Just in time multicriteria scheduling problem in two-machine flow shop

1st OUIDDIR Djedjiga
RECITS Laboratory, Faculty of Mathematics
University of Science and Technology Houari Boumediene
Algiers, Algeria
Djedjigaou@gmail.com

2nd BOUIBEDE HOCINE Karima
RECITS Laboratory, Faculty of Mathematics
University of Science and Technology Houari Boumediene
Algiers, Algeria
ka.hocine@gmail.com

3th BOUDHAR Mourad
RECITS Laboratory, Faculty of Mathematics
University of Science and Technology Houari Boumediene
Algiers, Algeria
mboudhar@usthb.dz and mboudhar@yahoo.fr

***Abstract*—In this paper, we are interested in the just in time multicriteria two-machine flow shop scheduling problem. Our aim is to highlight basic scheduling objective functions and the notion of just in time in production management. A cost function is defined for each job, taking into account additional constraints. We also position our problem with regard to what exists in the literature. First, we provide a brief literature review concerning the bi-criteria flow shop scheduling problem (FSP). Afterwards, we propose a mixed integer linear programming (MILP) formulation and two heuristics to solve the problem along with an experimental study we conducted in order to test their performance.**

***Index Terms*—scheduling, flowshop, just in time, multicriteria, MILP model, heuristic.**

I. INTRODUCTION

Flow shop scheduling problems (FSP) represent an important class of scheduling problems. Their importance and practical relevance to the industry have prompted researchers to consider them from different angles for decades. Indeed, there are several variants that take into account different constraints and optimize one or multiple criteria at a time. This scheduling problem can be defined generally as sequencing n jobs on m machines according to certain performance measure(s). Each job has to visit all the machines of the workshop. The running order is the same for all the jobs (unidirectional stream, i.e., jobs are sequenced in a linear way following a chain). A schedule meets these assumptions:

- A job can only be processed on one machine at a time.
- A machine can only process one job at a time.
- The processing of a job on a machine cannot be interrupted (no preemption).
- All jobs are independent and are available for processing at time zero.
- The machines are continuously available.

In the literature, generally, flow shop scheduling problems have been considered mono-criterion. However, multicriteria scheduling problems satisfy better the manager's requirements since many real-world scheduling problems are multi-criteria by nature. Therefore, this type of applications gets more and more interest from researchers.

Both tardiness and earliness cause penalties that can implicate losing customers and increasing inventory cost, respectively. Therefore, minimizing tardiness and earliness of jobs are the two most important objectives considered in the just-in-time systems. This problem has been handled for the first time by [1]. They proposed a simulated annealing (SA) metaheuristic to solve the $F|prmu, d_i, nmit|F_l(\bar{E}_w, \bar{T}_w)$ problem. In [2] authors addressed a two machine flow shop scheduling problem in which the objective function is to minimize the sum of maximum earliness and tardiness (n/2/P/ET_{max}) and proposed a B&B algorithm. To the best of our knowledge, no work related to our problem exists. We refer, using the standard three field notation of scheduling problems, to our problem by : $F2|d_i, s_{ii'j}| \sum T_i * B_i, \sum E_i * k_i$.

This paper is organized as follows. Section 2 is devoted to the literature review. In section 3, we define the problem understudy. In section 4 and 5, we present the MILP model and the heuristics, respectively, we developed to solve the problem. In section 6, we describe the extended experimental study we conducted and then we present the obtained results in section 7. We conclude the paper in section 8.

II. LITERATURE REVIEW

The literature contains several states of the art survey and numerous works that addressed the just in time multi-criteria scheduling problems. Moreover T'kindt V. and Billaut J-. C. dedicated a whole part of their work to it [3]. In the following we give a literature review on works that addressed the just in time multi-criteria flow shop scheduling problems. The works are classified according to the solving method.

A. Exact methods

In [4] is proposed a B&B approach for the two-machine flow shop scheduling problem with respect to the makespan and maximum tardiness for computing the pareto solution. [5] also presented a B&B procedure along with a heuristic approach for the two-machine flow shop scheduling problem to minimize total flow time subject to optimal makespan. [6] proposed a B&B procedure for the two machine flow shop problem to minimize a weighted sum of flow time and the makespan. They also presented a greedy algorithm for the upper bound of the B&B algorithm.

In [7] is developed a B&B procedure for the two-machine flow shop problem with respect to the makespan and number of tardy jobs. They also considered the makespan and total tardiness.

In [8] is developed three B&B algorithms with different branching strategies. They proposed two heuristics for the two-machine flow shop to minimize the makespan and the mean flowtime. [9] proposed an integer programming model and a heuristic algorithm for the two-machine problem in order to minimize a weighted sum of makespan and total flowtime.

In [10] is presented a B&B algorithm for a two-machine flow shop with makespan and flowtime.

In [11] is considered the minimization of a combination of total flow time and total tardiness in a two-machine flow shop. They proposed a B&B along with a lower bound and dominance properties to increase the performance of the algorithm.

[12] designed mathematical programming formulations, a B&B algorithm and a heuristic algorithm for the two-machine flow shop where the objective is to minimize total completion time and the makespan. The results showed that the B&B method proposed quite in handing MFSP.

In [13] is introduced a B&B approach for the two machine flow shop to minimize the makespan and the maximum tardiness. The algorithm was only able to solve small problem instances. Therefore they constructed a heuristic method. The tests shown that the heuristic was quite effective. Another B&B algorithm was presented in [14] for the two-machine flow shop, minimizing a weighted sum of the makespan and the total completion time, the proposed algorithm was able to solve up to 30 jobs.

[3] presented the B&B method for the two machine flow shop problem with the objective of minimizing the sum of maximum earliness and tardiness. The authors also introduced some useful lemmas and policies to reduce computational time. The performance of the B&B algorithm was evaluated over many problems of various randomly generated sizes. The results were quite encouraging.

B. Heuristics

In [15] is presented nine heuristics for the two machine flow shop scheduling problem minimizing the total flowtime subject to the makespan. The authors described some polynomially solvable cases and evaluated the performance of the proposed heuristics for finding approximate solutions. They showed that insertion heuristic based on the B&B algorithm proposed by [7] yielded the best results within satisfactory computational time duration.

[16] considered the minimization of the weighted sum of the makespan and the mean flowtime in the two and m-machine flow shop scheduling problem. The author compared existing heuristics for the two machine case. Three heuristics were proposed and they were shown to outperform the existing heuristics.

C. Meta-heuristics

In [17] was developed a simulated annealing algorithm for the two machine flow shop optimization problem with the objectives of minimizing the weighted total flowtime and the total tardiness.

[18] proposed a tabu search for solving the two machine flow shop problem with the makespan and the total flowtime as objectives.

III. Problem description

In this work, we are interested in solving the just in time bi-criteria two-machine flow shop scheduling problem, JITFSP. Given is a set $T = \{T_1, T_2, ..., T_n\}$ of n independent orders. Order T_i contains q_i pieces, that must be processed on two-machine M_1 and M_2 in a flow shop environment.

Each piece of an order T_i requires an execution time $p_{ij} \geq 0$ on the machine M_j. Let us point out that the pieces of the same order have the same processing time. We denote by:

- d_i : the delivery date of the order T_i.
- B_i : the tardiness cost of the order T_i.
- k_i : the earliness cost of the order T_i.
- $s_{ii'j}$: setup times representing the preparation time of machine M_j to receive the next order after the completion of the current one.

Since scheduling tasks on just-in-time consists of processing orders without advances and delays. Therefore we seek a schedule that minimizes both the earliness and tardiness that can occur during the processing of orders.

Just in time scheduling problems are an encompassing approach in order to make the finished product without the defect of all the required operational activities. Starting from the engineering of conception to the definitive delivery including all the stages of manufacturing.

We defined as objective, the minimization of two criteria, the earliness denoted, $f_1 = \Sigma\ k_i * Y_{1i}$, and the tardiness denoted, $f_2 = \Sigma\ B_i * Y_{2i}$, of the orders. Where Y_{1i} = max $\{0, d_i - C_i\}$, Y_{2i} = max $\{0, C_i - d_i\}$ and C_i represents the date of the end of treatment of the order T_i. For a better explain the problem we consider the following example summarized in Table I:

The optimal solution is given in Fig. 1, obtained by enumerating all the possible solutions.

T_i	T_1	T_2	T_3
p_{i1}	5	6	9
p_{i2}	7	8	5
q_i	3	1	1
d_i	75	80	81
B_i	7	7	10
k_i	16	6	11
s_{1ij}	(0,0)	(1,9)	(6,15)
s_{2ij}	(17,9)	(0,0)	(15,12)
s_{3ij}	(0,1)	(8,23)	(0,0)

TABLE I
EXAMPLE

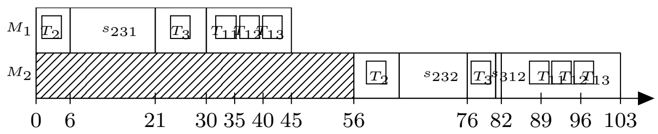

Fig. 1. Optimal solution.

IV. MILP MODEL AND COMPLEXITY

Referring to the work of Karp (1972) [19] the problem $F2||\sum T_i$ is NP-hard what makes the bi-criterion studied problem $F2|d_i, s_{ii'j}|\sum T_i * B_i, \sum E_i * k_i$ also NP-hard. Hence, no polynomial or psœudo polynomial algorithm exists for the resolution of this problem. Therefore, we propose an exact resolution using the following mathematical model.

We define the following decision variables:

- $x_{ii'} = \begin{cases} 1 & \text{if i precedes i'}, \\ 0 & \text{otherwise.} \end{cases}$
- y_{1i} : earliness of the order T_i.
- y_{2i} : tardiness of the order T_i.
- t_{ilj} : start time of the piece l of the order T_i on the machine M_j.

We also define:

M: a large number.

The objectif is to minimize the sum of the total cost of the earliness and the total cost of the tardiness of the orders ($\min \sum E_i * k_i$ + $\sum T_i * B_i$).

The MILP model is summarized in (P) :

$$
(P)\left|\begin{array}{ll}
\min & \sum y_{1i} * k_i + \sum y_{2i} * B_i \\
\text{s.t} & t_{iq_ij} + p_{ij} + s_{ii'j} \leq t_{i'1j} + M * (1 - x_{ii'}) \\
& \forall i, i' = \overline{1,n}, \forall j = \overline{1,2}, i < i'. \qquad (1) \\
& t_{i'q_{i'}j} + p_{i'j} + s_{i'ij} \leq t_{i1j} + (M * x_{ii'}) \\
& \forall i, i' = \overline{1,n}, \forall j = \overline{1,2}, i < i'. \qquad (2) \\
& t_{ilj} + p_{ij} \leq t_{il+1j} \\
& \forall i = \overline{1,n}, \forall l = \overline{1,q_i}, \forall j = \overline{1,2}. \qquad (3) \\
& t_{il2} \geq t_{il1} + p_{i1} \\
& \forall i = \overline{1,n}, \forall l = \overline{1,q_i}. \qquad (4) \\
& d_i - (t_{iq_i2} + p_{i2}) \leq y_{1i} \\
& \forall i = \overline{1,n}. \qquad (5) \\
& t_{iq_i2} + p_{i2} - d_i \leq y_{2i} \\
& \forall i = \overline{1,n}. \qquad (6)
\end{array}\right.
$$

(1) and (2) ensure the non overlapping of the orders on the same machine.

(3) ensures the non overlapping of pieces belonging to the same order T_i on the same machine.

(4) specifies that the piece of an order T_i can not begin processing on machine M_2 until it has completed its processing on machine M_1.

(5) and (6) equates earliness and tardiness of the order T_i.

V. HEURISTICS

We developed two heuristics H_1 and H_2 to solve the JITFSP which are presented in the following.

A. Heuristic H_1

Given the importance of due dates in the objective function $\sum E_i * k_i$ + $\sum T_i * B_i$. A possible approach to minimize the tardiness is to prioritize the jobs with smallest due date. Let us remind that the EDD order ranges the orders in the increasing order of : d_i.

Using the pervious reasoning we develop the following heuristic.

Algorithm 1 Heuristic H_1

```
All orders are arranged according to the EDD order;
The orders are sequenced on M_1 following the obtained order, without forgetting to insert setup times between two successive orders;
Sequencing on the second machine M_2:
We denote by S_{i-1} the partial sequence obtained at iteration i - 1.
for i = 1 to n do
    if max{C_max(S_{i-1}), (t_{i11} + p_{i1} + C_{iq_i2} - t_{i12})} <= d_i then
        Order T_i starts at d_i - C_{iq_i2} - t_{i12} and an idle time is inserted before T_i,
    else
        T_i is processed just after sequence S_{i-1}
    end if
end for
while the objective function is improved do
    the current partial sequence is shifted to the right.
end while
while the objective function is improved do
    the current partial sequence is shifted to the left.
end while
Calculate the objective function.
```

B. Heuristic H_2

The second heuristic, that we denote H_2, is presented as a new heuristic of solving by replacing the EDD order with the order which ranges the orders in the increasing order of: $d_i - p_{i1} * q_i$, minimizing the earliness of the orders in the case where $p_{i1} \leq p_{i2}$, in the first heuristic H_1 proposed.

VI. The computational study design

In order to test the previous proposed methods we conducted the following experimental study. The instances are generated randomly following the method described in [18]. The processing time p_{ij} of one piece of the order T_i is generated from a uniform distribution in [5,25]. Then q_i the number of pieces by order is generated from a uniform distribution in [1,50]. Due dates of the orders are generated using two parameters, the due date range R, (R=0.2, 1.6) and τ the tardiness factor ($\tau = 0.2, 0.6$) which are generated from a uniform distribution $[1-\tau-\frac{R}{2}, 1-\tau+\frac{R}{2}]$ where M=$\sum\sum p_{ij} * q_i + (n-1) * (\max(s_{ii'1}) + \max(s_{ii'2}))$.

To determinate the efficiency of the MILP model and the two proposed heuristics, the instances are classified in two groups. Each group is also divided into two types. The first group includes instances that mostly have tardiness and the second one those that mostly have earliness.

VII. Computational results

We start by testing the performance of the MILP model. Afterward we compare the results given by the two heuristics.

A. Computational results of the MILP model

In order to see the efficiency of the MILP model, the orders of the tested instances are classified into 8 different sizes with 5, 10, 20, 30, 40, 50, 100 and 120 orders. A run time is limit of 3600 seconds is imposed.

1) Computational results of the first group: In the first group two types of instances are generated according to the combination of the two factors of tardiness τ and range of due dates R. Type 1 has the values τ=0,2 and R=0,6. Type 2 has τ=0,6 and R=0,6. The computational results of this group are presented in Fig. 2 and 3. As shown in the two figures, the MILP model solves instances of Type 1 with up to 120 orders. The average maximum running time reaches 2098,76 seconds. For instances of Type 2, it is able to solve instances of 100 orders. The average maximum running time is of 1786,57 seconds.

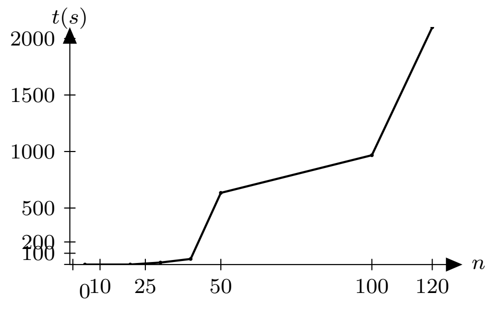

Fig. 2. Execution time for instances of Type 1.

In the second group, two types of instances are generated according to the combination of the two factors of tardiness τ and range of due dates R. Type 3 has the values τ=0,6 and R=1,6 and Type 4 has τ=0,2 and R=1,6, and the computational

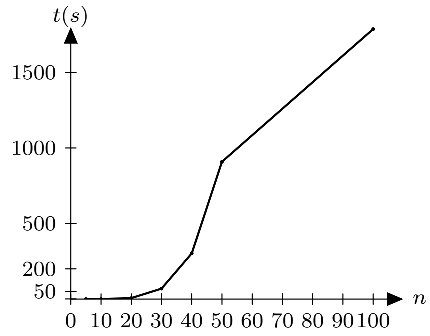

Fig. 3. Execution time for instances of Type 2.

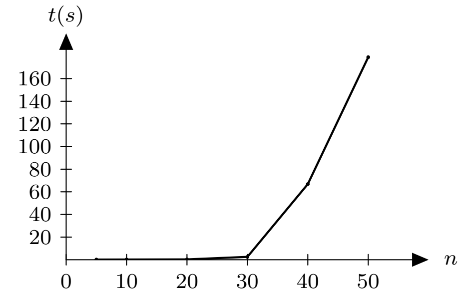

Fig. 4. Execution time for instances of Type 3.

results of this group are presented in Fig. 4 and Fig. 5. The MILP model is able to solve instances of Type 3 with up to 50 orders instances and the average maximum running time equal to 178,9 seconds. It is only able to solve instance of Type 2 with 30 orders. The average maximum running time reaches 1485,75 seconds.

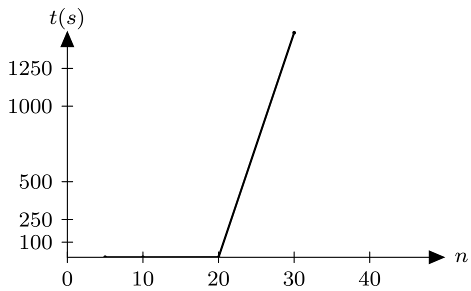

Fig. 5. Execution time for instances of Type 4.

From the precedent results, we can deduce that the model is efficient for most of time. We can conclude that the hard type instances are those of Type 4. Even for this type the model was able to solve instances with up to 30 orders. For another type, the model solved instances with a number of orders that reached 120.

B. Comparison of the two heuristics H_1 and H_2

To compare the results of the two heuristics proposed. The orders of the tested instances in the two groups are classified into 8 different sizes with 5, 10, 15, 20, 50, 100, 200 and 1000.

1) Solving first group instances: In the first group two types of instances are generated according to the combination of the two factors of tardiness τ and range of due dates R. Type 1 has the values τ=0.2 and R=0.6 and Type 2 has τ=0.6 and R=0.6, and the results obtained representing the number of times each heuristic is better than the other one are presented in Fig. 6 and Fig. 7.

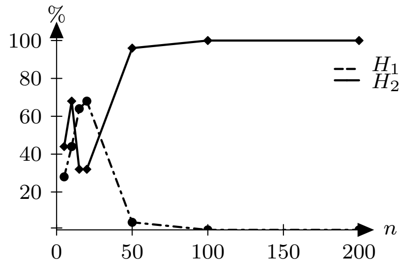

Fig. 6. Comparison of the two heuristics for instances of type 1.

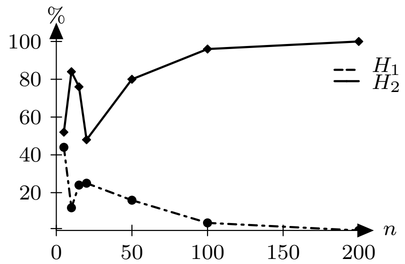

Fig. 7. Comparison of the two heuristics for instances of type 2.

2) Solving second group instances: In the second group, two types of instances are generated according to the combination of the two factors of tardiness τ and range of due dates R. Type 3 has the values τ=0,6 and R=1,6 and Type 4 has τ=0,2 and R=1,6, and the results obtained representing the number of times each heuristic out performs the other one are illustrated in Fig. 8 and Fig. 9.

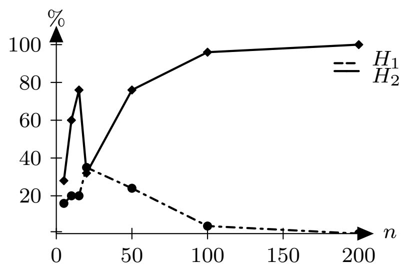

Fig. 8. Comparison of the two heuristics for instances of type 3.

As the four types shown, heuristic H_2 outperforms heuristic H_1. Therefore we compare in what follows the outperforming

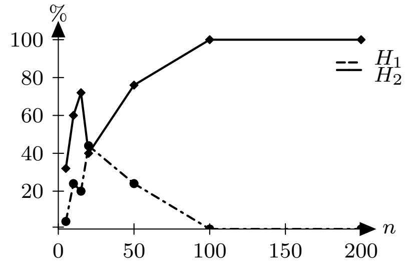

Fig. 9. Comparison of the two heuristics for instances of type 4.

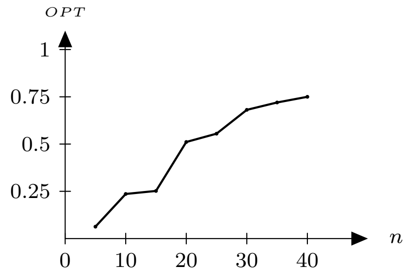

Fig. 10. Deviation between heuristic H_2 and the model.

heuristic and the MILP model. We generated instances for each type of instances and the orders are classified into 8 different sizes with 5, 10, 15, 20, 25, 30, 35 and 40 orders.

The parameter that we considered to compare heuristic H_2 and the MILP model is the deviation between the value of the objective function resulted using the heuristic denoted hereafter Zh_2 and the one given by the model denoted by OPT, that is calculated by the following formula: $1 - OPT/Zh_2$. The obtained results are summarized in Fig. 10. We can notice that the efficiency of the heuristic diminishes while increasing the number of orders. However, the difference between the two results is not huge.

VIII. Conclusion

In this paper, we have considered the bi-criteria just in time two-machine flow shop scheduling problem, in which the objective function is to minimize both total cost of tardiness and total cost of earliness. We have proposed a MILP model and two heuristics H_1 and H_2 to solve the problem. The experimental study shown the efficiency of the model and heuristic H_2.

References

[1] S.H. Zegordi, K. Itoh and T. Enkawa, "A knowledgeable simulated annealing scheme for the early/tardy flow shop scheduling problem", International Journal of Production Research, 33, 1449-1466, 1995.

[2] G. Moslehi, M. Mirzaee, M. Vasei, M. Modarres and A. Azaron, "Two-machine flow shop scheduling to minimize the sum of maximum earliness and tardiness", Int J Prod Econ, 122 (2), 763-773, 2009.

[3] V. T'Kindt and J. C. Billaut. "Multicriteria Scheduling: Theory, Models and Algorithms". Springer 2006.

[4] R. L. Daniels and R. J. Chambers," Multiobjective flow shop scheduling", Nav Res Logist Q, 37, 981-995, 1990.

[5] C. Rajendran, "two-stage flowshop scheduling problem with bi-criteria", J Oper Res Soc, 43 (9), 871-884, 1992.

[6] A. Nagar, S.S. Heragu and J. Haddock, “A branch and bound approach for two-machine flowshop scheduling problem”, J Oper Res Soc, 46, 721-734, 1995.
[7] C.J. Liao, W.C. Yu and C.B. Joe, “Bicriterion scheduling in the two-machine flowshop”, J Oper Res Soc, 48 (9), 929-935, 1997.
[8] F. Sivrikaya-Serifoglu and G. Ulusoy, “A bicriteria two-machine permutation flowshop problem”, Eur J Oper Res, 107, 414-430, 1998.
[9] F.D. Chou and C.E. Lee, “Two-machine flowshop scheduling with bicriteria problem”, Comput Ind Eng, 36, 549-564, 1999.
[10] S. Sayin and S. Karabati, “A bicriteria approach to the two-machine flowshop scheduling problem”, Eur J Oper Res, 112, 435-449, 1999.
[11] W.C. Lee and C.C. Wu, “Minimizing the total flow time and the tardiness in a two-machine flow shop”, Int J Syst Sci, 32 (3), 365-373, 2001.
[12] V. T’kindt, J.N.D. Gupta and J.C. Billaut, “Two-machine flowshop scheduling with a secondary criterion“, Comput Oper Res, 30 (4), 505-526, 2003.
[13] B. Toktas, M. Azizoglu and S.K. Kksalan, “Two-machine flow shop scheduling with two criteria: Maximum earliness and makespan”, Eur J Oper Res, 157 (2), 286-295, 2004.
[14] B.M.T. Lin and J.M. Wu, “Bicriteria scheduling in a two-machine permutation flowshop”, Int J Prod Res, 44 (12), 2299-2312, 2006.
[15] J.N.D. Gupta, K. Hennig and F. Werner, “Minimizing total flowtime in a 2-machine flow shop problem with minumum makespan”, Int J Prod Econ, 69, 323-338, 2001.
[16] A. Allahverdi, “The two- and m-machine flowshop scheduling problems with bicriteria of makespan and mean flowtime”, Eur J Oper Res, 147 (2), 373-396, 2003.
[17] M. Mesgarpour, N. Kirkavak and H. Ozaktas, “Bicriteria scheduling problem on the two-machine flowshop using simulated annealing”, Evolutionary Computation in Combinatorial Optimization, Lecture Notes in Computer Science, vol. 6022, Springer, pp. 166-177, 2010.
[18] J.N.D. Gupta, N. Palanimuthy and C.L. Chen, “Designing a tabu search algorithm for the two-stage flowshop problem with secondary criterion”, Prod Plan Control, 10, 251-265, 1999.
[19] KARP. R. M., “Reducibility among combinatorial problems. In Complexity of Computer Computations”. R. E. Miller and J. W. Thatcher (eds) (New York: Plenum Press). pp.83-103, 1972.