CRITICAL STUDY ON MINIMIZATION OF RENTAL COST IN FLOW SHOP UNDER FUZZY ENVIRONMENT

Soumen Chowdhury, Dr. Annapurna Ramakrishna Sinde
Department of Mathematics, Dr. A.P.J. Abdul Kalam University, Indore (M.P.), India

ABSTRACT :
Each work must be processed on a set of machines in a specific order according to the flow shop scheduling concept. The purpose is to establish the work hierarchy that will best optimise a particular predetermined objective function. Each machine can only handle one job at a time, and only one machine can handle any particular job. This paper reflects critical study on minimization of rental cost in flow shop under fuzzy environment.
KEYWORDS : flow shop , scheduling , approaches , environment
DOINumber:10.48047/nq.2022.20.22.NQ10265 **NeuroQuantology2022;20(22):2733-2741**

INTRODUCTION

Flow shops are a common difficulty in scheduling in real-world applications. Most flow shop reports assume that the time it takes to complete each task on each machine is a precise measurement. However, in practice, it is difficult to accurately estimate the time it takes to complete each work because of the ambiguity of information and the variety of management scenarios. As a result, the conventional approaches, both deterministic and random, tend to be less successful in portraying the ambiguity or imprecision of language values. Unquantifiable data, poor data, incomplete data, and unreachable data are all examples of imprecision, according to Ribeiro's definition of it. Fuzzy set theory may be used as an alternate solution to this sort of processing time issue. This chapter focuses on fuzzy set theory, which was first developed by Zadeh.

The order of fuzzy subsets of the unit interval was later introduced by Yager. McCahon and Lee devised an effective method for reducing the makespan for the F2/perm/Cmax issue by using triangular fuzzy numbers to represent the processing time. In order to defuzzify the triangular fuzzy integers, they employed generalized mean values (GMVs). Because GMVs can't capture real-world uncertainty, Sanuja and Xueyan looked at the flow shop model for n tasks on two machines to see how much time might be saved, and they found that their solution outperformed McCahon and Lee. Fuzzy flow shop model with fuzzy process time was developed by Ishibuchi and Lee, using the average high ranking (AHR) idea to solve the issue. There is a flow shop model with fuzzy due dates and fuzzy process times established by Ishibuchi and Murata. Gupta et al. explored the F2/perm/Cmax issue with a single transport facility under a fuzzy environment. There has been a long history of research on the flow shop scheduling issue, which was first proposed by Johnson in 1954. When more than two machines are involved, the task becomes much more difficult.

To get a precise answer, we'll have to use implicit enumeration operations like 'branch and bound.' These NP-hard problems proved Branch and Bound algorithms to be among the most intensive ways for solving them. Branching, Bounding, and Search procedures are all shown in the following key three components. There has been a lot of focus on strengthening lower limits in the branch and bound techniques for flow shop issues that have been previously published. In the literature, a number of scholars explored flow shop issues using the branch-and-bound technique.

With Ignall and Scharge, a more efficient flow sequence was achieved by using the branch and bind approach. The flow shop models were also solved using the branch and bind approach by Lomnicki. To that end, consider criteria such as transit time, the weight of tasks, and break-down intervalence when promoting work in this direction (Chandrasekharan, Brown and Lomnicki, Gupta et al). This chapter explains how to solve the n-job, 3-machine flow shop issue in a fuzzy environment using a branch and bound strategy. The study's goal is to find the most efficient or near-optimal timetable for tasks while keeping the entire rental cost of hired machinery in mind. To de fuzzily the supplied issues, the chapter use a widely used defuzzification approach known as Yager's first index.

BASIC CONCEPTS OF FUZZY SET THEORY

The following are a few examples of the numerous fuzzy notions we utilized in our research:

1. FUZZY WORKING TIME

Depending on how you look at it, a job's working hours might vary greatly. It's possible that this is due to a frequent occurrence or to the unique nature of the workplaces in question. When a contractor signs a contract, we've seen that he usually remembers the total amount of materials he's used. Due to various factors, such as the inability to go to work, poor working conditions, or other unusual circumstances, the cost may vary. In the future, owing to these reasons, work may be done late, which causes the order to not be delivered on time; and if the work is completed early, the inventory problem may arise.

2. FUZZY SET

Let X is a nonempty set and x E X be an element of X, then a Fuzzy set Pin X is defined by a set of ordered pairs

$\check{F} = \{x, \pi_{\tilde{F}}(x): x \in X\}$

Where$\mu_{\tilde{F}}(X)$ is called the membership function or grade of membership of x in P which maps X to the membership space N which is considered as the closed interval. Note: When N contains only two points 0 and I, then Fuzzy sett transforms to a crisp set (non- fuzzy set) and membership function $\mu_{\tilde{F}}(X)$ reduces to the characteristic function of the membership function

3. FUZZY NUMBER

A fuzzy subset P in91 (real line) is called a fuzzy number if it must possess the following three properties:

(i) Convexity Property:

A fuzzy subset is said to be convex if $\tilde{F}(ax_l + (1-a)x_2$ $\geq min\{\tilde{F}(x_1),\}\tilde{F}(x_2)$ For all $x_1, x_2 \in \Re$ and $a \epsilon [0,1]$ I.e., convex property states that the line formed by a - cut is always continuous.

(ii) Normality Property: A fuzzy subset is said to be normal if $\exists$ $X_0 \in \Re$ such that $\check{F}(X_0) = 1$ i.e. normalization property states that maximum value of membership is one.

(iii) Pies piecewise continuous.

In another way, we can define: A fuzzy number is normalized convex fuzzy set defined on the real line 91 whose membership function is piecewise continuous.

4. TRIANGULAR FUZZY NUMBER (TFN)

A fuzzy number $\check{F} = (u, v, w), u < v <$ won a set of real numbers is said to be a TFN number if its membership function $\mu_{\tilde{F}}: \Re \rightarrow [0,1]$ has the following characteristics:

I. If the mapping $\mu_{\tilde{F}}: \Re \rightarrow [0,1]$ is a continuous.

2. $\mu_{\tilde{F}}(x) = 0 \forall x \in (-\infty, u] U(w, \infty]$.

3. $\mu_{\tilde{F}}(x)$ is strictly increasing and continuous on [u, v].

4. $\mu_{\tilde{F}}(x) = 1 \forall x = v$

5. $\mu_{\tilde{F}}(x)$ is strictly decreasing and continuous on [v, w]

$$\mu_{\tilde{F}}(x) = \begin{cases} 0: x \leq u, \\ \frac{x-u}{v-u}: u \leq x < v, \\ 1: x = v, \\ \frac{w-x}{w-v}: v < x \leq w, \\ 0; x \geq w \end{cases}$$

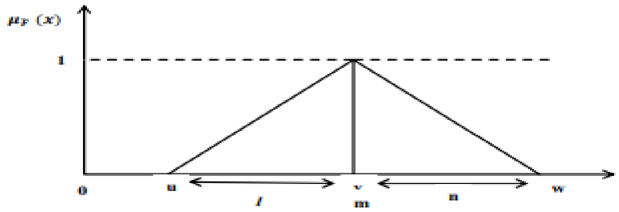

Figure 1. triangular fizzy number F̃=(u,v,w)=(l,m,n)

$l = (v, u)$Is called as the left spread.

$m = v$ Is known as the modal value

n = (w-v) is called as the right spread of the triangular fuzzy number $\tilde{F} = (u, v, w)$

Hence TFN number $\tilde{F} = (u, v, w)$can be denoted by the triplet $\tilde{F} = (l, m, n)$

5. AVERAGE HIGH RANKING <AHR> 2735

The crisp value of the given TFN number $\tilde{F} = (\alpha, \beta, \gamma)$is called as Average High Ranking (AHR) given by Yarer where an in favorable condition, fl in normal (mid value) condition and y is in worst (bad) condition, calculated by the formula defined as:

$$AHR\,(\widetilde{F}) = \frac{3\beta = \gamma - A}{3}$$

6. VARIOUS FUZZY ARITHMETIC OPERATIONS

Let $(T_{F1} = u_1, v_1, w_1$ and T_{F2} = u_2, v_2, w_2be two TFN numbers Then the arithmetic operations on these fuzzy numbers can be defined as follows:

I. Addition: $T_{F1} + T_{F2} = u_1 + u_2, v_1 + v_2, w_1 + w_2$

2. Subtraction: $T_{F1} + T_{F2} = u_1 - u_2, v_1 - v_2, w_1 - w_2$

This new subtraction operation exists only if the condition $DP(T_{F1}) \geq DP(T_{F2})$ is satisfied where $DP(T_{F1}) = (w_1 - u_1)/2$ and$DP(T_{F2}) = (w_2 - u_2)/2$. Here DP denotes the difference point of a TFN. Otherwise, $T_{F1} = T_{F2}$ if $u_1 = u_2, v_1 = v_2, w_1 = w_2$

3. Equality: $T_{F1} = T_{F2}$ if $u_1 = u_2, v_1 = v_2, w_1 = w_2$

4. Multiplication: Suppose A = hi, ci) be any TFN number and B = (a2, b2, c2) be non-negative TFN number, then we define:

$$A \times B = \begin{cases} (a_1a_2, b_1b_2, c_1c_2), a_1 > 0; \\ (a_1c_2, b_1b_2, c_1c_2), a_1 < 0, c_1 \geq 0; \\ (a_1c_2, b_1b_2, c_1a_2), c_1 > 0; \end{cases}$$

5. $\max[(T_{F1}), (T_{F2})] = T_{F1}$ If $u_1 > u_2; v_1 > v_2; w_1 > w_2$

7. MINIMIZING OPERATIONAL COST OF MACHINES IN THREE MACHINE FLOW SHOP WITH UNCERTAIN PROCESSING TIMES

Branch and bind approach are used to solve the n-jobs, 3-machine flow shop issue in this part The triangle fuzzy membership function shows how long it takes for each machine to handle all of the tasks in the fuzzy environment. It is the goal of this part to produce an optimal or near-optimal work schedule, bearing in mind the planned recruiting strategy, to reduce the overall operating cost of machines. When it comes to renting, Bagga spoke about flow sequencing issues. Narian and Bagga developed algorithms for two and three-machine flow shop scheduling with a defined rental policy.

To lower the cost of renting machines, Singh and Viz have worked on two, three, and m-machine flow shop problems, respectively. In addition, Singh et al. expand their findings for the three and multi-machine issue by using a simple heuristic technique to include the task block criterion. They created effective heuristic models to reduce the cost of renting machines for flow shop models having two or three machines, according to Gupta and colleagues. Gupta et al. reduced rental costs by introducing the notion of an optimal waiting time into the research. A novel heuristic method was recently proposed by Narian in an effort to reduce the cost of recruiting in a multistage flow shop. The branch and bind approach are used in this section to reduce operating costs for the n3 flow shop issue.

Finally, the problem's computational results are presented for evaluation of the algorithm's execution. As a case study, this research might be used to a medicinal business. Even if one has to acquire machines to complete their work, the motivation for him is to keep costs low by avoiding speculative purchases of such equipment on a massive scale. For example, a doctor may recommend physiotherapy to a patient who has undergone surgery on a fractured or broken arm. To aid with recovery, physiotherapists recommend high-tech gear and treatments, which may be pricey over time. Renting the equipment rather than purchasing it makes sense in this situation.

1. HIRING STRATEGY (SH)

There are several scenarios in real life when one has tasks but lacks sufficient funds or does not like to take a risk by donating a significant sum of money to acquire the equipment needed to complete the duties. In these cases, the machines are rented using the established hiring tactics. Bagga and Narian outlined three distinct approaches to hiring:

Using Strategy, all computers are rented and returned at once. In the second strategy, all of the machines are rented at once and returned when they are no longer in use.

The third approach involves renting and then returning machines that will no longer be needed. In the actual world, the third kind of recruiting strategy (SH-III) is used. This policy states that all equipment is rented out as needed and returned as soon as it is no longer needed. Strategy III is tweaked in this section. As a result, the first machine will not be rented in this case, but the second and third machines will be rented as soon as they are needed for the processing of tasks. 2736

Following are the many notations that have been used in the progression of various sections of this chapter:

A description of the notations: Total number of jobs.

i: The indicator of employment.

j: An automated index.

SK: Branch and bound sequence created with the use of this approach, k = 1, 2.... n.

fij: This task's processing time on the jth machine is unknown.

hij: The amount of time it took to complete the task on machine Mj in terms of average hourly rate.

Jr: The amount of time it took to complete the task on machine Mj in terms of average hourly rate.

J’r: A position other than Jr.

Lb [Jr, c]: Lower bound on rental cost for schedule Jr. Other than Jr.

RC[j]: jth machine rental costs.

2. PROBLEM FORMULATION

Assuming that there are three computers, in a flow shop setting, M1, M2, and M3 are processing n tasks. Allow enough time to complete the current task(i=1,2,3.... n) on jth (j=1,2,3)TFN numbers are used to characterize the machine. As a result, the issue may be written as follows in mathematical form:

Table 1. Jobs with uncertain process time

Jobs	Machine M1	Machine M2	Machine M3
i	f_{i1}	f_{i2}	f_{i3}
1	$(\alpha_{11}, \beta_{11}, \gamma_{11})$	$(\alpha_{12}, \beta_{12}, \gamma_{12})$	$(\alpha_{13}, \beta_{13}, \gamma_{13})$
2	$(\alpha_{21}, \beta_{21}, \gamma_{21})$	$(\alpha_{22}, \beta_{22}, \gamma_{22})$	$(\alpha_{23}, \beta_{23}, \gamma_{23})$
3	$(\alpha_{31}, \beta_{31}, \gamma_{31})$	$(\alpha_{32}, \beta_{32}, \gamma_{32})$	$(\alpha_{33}, \beta_{32}, \gamma_{33})$
.	.	.	.
n	$(\alpha_{n1}, \beta_{n1}, \gamma_{n1})$	$(\alpha_{n2}, \beta_{n2}, \gamma_{n2})$	$(\alpha_{n3}, \beta_{n3}, \gamma_{n3})$

When it comes to solving the defined issue, the goal is to find an optimal employment plan that minimizes overall machine hire costs.

3. ALGORITHM FOR THE FORMULATED PROBLEM

The following approach is presented to investigate the nx3 FSS issue in a fuzzy environment in order to reduce machine hiring costs.

Step1: Apply the algorithm in 5.1.5 to de fuzz if y the provided processing time for tasks.

Step2: All schedules that begin with schedule Jr =J1 should be evaluated using the formula below:

(a) In the following formula, find the value of 'l' 2737

$$l = max\begin{Bmatrix} l_1 = (j,1) + \sum_{icj_r^n} h_{il} + min_{icj_r^n h_{i2}} \\ l_2 = (J_r,2) + \sum_{icj_r^n} h_{i2} \end{Bmatrix} - t_i, l$$

(b) Calculate 'L' using the following equation:

$$L = max\begin{Bmatrix} L_1 = t(j_r,1) + \sum_{icj_r^n}(h_{i2} + h_{i3}) \\ L_2 = t(j_r,2)\sum_{icj_r^n}(h_{i2} + min_{icj_r^n(h_{i2})} \\ L_3 = t(j_r,3) + \sum_{icj_r^n}(h_{i3}) \end{Bmatrix} - t_i, l$$

where hi1, hi2, and hi3 indicate the periods at which the machine should perform the ith task Mj, j=1,2,3 in terms of AHR and (,1) r t J, (,2) r t J, (,3) r t J machine Mj's for any partial scheduler completion timings.

Step3: Obtain Lb [Jr, c] = Scheduler rental costs cannot go below this threshold = J1of n× 3 inflow and outflow in a flow shop

$$L_b[J_{r,c}] = l \times R_c[2] + L \times R_c[3]$$

Step4: Choose the smallest possible lower limit Lb [Jr, c] among all the nodes that are not forked. Now let's have a look atLb [Jr, c] for a portion of the timetableJr =J2 of two jobs for the (n-1) subclasses that begin with the node in question and then narrow in on the bare essentials Lb [Jr, c] node. For as long as it takes us to get to the end of the tree, we'll continue to do it in this manner.

Step5: Step 4's sequence is the ideal one for achieving the goal of the n x 3 flow shop issue with the lowest rental cost, as was the case in step 3.

4. NUMERICAL ILLUSTRATION

The following example demonstrates how the algorithm works. Fuzzy process timings for a 4x3FSSP are shown in Table 2. The machines M2 and M3 are supposed to be rented at a cost of Rs. 4 per unit time and Rs. 6 per unit time, respectively. An ideal timetable that reduces the overall cost of employing machines is our goal.

Table 2. Description of processing time on the machines

Jobs	Machines		
	M_1	M_2	M_3
(i)	f_{i1}	f_{i2}	f_{i3}
1	(5,22,29)	(9,13,18)	(15,23,39)
2	(19,21,28)	(3,4,30)	(13,15,16)
3	(10,15,16)	(5,7,26)	(9,12,27)
4	(7,13,16)	(8,10,14)	(7,17,49)

Solution: The following is a description of how we came up with the above-mentioned solution:

Table 3. Machines with AHR

Jobs	Machine M_1	Machine M_2	Machine M_3
(i)	h_{i1}	h_{i2}	h_{i3}
1	30	16	31
2	24	13	16
3	17	14	18
4	16	12	31

The bottom limit of the range Jr = 1,2,3,4 Table 5.4 lists the nodes, and the scheduling tree is shown below:

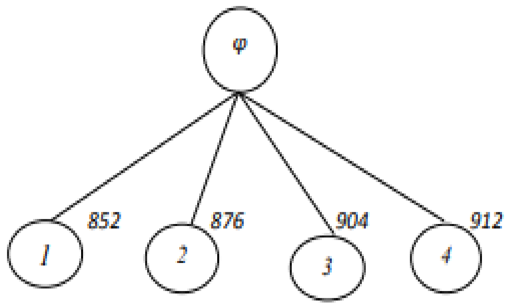

Figure 2. Branch and bound tree for first level of nodes

Table 4. Evaluation of first level lower bound

$J_r = J_1$	l_1	l_2	L_1	L_2	L_3	l	L	$L_b[J_{r,c}]$
1	**99**	**85**	**116**	**101**	**142**	**69**	**96**	**852**
2	99	79	119	97	133	75	96	876
3	99	72	116	88	127	82	96	904
4	100	71	116	87	124	84	96	912

The unbranched nodes' minimum node value is '1' with RC = Rs852. As a result, the scheduling tree's branching point is Jr=1. Take the two-job partial schedule Jr = J2, which originates at the branching node '1'as J2 = (12), (13), (14). Table 5.5 and Figure 5.3 include the necessary rental cost estimates.

Table 5. Evaluation of second level lower bound

$J_r=J_2$	l_1	l_2	L_1	L_2	L_3	l	L	$L_b[J_{r,c}]$
12	99	93	119	111	142	45	75	630
13	99	86	116	102	142	52	81	694
14	100	85	116	101	142	54	84	720

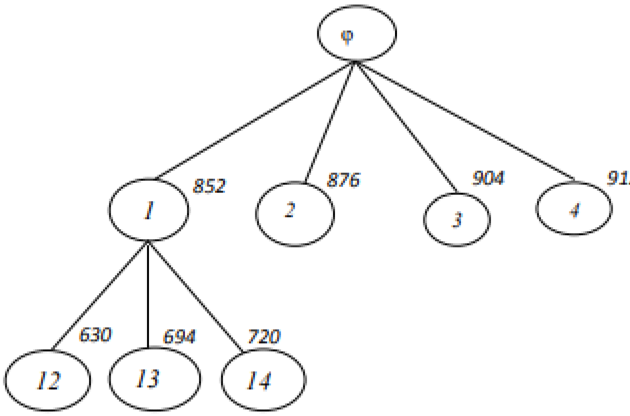

Figure 3. Branch and bound tree for second level of nodes

Here, min [Lb (Jr, c)] = (12), (13), (14)] = 630 which of the following is true Jr = (12). Hence Jr = (12) is the subsequent node in the tree. Take a look at the partial schedule now Jr = J3 Beginning with the branching node that was found in the last search '12'as J3 = (123), (124).

The rental price cap for Jr = J3 as seen in the table below:

Table 5. Evaluation of rental cost for Jr = J3

$J_r = J_3$	l_1	l_2	L_1	L_2	L_3	l	L	$L_b[J_{r,c}]$
123	99	97	130	128	142	28	57	454
124	101	96	119	114	142	31	60	484
132	99	96	130	127	142	28	58	460
134	100	88	116	104	142	37	67	550

2739

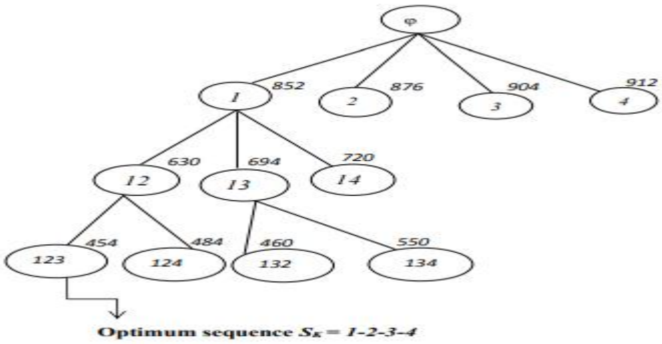

Figure 4. Branch and bound tree for third level of nodes

In light of the fact that the minimal amount of Lb [Jr, c] is Rs454 for Jr = 123 among all the nodes that are not forked. As a result, the minimal sequence is 1-2-3-4Lb [Jr, c] as 454. Consequently, at the third branch, we have found the best sequence SK = 1-2-3-4 reducing operating expenses. This section's approach is thus highly effective in solving the n x 3 flow shop issue to lower the operating expenses of the rented equipment.

5. RESULT ANALYSIS

In this part, a simple method and a numerical example are provided to demonstrate the validity of the suggested approach. We've also compared our algorithm's results to those obtained using Johnson's method. In comparison to Johnson's technique, the results obtained by the presented method are more effective in determining operational expenses. Figure 5 displays a comparative examination of the above-described situation, as given in the accompanying table.

Table 6. Comparison of the results

Technique	Total operational cost
Johnson's technique	(398,684,1034) ≅896
Proposed technique	454

2740

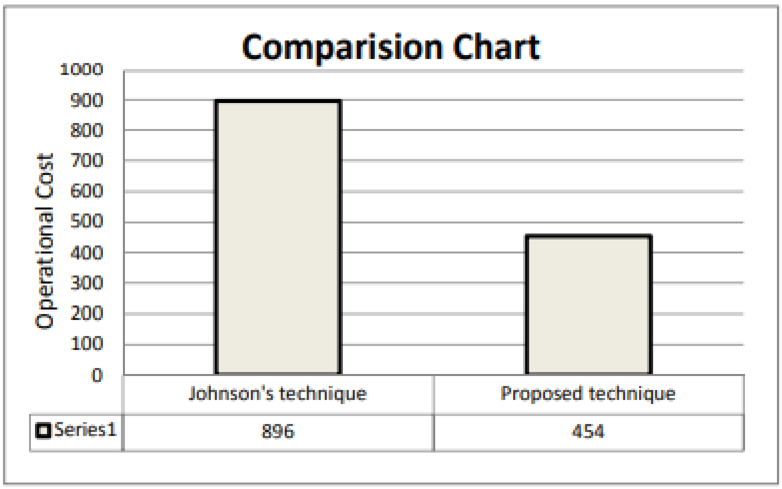

Figure 5. Comparison of the results between Proposed and Johnson's method

REFERENCES :

1. Oguz, C & Ercan, MF 2005, 'A genetic algorithm for hybrid flow-shop

scheduling with multiprocessor tasks', Journal of Scheduling, vol. 8, no. 4, pp. 323-351.

2. Osman, IH & Potts, CN 1989, 'Simulated annealing for permutation flowshop scheduling', OMEGA vol. 17, no. 6, pp. 551–557.
3. Palmer, DS 1965, 'Sequencing jobs through a multi-stage process in the minimum total time: a quick method of obtaining a near optimum', Operations Research, vol. 16, no. 1, pp. 101-107.
4. Pan, QK, Fatih Tasgetiren, M & Liang, YC 2008, 'A discrete particle swarm optimization algorithm for the no-wait flow-shop scheduling problem', Computers and Operations Research, vol. 35, no. 9, pp. 2807-2839.
5. Pan, QK, Tasgetiren, MF, Suganthan, PN & Chua, TJ 2011, 'A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem', Information Sciences, vol. 181, pp. 2455-2468.
6. Pavol Semančo & Vladimír Modrák 2012, 'A Comparison of Constructive Heuristics with the Objective of Minimizing Makespan in the Flow-Shop Scheduling Problem', Acta Polytechnica Hungarica, vol. 9, pp. 177-190.
7. Pempera Jarosław, Smutnicki Czesław & Żelazny Dominik 2013, Optimizing Bicriteria Flow Shop Scheduling Problem by Simulated Annealing Algorithm. Procedia Computer Science vol. 18, pp. 936- 945.
8. Pranzo, M 2004. Batch scheduling in a two-machine flow shop with limited buffer and sequence independent setup times and removal times. European Journal of Operational Research, vol. 153, no. 3, pp. 581-592.
9. Rajendran, C 1995, 'Heuristics for scheduling in flow shop with multiple objectives', European Journal of Operational Research, vol. 82, pp. 540-555.
10. Rajkumar, R & Shahabudeen, P 2009, 'Schdeduling Jobs on Flowshop EnvironmentApplying Simulated Annealing Algorithm', Intl J of Services and Operationsand Informatics, vol. 4, no. 3, pp. 212-231.
11. Rajkumar, R & Shahabudeen, R 2008, 'Scheduling jobs on flowshop environment with mean flowtime objective', IETECH Journal of Mechanical Design, vol. 2, no. 1, pp. 011-019.
12. Reeves, CR 1995. 'A genetic algorithm for flowshop sequencing', Computers and Operations Research, vol. 22, no. 1, pp. 5-13.
13. Al-Anzia & Allahverdi 2009, 'Heuristics for a two-stage assembly flow shop with bicriteria of maximum lateness and make span', Computers & Operations Research, vol. 36, no. 9, pp. 2682-2689.