Journal of Information Systems Engineering and Management
2026, 11(2s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

A Novel Approach for Scheduling Problems on Machine Single

Aissa LAKHAL[1] ,omar selt[2]
[1]*Faculty of Mathematics and computer science, University of M'sila, PO Box 166 Ichebilia, M'sila 28000, Algeria.*
Corresponding Author, aissa.lakhal@univ-msila.dz
[2]*university ziane achour of djelfa ,selt.omar@univ-djelfa.dz*

ARTICLEINFO

Received: 29 Dec 2024

Revised: 15 Feb 2025

Accepted: 24 Feb 2025

ABSTRACT

In this paper, we will present a comparative study between two neighborhoods for solving scheduling problems on machine single to minimize the weighted sum of the task's end dates; since this problem is NP-hard in the strong sense, exact methods require a computational effort that increases exponentially with the size of the problem.

Approximate methods allow this problem to be solved reasonably.

In this paper, we present the tabu search metaheuristic; it aims to find an approximate approach to the problem under consideration.

We present the tabu search method and detail the parameters and main steps of the proposed approach, and the results obtained are applicable in economics and industry.

Keywords: NP-hard-planning-scheduling-tabu search- neighborhoods

1. INTRODUCTION

The scheduling problem consists of organizing the time taken to complete tasks, taking into account time constraints (deadlines, sequencing constraints) and constraints relating to the use and availability of the required resources.

A schedule is a solution to the scheduling problem; it describes the execution of tasks and the allocation of resources over time and aims to satisfy one or more objectives. More precisely, we speak of scheduling when we reserve the term sequencing for the case where only the relative order of tasks is fixed independently of the execution dates.

Tasks

A task **i** is an elementary work entity (operation or set of operations), located in time by a start date ri or end date $\mathbf{C_i}$, the completion of which requires a duration $\mathbf{p_i = C_i - r_i}$ and which consumes resources (material, personnel, monetary, etc.).

The constraints

In most scheduling problems, the tasks to be executed are subject to constraints that must be satisfied when searching for an optimal solution. There are three types of constraints.

Potential constraints

These are the time location constraints, for example, "task **i** must precede task **j** " or task **i** must be completed before a certain date. With this type of constraint, the problem is called a "central scheduling problem".

Journal of Information Systems Engineering and Management
2026, 11(2s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Disjunctive constraints

When two tasks cannot be performed at the same time, we say that there is a disjunctive constraint that both tasks must satisfy.

Cumulative constraints

They concern the evolution over time of the total volume of human or material resources devoted to the execution of tasks.

Resources

Resources are the means required for the execution of tasks, they are of two types.

A resource is consumable if, after being allocated to one task, it is no longer available for other tasks, similar to the case of a raw material or money.

A resource is renewable if, after being allocated to a task, it becomes available again, upon completion of that task, for other tasks, such as a machine, a process, a printer, etc.

The criteria

The criterion of a scheduling problem is the economic function that one aims to optimize. It is also called in another sense (objective function).

In general, the criterion is considered as an application **F** from the set $\sum$ of permutations of the tasks of the problem posed to $\mathbf{R}^+$ which, at each permutation

$(j_1, j_2, \ldots j_n)$ associates a positive real number

$F(j_1, j_2, \ldots j_n)$ and which expresses, in addition to the efficient use of resources, the overall execution time and compliance with the greatest number of constraints, the objective function associated with a scheduling problem is defined by: $F\ obj = \underset{\sigma \in \Sigma}{OPT}\ \ F(\sigma)$

2. NEIGHBORHOOD METHODS

Neighborhood methods are based on the notion of neighborhood.

Definition

Let X be the set of admissible configurations of a problem. A neighborhood is any application $N : x \rightarrow 2^x$, and a neighborhood exploration mechanism is any procedure that specifies how the search moves from a configuration **S** $\in$ **X** to a configuration.**S′** $\in$ **N(s)**

Configuration S is said to be a local optimum (minimum) concerning the neighborhood N if
$\forall S' \in N(S) : f(S) \leq f(S')$

A typical neighborhood matching method starts with an initial configuration and then performs an iterative process that consists of replacing the current configuration with one of its neighbors, taking into account the cost function. This process stops and returns the best configuration found when the stopping condition is met. This condition can generally be a limit on the number of iterations.

Initialization: Generate an initial population P of solutions of size |P| = n

Algorithm

Repeat

Crossover: Combine the two parent solutions X and Y to form a child solution z

Conditional mutation of z

Journal of Information Systems Engineering and Management
2026, 11(2s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Choose an individual solution to be replaced in the population

Replace the chosen individual with Z in the population

Until the stopping criterion is met

3. PROPOSED METHOD

Tabu Search

The Tabu method was introduced simultaneously by several researchers, including P. Hansen, F. Glover,and B. Jaumard(1986).

Tabu search is used because it prohibits the retrieval of recently visited solutions.

At each iteration, the least-worst neighbor is chosen.

To avoid cycles, i.e., the infinite repetition of a sequence of moves, the last L moves are considered forbidden, and L is the size of the Tabu list.

At each iteration, the move performed is therefore the least-worst non-Tabu move.

Tabu List

The Tabu list represents short-term memory; it contains the attributes of the most frequently performed moves.

This list is maintained to guide the search.

Aspiration Criterion

This criterion is often used to remove Tabu restrictions from a high-quality move. It allows bypassing certain forbidden cases. Its main use is to override the prohibition of a move if it allows to obtain an element better than the solution found so far (consists of revoking the Taboo status of a move if the latter allows to obtain a higher-quality than that of the best solution found so far).

Intensification Criterion

Intensification consists of returning to one of the best solutions found so far and then resuming the search for this solution.

The intensification strategy is embodied in the following algorithm by reinforcing the search in the list of best moves.

Diversification Criterion

This consists of generating a new solution, different from the one already explored, with the aim of moving in a new direction, to explore another region.

Stopping Criterion

Generally, to stop the algorithm, two criteria are taken into account or ignored. First, at each iteration, the total iteration counter has not exceeded a maximum number since the beginning of the process.

We base our calculations on another threshold value, which corresponds to the maximum number of iterations we allow between two consecutive modifications (improvements) of the best solution.

But instead of considering the number of iterations, we could also base our calculation on the total time, which should be less than a maximum value.

General Tabu Search Algorithm

Below we present the general Tabu Search algorithm, with an emphasis on short-term memory and aggressive exploration:

Generate an initial solution S randomly

Journal of Information Systems Engineering and Management
2026, 11(2s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

S*← S ; C*← F(S) / S* is the best solution encountered, C* is its cost, and F is the objective function

Add S to the tabu list; K← 0

Repeat until an end criterion is met

Choose from the neighborhood of S_K, $V(S_K)$, the move that minimizes and that does not belong to the tabu list, best(S_K)

S_{K+1} ← best (S_K)

If the tabu list is full, then

Replace the last element with S_{K+1}

End if

Add S_{K+1} to the tabu list

If (C (S_{K+1}) < C*) then

S*← S_{K+1}, C*← C (S_{K+1})

End if

End

4. FORMULATION MATHEMATICS

This problem is formulated as:

$$\left.\begin{array}{rrcl} \min_x & \sum_{i=1}^{m} f_i x_i & & \\ \forall\, j \leq n & \sum_{i=1}^{m} a_{ij} x_i & \geq & 1 \\ \forall\, i \leq m & x_i & \in & \{0,1\} \end{array}\right\}$$

Proposition:

In each availability period, jobs must be scheduled according to the WSPT rule in an optimal schedule.

$$\frac{p_h}{w_h} = \min\left\{\frac{p_k}{w_k}\right\}$$

Table 1. Results by two approaches

Jobs		IS	TS		TI		TIB		BC
			AS	TM	AS	TM	AS	TM	
N	**20**	**29190**	41647	0,65	31779	0,307	35466	0,13	**29190**
		45768	40772	0,57	**29096**	0,224	32259	0,166	**29096**

Journal of Information Systems Engineering and Management
2026, 11(2s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

		30731	**29720**	0,57	37763	0,447	33526	0,161	**29720**
N	**50**	205130	189551	0,63	223921	1,524	**2E+05**	0,328	**2E+05**
		207358	**2E+05**	0,60	219091	1,019	229906	0,437	**2E+05**
		214071	209126	0,64	**2E+05**	0,973	208584	0,437	**2E+05**
	100	734976	682309	3,68	**6E+05**	5,179	707554	1,617	**6E+05**
		2E+06	**7E+05**	3,38	786843	5,356	761648	1,472	**7E+05**
		839684	707977	3,29	**7E+05**	5,808	703850	1,700	**7E+05**
	250	5E+06	4E+06	7,73	5E+06	9,18	**4E+06**	6,56	**4E+06**
		5E+06	**3E+06**	7,61	4E+06	9,63	4E+06	6,49	**3E+06**
		5E+06	4E+06	6,95	**3E+06**	8,89	4E+06	6,92	**3E+06**
		8E+07	**6E+07**	20	7E+07	25	7E+07	18	**6E+07**
		7E+07	7E+07	20	**6E+07**	25	7E+07	18	**6E+07**

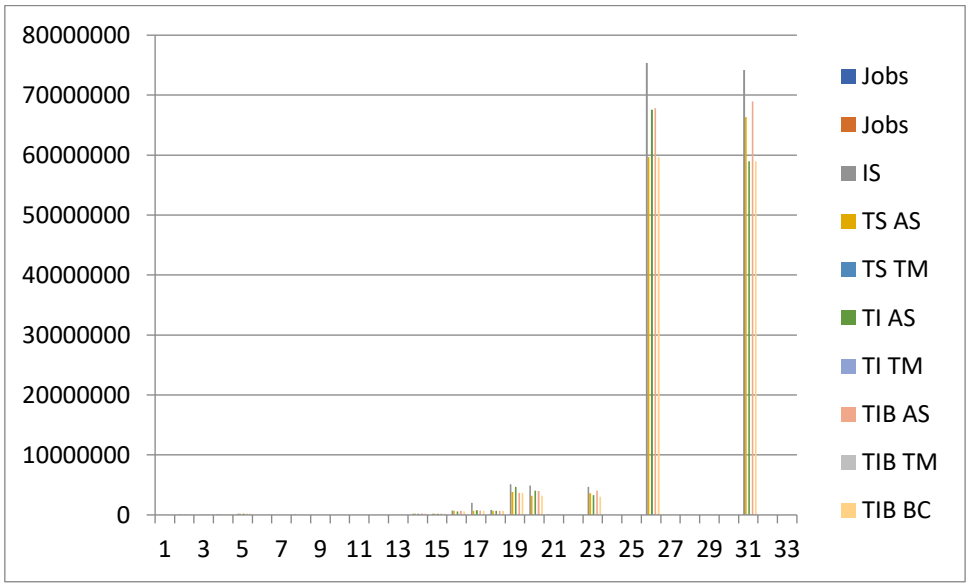

Fig 1 .amelioration cost between two neighborhoods

Journal of Information Systems Engineering and Management
2026, 11(2s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

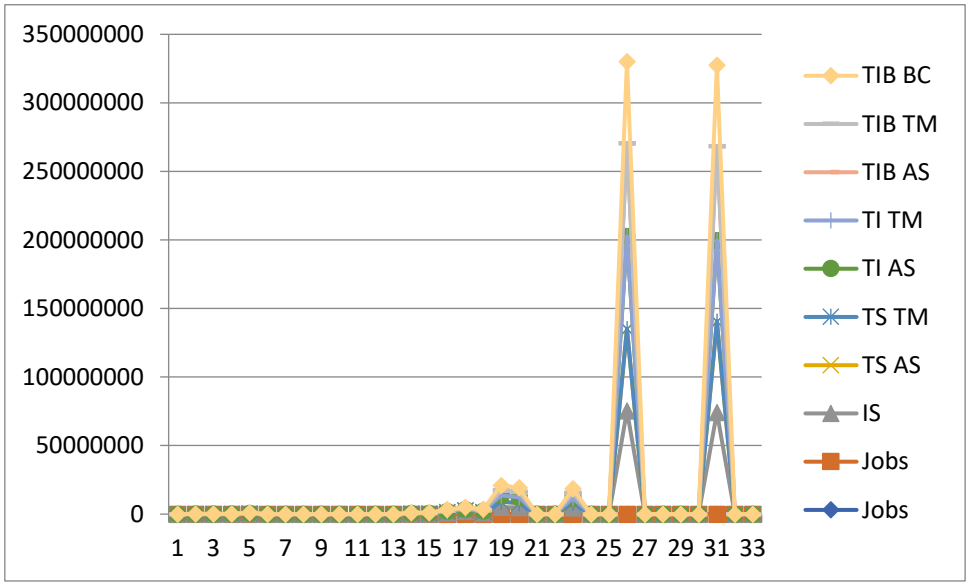

Fig 2. The best cost

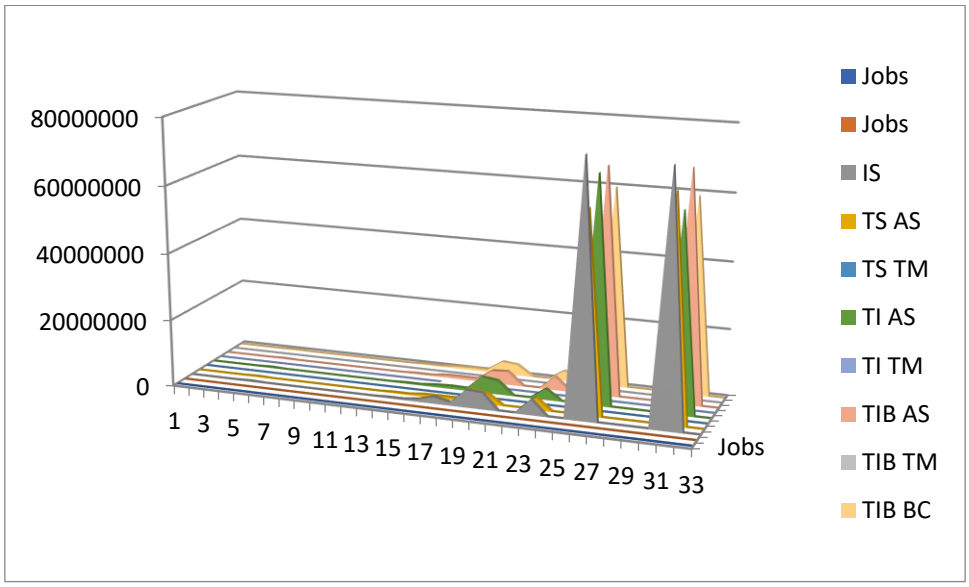

Fig 3 .amelioration cost between two neighborhoods

5. CONCLUSION

In this work, we presented a tabu search method for solving the scheduling problem on a single machine with unavailability periods and we suggested the different types of neighborhoods. Each time, we noticed the improvement in the solutions for our problem and the best solutions based on neighborhood swapping.

Therefore, the neighborhood improvement strategy (swapping two adjacent jobs) is better than the neighborhood by block; when the number of jobs is less than 50, but for jobs exceeding 50, neighborhood by block gives better results with minimal execution time.

REFERENCES

[1] Adamu, MO., and Adewunmi, A. (2012) Metaheuristics for scheduling on the parallel machine to minimize the weighted number of early and tardy jobs. Int. J. Phys. Sci. 7(10): 1641-1652.

[2] Glover, F. and Hanafi, S . (2002) Tabu Search and Finite Convergence, Special Issue on Foundations of heuristics in Combinatorial Optimization. Discrete Appl. Math. 119: 3-36.

[3] Glover, F. (1986) Futurepathsfo integer programming and links to artificial intelligence, Comput. Open Res. 13: 533-549.
[4] Hansen, P. (1986) The steepest ascent mildest descent heuristic for combinatorial programming. In: Proceedings of the Congress on Numerical Methods.
[5] Ho, JC., and Chang, YL.(1995) Minimizing the number of tardy jobs from parallel machines. Eur. J. Oper. Res. 84: 334-355.
[6] Lee, CY.(1996) Machine scheduling with an availability constraint. J. Global Optim. 9:395-416.
[7] Lee, C.Y.(1997) Minimizing the makespan in two machines flow shop scheduling problem with availability constraints. Oper. Res. Lett. 20:129-139.
[8] Lee, C.Y.(1999) Two machines flow shop scheduling problem with availability constraints. European J. Oper. Res. 114:420-429.
[9] M'Hallah, R., and Bulfin, RL.(2005) Minimizing the weighted number of tardy jobs of parallel processors. Eur. J. Oper. Res .160: 471-484.
[10] Schmidt, G.(2000) Scheduling with limited machine availability. European J. Oper. Res. 121:1-15.
[11] Smith, WE. (1956) Various optimizers for single-stage production. Naval Res. Logistics. 3:59-66.
[12] Yun-Chia, L, H., Yu-Ming. and Chia-Yun, T.(2013) Taiwan PCB industries. Int. J. Prod. Econ. 141(1):189-198.
[13] Zribi. I, Kacem. I, and Borne. P. (2005) Minimisation de la Somme des retards dans un job shop flexible. Revue e-STA (SEE). 2(2):2-11
[14] Zitouni. R and Selt. O (2016) Metaheuristics to solve task scheduling problems in parallel identical machines with unavailability periods, RAIRO. Oper. Res,50(1), pp. 83-90.