Production Research, Vol. 24, No. 6, 1986, pp. 1459-1474.
[7] B. Naderi, M. Zandieh, A. K. G. Balagh, and V. Roshanaei, "An improved simulated annealing for hybrid flow shops with sequence-dependent setup and transportation times to minimize total completion time and total tardiness", Expert Systems with Application, Vol. 36, No. 6, 2009, pp. 9625-9633.
[8] B. Naderi, M. Zandieh, and V. Roshanaei, "Scheduling hybrid flow shops with sequence dependent setup times to minimize makespan and maximum tardiness", International Journal of Advance Manufacturing Technology, Vol. 41, No.11-12, 2008, pp. 1186–1198.
[9] C. Low, Y. J. Yeh, and K. I. Huang, "A robust simulated annealing heuristic for flow shop scheduling problems", International Journal of Advance Manufacturing Technology, Vol. 23, No. 9-10, 2004, pp. 762–767.
[10] C. Zhao, and H. Tang, "Single machine scheduling with past-sequence-dependent setup times and deteriorating jobs", Computers & Industrial Engineering, Vol. 59, No. 4, 2010, pp. 663–666.
[11] D. Alcaide, A. G. Rodriguez, and J. Sicilia, "An approach to solve the minimum expected makespan flow-shop problem subject to breakdowns", European Journal of Operational Research, Vol. 140, No. 2, 2002, pp. 384–398.
[12] D. Laha, U. K. Chakraborty, "An efficient hybrid heuristic for makespan minimization in permutation flow shop scheduling", International Journal of Advance Manufacturing Technology, Vol. 44, No.5, 2008, pp. 559-569.
[13] E. Safari, and S. J. Sadjadi, "A hybrid method for flow shops scheduling with condition-based maintenance constraint and machines breakdown", Expert Systems with Applications, Vol. 38, No. 3, 2010, pp. 2020–2029.
[14] E. Taillard, "Benchmarks of basic scheduling problems", European Journal of Operation Research, Vol. 64, No. 2, 1993, pp. 278–285.
[15] G. L. Ragatz, "A branch-and-bound method for minimum tardiness sequencing on single processor with sequence-dependent setup times", In Proceedings: Twenty-fourth Annual Meeting Of the Decision Sciences Institute, 1993, pp. 1375-1377.
[16] J. Riezebos, G. J. C. Gaalman, and J. N. D. Gupta, "Flow shop scheduling with multiple operations and time lags", Journal of Intelligent Manufacturing, Vol. 6, 1995, pp. 105-115.
[17] K. C. Tan, and R. Narasimhan, "Minimizing tardiness on a single processor with sequence-dependent setup times: a simulated annealing approach", Omega, International journal of management and science, Vol. 25, No. 6, 1997, pp. 619-634.
[18] K. Chakravarthy, and C. Rajendran, "A heuristic for scheduling in a flow shop with the bi-criteria of makespan and maximum tardiness minimization", Production Planning and Control, Vol. 10, No. 7, 1999, pp. 707–714.
[19] K. R. Baker, D. Trietsch, "Introduction to sequencing and scheduling", New York, Wiley, 2009.
[20] M. B. Fakhrzad, and M. Heydari, "A heuristic algorithm for hybrid flow shop production scheduling for minimizing the sum earliness & tardiness costs", Journal of the Chinese Institute of Industrial Engineers, Vol. 25, No. 2, 2008, pp. 105-115.
[21] M. Gholami, M. Zandieh, and A. Alem-Tabriz, "Scheduling hybrid flow shop with sequence- dependent setup times and machines with random breakdowns", International Journal of Advance Manufacturing Technology, Vol. 42, No. 1-2, 2009, pp. 189-201.

[22] M. Nawaz, E. Enscore, and I. Ham, "A heuristic for the m-machine, n-job flow shop sequencing problem", Omega, Vol. 11, No. 1, 1983, pp. 91–95.
[23] P. A. Rubin, and G. L. Ragatz, "Scheduling in a sequence-dependent setup environment with genetic search", Computers and Operation Research, Vol. 22,No.1, 1995, pp. 85-99.
[24] R. Ruiz, and T. Stutzle, "An iterated greedy algorithm for the flow shop problem with sequence dependent setup times" The 6th Metaheuristics International Conference Vienna, Austria, 2005, pp. 817-823.
[25] R. Z. R´ios-mercado, and J. F. Bard, "An enhanced tsp-based heuristic for makespan minimization in a flow shop with setup times", Journal of Heuristics, Vol. 5, No. 1, 1999, pp. 53–70.
[26] S. A. Mansouri, "A simulated annealing approach to a bi-criteria sequencing problem in a two stage supply chain", Computers and Industrial Engineers, Vol. 50, No. 1-2, 2006, pp. 105-119.
[27] S. Parathasarathy, and C. Rajendran, "A simulated annealing heuristic for scheduling to minimize mean weighted tardiness in a flow shop with sequence dependent setup times of jobs - a case study", Production Planning and Control, Vol. 8, No. 5, 1997, pp. 475-483.
[28] T. C. E. Cheng, J. N. D. Gupta, and G. Wang, "A review of flow shop scheduling research with setup times", Production and Operations Management, Vol. 9, No. 3, 2000, pp. 262-282.
[29] T. Eren, "A bi-criteria M-machine flow shop scheduling with sequence-dependent setup times", Applied Mathematical Modeling, Vol. 34, No. 2, 2010, pp. 284-293.
[30] W. H. Yang, "Survey of scheduling research involving setup times", International Journal of Systems Science, Vol. 30, No. 2, 1999, pp. 143–155.
[31] X. Zhu, and W. E. Wilhelm, "Scheduling and lot sizing with sequence-dependent setup: A literature review", IIE Transactions, Vol. 38, No.11, 2006, pp. 987–1007.