International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
IJISAE
ISSN:2147-6799 www.ijisae.org Original Research Paper

# Energy-Efficient Scheduling of Uniform Parallel Machines Problem with Deterioration Effect

**Norelhouda Sekkal*[1], Mouna Elouchdi[2], Khadidja Bakdi[2]**

**Submitted**: 07/01/2024 **Revised**: 13/02/2024 **Accepted**: 21/02/2024

**Abstract:** Uniform Parallel Machine (UPM) scheduling problems have received considerable attention due to their numerous industrial applications. These systems are, however, prone to deterioration. Therefore, preventive maintenance activities are integrated to the system, creating unavailability periods. Meanwhile, deterioration and maintenance activities increase the energy consumption in the workshop. And with growing concerns about the environment and global warming, energy consumption has become a priority for manufacturers. Thus, we treat a uniform parallel machine scheduling problem with deterioration effect and maintenance activities to minimize the total energy consumed. First, we present a mathematical programming model. This model is not able to provide a solution for medium instances in an affordable computational time. However, we adapt to our problem a simulated annealing algorithm. To evaluate the efficiency of the algorithm, we test it and c results with the mal solutions given by the model. Then, we test the algorithm for medium and large instances.

***Keywords:*** *Energy efficiency, uniform parallel machines, deterioration effect, maintenance activities*

## 1. Introduction

The excessive use of energy is one of the factors leading to all these problems from which nature suffers, such as global warming, air pollution, and the depletion of fuel resources. According to [1] the industrial sector is responsible for 50% of total energy consumption. As a result, the industrial sector is responsible for increasing energy consumption and all consequent damages, hence the need for more attention and effort for more energy efficiency. We mean by energy efficiency the reduction of energy consumption with the same quality of service.

As the demand for energy increases, the use of energy efficiency becomes vital. For this, different strategies have been developed. Strategies to reduce total energy consumption, and others to increase the rate of production per unit of energy consumed. According to [2] and [3] the implementation of new machines that consume less energy is one of the methods used to optimize energy consumption. But these methods are not always efficient, where the need for a production scheduling plan.

Production scheduling problems are widely studied in the literature. Traditional ones consider the job processing time as fixe throughout the whole process. However, there are numerous situations in which the real processing times are dependent on the position of the product in the machine or on its starting time. [4] Addressed a scheduling problem in which the processing time is position dependent. The deterioration effect increases the processing time, which may generate a lot of loss in terms of time and energy. Meanwhile, it is possible to restore the deteriorated processing time to its original value through e.

Also, in real world applications, each machine works with variable efficiency and depreciate because of usage and age at different rate. By this effect, the job processing time varies from a machine to another according to the speed of the machine. [5] and [6] addressed scheduling problems involving uniform parallel machines.

In this sense, we propose an integrated model to formulate a uniform parallel machines problem with:

- Deterioration: where the processing time of a job becomes variable rather than constant and depends on the starting time or the position of the job in the machine. This constraint is found particularly in steel rolling mills, in the health sector, where the slightest delay can worsen the situation. But in general, any machine deteriorates and loses its performance over time.
- The unavailability of the machine due to maintenance activities; these activities are essential to reduce the influence of deterioration effect on the system by restoring the machine to its initial state.

Job processing times are dependent on two parameters:

- The speed of the machine where the job is assigned.

[1] Ecole Supérieure en Sciences Appliquées Tlemcen, ESSA-Tlemcen, BP 165 RP Bel Horizon, Tlemcen 13000, Algeria
University Abou Bakr Belkaid Tlemcen, Faculty of technology, MELT laboratory, Tlemcen, Algeria
* Corresponding Author Email: sekal.nor@hotmail.fr

- The position where the job is processed.

The maintenance activity is made after processing a specified number of jobs. The maintenance duration in a machine is predetermined according to the speed of the machine, the higher the speed of the machine, the longer the maintenance. After each maintenance activity the machine get back its initial performances, the positions are reset to 0 and starts a new production period or as named in [7] “a bucket”.

The amount of energy each machine consumes depends not only on the machine's operating time (the time it takes to process the jobs assigned to it), but also on the duration of maintenance. This problem is solved by finding the right assignment of jobs to machines and deciding on the number of maintenance activities, while minimizing the total energy consumed.

## 2. Literature Review

In manufacturing, processing time may be subject to change due to several effects. [8] were among those who introduced the deterioration effect to scheduling problems. Over the years, more and more attention has been paid to planning with deterioration constraints. [9] Reviewed the existing scheduling models with time dependent processing time. And [10] is a very recent study, which proposed a decomposition algorithm to solve the parallel machines scheduling problem with deterioration jobs. Also, [10] addressed a parallel machines scheduling problem involving linear job deterioration in order to minimize delays and machine deterioration costs. [11] Proposed a novel algorithm to solve an identical parallel machines scheduling problem with deteriorating jobs. [12], [13] both proposed approaches for minimizing the makespan in a system of parallel machines under start time dependent processing time. [14] Considered unrelated parallel machiness scheduling problem with deteriorating effect to minimize makespan. The deterioration of a machine depends on the sequence of jobs processed in it. They developed a simulated annealing algorithm to adress the problem. In the same context, [15] used a simulated annealing algorithm in a decomposition approach to solve a parallel machines scheduling problem with deterioration and human resources constraints.

In parallel, a deteriorating processing time can be restored to its original length through a maintenance activity. These activities, named in the literature RMA (rate modifying activity), were introduced for the first time by [16]. [17] considered a single machine scheduling problem with positioning RMAs. [18] developed an algorithm for solving two identical parallel machines with no more than one RMA in each machine. [19] considered a parallel machines scheduling problem with deterioration and RMAs, to minimize completion times. [20] considered a single machine scheduling problem with deteriorating jobs and RMAs. [7] studied the single machine scheduling problem where a piecewise linear processing time function is used to minimize the makespan.
$p_i \text{OBJ} x_i \le d_i \text{OBJ} p_i + r_i . d_i \text{OBJ} x_i \text{OBJ} d_i \text{OBJ} x_i \text{OBJ} d_i \text{OBJ} r_i \text{OBJ} \text{OBJ} \text{OBJ}$
proposed a mathematical programming model to scheduling either production or maintenance in a single machine, by studying the machine’s reliability.

Not only single machine environments are considered, but also parallel machines scheduling problems also arise in many production systems, and they receive an important intention in literature. [22] considered an unrelated parallel machiness scheduling problem with maintenance activities. [23] extended the study of [18] to an unrelated parallel machiness scheduling problem with RMA and presented an efficient algorithm to minimize the total completion time. And recently, [24] treated an unrelated parallel machiness scheduling problem with machine unavailability due to maintenance activities.

The deterioration effect can also influence the duration of a maintenance activity: [24] considered an unrelated parallel machiness scheduling problem, with at most one RMA by machine. The RMA duration is a function of it starting time. [25] considered a single machine scheduling problem with aging effect (deterioration) and RMAs. Two effects influence the RMA duration, learning effect and deterioration effect. [26] studied a scheduling problem on a single machine. It was assumed that the processing time of a job depended on the sum of certain parameters linked to the jobs previously processed. They considered the case where a maintenance activity is included.

The deterioration increases processing time of jobs and so energy consumption of the machine, a maintenance activity reduces the deterioration effect by restoring the machine to its initial performances. But also, a maintenance activity consumes energy. So, the objective is to schedule jobs and RMAs in machines to minimize the total energy consumption

### 2.1. Contribution

The literature shows a lack of papers dealing with deterioration effect, maintenance and energy consumption at the same time. This paper treats a uniform parallel machines scheduling problem with deteriorating jobs, maintenance activities to minimize the total consumed energy.

Compared to the mentioned above, this study is specified by:

- A developed mathematical model which aims to minimize energy consumption, considering rate modifying activities.

- A proposed lemma, which is mathematically proven.
- Most previous studies consider maintenance activities as a period of machine unavailability. This study considers the maintenance activity in terms of energy consumption.
- Energy consumption is a function of the processing time of the machine and the maintenance activities duration
- A Simulated annealing and lemma-based algorithm is developed to solve efficiently the problem.

## 3. Problem Statement

In this section we present a mathematical model for a uniform parallel machine problem with deterioration and maintenance activities to minimize the total consumed energy. We assume that there are N=1…j independent jobs to be scheduled in M=1…i uniform parallel machines. A job deteriorates and machines have different speed $V_i$. So, a processing time of a job depends on it position in the machine and on the machine's speed $V_i$, it is formulated as follow: $Pr_{pij} = \frac{a_j}{V_i} + \alpha.(p-1)$. The deteriorated processing times of jobs are restored to their original processing time by RMAs. Period between RMAs in a machine are named bucket. In each bucket unlimited number of jobs can be processed. A machine j consumes δ energy per unit of time.

Parameters:

N=1…j number of independent jobs.

M=1…i number of uniform machines.

B=1…k number of bucket.

P=1…p number of position in a bucket

$Pr_{pik}$: job processing time in position p of bucket k machine i

$V_i$: speed of machine i

α deterioration compression rate

$\delta_1, \delta_2$processing and maintenance time compression rate, respectively.

$d_{pik}$: job start time in position p, bucket k, machine i.

$C_{pik}$:: job completion time in position p bucket k machine i.

$Cm_{ik}$:total completion time in bucket b machine i

$E_i$: consumed energy in machine i.

$nk_i$: Number of maintenance activities.

RMA: duration of a maintenance activity.

Decision variables:

$X_{jpik}$= 1 if job j is treated in machine I, bucket k, position p.

$Y_{ik}$=1 if bucket k of machine I is loaded

### 3.1. The mathematical linear programming model

| | |
|---|---|
| $OF = \sum_{i=1}^{M} E_i$ | (1) |
| $\sum_{j=1}^{N} X_{jpik} \leq 1 \quad \forall p,i,k$ | (2) |
| $\sum_{i=1}^{M} \sum_{k=1}^{B} \sum_{p=1}^{P} X_{jpik} = 1 \quad \forall j$ | (3) |
| $\sum_{j=1}^{N} X_{jpik} \leq \sum_{j=1}^{N} X_{j(p-1)ik} \quad \forall i,k,p>1$ | (4) |
| $\sum_{j=1}^{N} X_{j1ik} = Y_{ik} \quad \forall i,k$ | (5) |
| $Y_{ik} \leq Y_{i(k-1)} \quad \forall i,p>1$ | (6) |
| $Pr_{pik} \geq \sum_{j=1}^{N} X_{jpik} \frac{a_j}{V_i} + \alpha(p-1) \quad \forall p,i,k$ | (7) |
| $d_{1p1} \geq 0 \quad \forall p$ | (8) |
| $C_{pik} = d_{pik} + Pr_{pik} \quad \forall p,i,k$ | (9) |
| $d_{pik} \geq C_{(p-1)ik} \quad \forall p,i,k$ | (10) |
| $Cm_{ik} \geq C_{pik} \quad \forall p,i,k$ | (11) |
| $d_{1ik} \geq (Cm_{i(k-1)} + R).Y_{ik} \quad \forall i,k$ | (12) |
| $nk_i = \left( \sum_{j=1}^{N} \sum_{k=1}^{B} X_{j1ik} - 1 \right) \quad \forall i,j,k$ | (13) |
| $E_i = \delta_1.\left( \sum_{k=1}^{B} \sum_{p=1}^{P} Pr_{pik} \right) + \delta_2.(nk-1).R \quad \forall i$ | (14) |

Equation (1) represents the objective function, which minimizes the total consumed energy, and allows us to equilibrate charges between machines. Equation (2) and (3) ensure that a position of a bucket of a machine treats at most one job, and a job is treated by one and only one position. Equation (4) keeps no empty position between two loaded positions. Equations (5) and (6) defines bucket loaded in a machine and keeps no empty bucket between two loaded buckets. Equation (7) defines the actual processing time. Equations (8), (9), (10), (11) and (12) define the start and completion processing time in a position of a bucket of a machine. Equation (13) defines the number of buckets in a machine. Equation (14) defines the energy consumed by a machine.

### 3.2. Illustrative example

Table 1 resumes instances of the illustrative example, where 12 jobs are available to be processed in 2 parallel uniform machines. We used Lingo 18 as solver, it took more than 3hours to give the optimal solution, which is illustrated in figure 1.

**Table 1.** Illustrative example instances

| j | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| aj | 4 | 12 | 2 | 5 | 11 | 4 | 6 | 8 | 7 | 13 | 9 | 10 |
| i | 1 | | | | | | 2 | | | | | |
| Vi | 2 | | | | | | 1 | | | | | |
| RMA=2, α=0.5 | | | | | | | | | | | | |

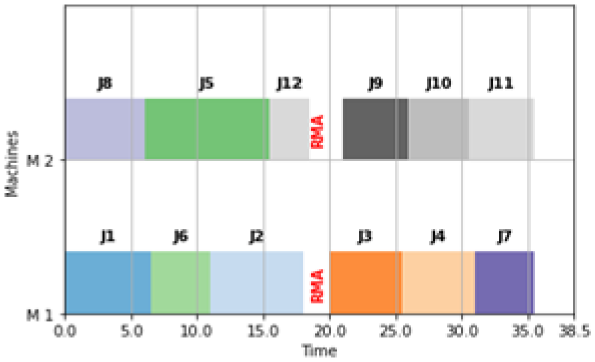


**Fig. 1.** Gantt diagram for the optimal solution

The OF=35.5, The maintenance activity in the first and the second machine is done after treating 3 jobs. we can remark that the solver affects the same number of jobs to each machine. Also, we remark that jobs with the largest processing time are affected to the speediest machine, and the reverse hold true. From this optimal solution, we infer some lemmas that are presented and proven in the next section.

### 3.3. Theoretical analysis

Observe that objective function depends only on the assignment of jobs to machine, no matter the sequence. Then, the objective function can be formulated as follow:

$$OF = \delta_1 . \sum_{i=1}^{M} \sum \frac{a_j}{V_i} + \delta_1 . \alpha . (p-1) + \delta_2 . \sum_{i=1}^{M} nk_i . R$$

**Lemma 1**: The first term $\sum \frac{a_j}{V_i}$ can be minimized by allocating the largest job to the speediest machine.

**Proof**

It's mathematically known that: if $a > a'$ then $a.x > a'.x$. so that $x > 0$.

If we put

$$a = a_j, \qquad a' = a'_j, x = \left(1 - \frac{1}{V_i}\right)$$

so $a_j$ is largest then $a'_j$, we obtain:

$$a_j . \left(1 - \frac{1}{V_i}\right) > a'_j . \left(1 - \frac{1}{V_i}\right) \rightarrow \left(a_j - \frac{1}{V_i}\right) > \left(a'_j - \frac{1}{V_i}\right)$$

The proportion:

$$\left(a_j - \frac{1}{V_i}\right)$$

represents the reduced time units when $V_i$. We can observe that the proportion reduced from $a_j$ is larger than the proportion reduced from $a'_j$. So, to minimize $\sum \frac{a_j}{V_i}$ largest jobs should be scheduled in the speediest machine.

**Lemma 2**: The second term can be minimized by allocating the same number of jobs to each machine.

**Proof**

From [27], the deterioration is minimized by allocating the same number of jobs to each machine.

**Lemma 3**: if in a bucket b: $\delta_2 R \leq \delta_1 . \alpha . (p-1)$, load more than one position in this bucket.

**Proof**

It's known that more positions are loaded in a bucket, less buckets are loaded in a machine. So, since the energy consumed due to deterioration in a given position of a bucket is inferior to energy consumed due to maintenance activity, it's more efficient to affect a job in the next position of this bucket than affect it to the next bucket.

## 4. Resolution Approach

Given the complexity of scheduling problems, the MILP doesn't provide solutions for medium and large instances. And since no paper studied this problem before, no polynomial time algorithm exists for the exact solution. Therefore, a meta-heuristic or a specific heuristic algorithm are necessary to solve large size instances of the problem.

Using the three proposed lemmas, we develop a heuristic; LBH heuristic to solve the problem.

### 4.1. LBH-Heuristic

The proposed heuristic proceeds by steps, where the first step follows lemma 1 and lemma 2, divides equivalently jobs to machines by affecting the largest jobs to the speediest machine. The second step uses lemma 3 to load buckets and affect jobs. This lemma-based heuristic is more detailed in algorithm 1.

The three lemmas are mathematically proven, and so is lemma 1. But it is proven to minimize only one term and not the objective function as a whole, and unaware the machines' multitude.

To deal with this, we integrate a simulated annealing metaheuristic, which permutates between two randomly chosen jobs, in order to improve the OF.

| | Algorithm 1: lemma-based heuristic "LBH" |
|---|---|
| | Initialize N=number of jobs j, M=number of machines i.<br>Initialize list of jobs $L_j$.<br>initialize $C_i$.: total completion time of machine i<br>initialize $M1_{ip}$.<br>Initialize $M2_{ikp}$ |
| 1. | Order job in decreasing order of $a_j$ in $L_j$. |
| 2. | Order job in decreasing order of $v_i$ |
| 3. | Initialize i=1, j=1 |
| 4. | Do {<br>Affect the [N/M] jobs from $L_j$ to $M1_{ip}$. |
| 5. | Update $i++,\ j=j+\left[\frac{N}{M}\right]$<br>} |
| 6. | while ($i < M$) . |
| 7. | If $j = N${<br>Go to step 9 |
| 8. | Else<br>Do {<br>$j++$<br>affect j to I' so I' is the machine with the smallest total completion time<br>}<br>while ($j < N$) |
| 9. | For $i = 1 \dots M$ {<br>Initialize p=1, p'=1, k=1 //p' position of bucket k |
| 10. | While $M1_{ip} > 0${<br>p'=1<br>While ($\delta_1.\alpha.(p-1) \leq \delta_2.R$){<br>Affect job from $M1_{ip}$. to $M2_{ikp}$,<br>$p'++$.<br>}<br>k+ +.<br>} |

### 4.2. LBH-SA heuristic

Simulated Annealing SA belongs to meta-heuristics based on solution modification. It is characterized by using one solution by iteration which reduces the number of parameters to fix and make it simpler to adapt to our system.

It is inspired by the physical process of metals cooling. The cooling process aims to order the atoms in most regular structure. For this, cooling rate has crucial impact on final structure.

Simulated Annealing is a finite number of iterations repeated under a decreasing temperature, algorithm 2.

It is noted that:

- Initial solution is disrupted to obtain new solutions

- A structure is a solution to the problem.

- After an iteration temperature is decreased using the following relation: $T_n = \alpha.T_{n-1}$

- Iterations stop when the stopping criterion is achieved.

| | Algorithm 2: LBH-SA heuristic |
|---|---|
| | INITIALIZE ($S_{LBH}$,$T_i$,$\alpha$,$P_a$ ,$\delta$) |
| | While ($T_n < \delta$): |
| i | Neighborhood generation:<br>Sn=mutation (Si) |
| ii | Fitness:<br>Calculate OF (Sn), and ΔS= OF (Sn) - OF (Si) |
| iii | Select Si for the next iteration: |
| | - If (ΔS<0) Then<br>$S_i = S_n$<br>- Else:<br>{ Calculate the probability<br>$P = e^{-\frac{\Delta S}{T_n}}$<br>- If (Pa>=P) then:<br>{Si=Sn accepted}<br>- Else { Si=Si rejected} |
| | Update ($T_i$) |

An initial structure and an initial temperature are introduced. Temperature needs to be high enough to allow flexibility to the algorithm, without slowing down the algorithm. The movement from a structure to another is done if the quality of the new structure is judged better than the quality of the previous one. The movement quality is judged through a proportion, named the acceptance probability or metropolis acceptance criterion. The acceptance probability at any given temperature is determined by the following equation:

$$P = \{e^{-\frac{\Delta S}{T}} \quad if\ \Delta S > 0 \quad 1 \qquad else$$

In order to use the simulated-annealing algorithm, we primarily need to well put the following parameters: initial solution Si, initial temperature Ti, acceptance probability Pa, cooling ratio α, stopping criterion δ. In addition, we need to specify a mutation scheme to generate neighborhood.

**Initial solution:** considers solutions given by the lemma-based heuristic

**Mutation "neighbourhood generation":** Since the objective function is calculated through the total flow time in machines, the sequence of jobs does not affect it. So, no permutation is made between two jobs of the same machine, a permutation is made between two jobs from two different machines, randomly chosen.

**Other parameters**: Ti=8000, α=0.99, Pa=70%. The stopping criterion is theoretically achieved when T=0, but in practice that may not be achieved. Thus, if there are L iterations with no improvement, the algorithm stops.

### 4.3. Computational experiment:

In this section, the designed algorithm is tested on small and medium instances, by comparing its results with optimal solutions of the MILP model. Parameters of the system are generated randomly, according to the following distributions:

- Normal processing time $a_j = U(5{,}20)$,
- Machine's speed $V_i = U(1{,}2)$,
- Maintenance duration $R_i = U(2{,}6)$,
- Position and time compression rate $\alpha, \delta_1, \delta_2 = U(0.25{,}1)$.

Table II shows the difference in objective functions by the developed LBH-SA and the MILP. The LBH-SA is programed in python and the mathematical model in Lingo 18.

Values in the column "Gap%" represent the percentage shift of solutions given by developed LBH-SA from the optimal solutions given by Lingo 18 solver of small instances, ranging from 6N2M to 12N3M. It is worth noting that, for each size, 10 different instances are generated. So, the minimum and maximum obtained gaps are mentioned in table 2.

**Table 2.** Lingo and SA algorithm results for small instances 20N4M

| Instances: | | Gaps% (MILP, LBH-SA) | | | Computational time | |
|---|---|---|---|---|---|---|
| Name | size | Min. | Max. | Avr. | SA | Lingo |
| I1 | 6N2M | 0 | 0 | 0 | ms | 1h40mn |
| I2 | 8N2M | 0 | 0 | 0 | ms | 1h50mn |
| I3 | 10N2M | 0 | 0.01 | 0.01 | Ms | 2h |
| I4 | 12N2M | 0 | 0.01 | 0.01 | Ms | 2h40 |
| I5 | 12N3M | 0 | 0.02 | 0.02 | Ms | 5h |
| I6 | 15N3M | 0.01 | 0.08 | 0.05 | ms | 5h |

Results of table 2 indicate that gaps between optimum and the proposed algorithm solutions doesn't exceed 0.08%. Bearing in mind that the solver requires several hours to provide an optimal solution, whereas the computation time of the proposed algorithm takes only milliseconds. This analysis allows us to conclude that the proposed BLH-SA algorithm delivers satisfactory results within an affordable computational time.

## 5. Conclusion

In this paper, an original mathematical model is developed for a uniform parallel machine scheduling problem under deterioration and maintenances activities to minimize the total consumed energy. Processing time of a job depends on machine's speed and position to which it is assigned, and this is due to machine deterioration. The deterioration effect is remedied by maintenance activities. The problem consists of affecting jobs to machine, deciding about the number of maintenance activities and their start and end time, in such way to minimize the total consumed energy.

## References

[1] K. Fang, N. Uhan, F. Zhao, et J. W. Sutherland, « A new approach to scheduling in manufacturing for power consumption and carbon footprint reduction », Journal of Manufacturing Systems, vol. 30, no 4, p. 234-240, oct. 2011, doi: 10.1016/j.jmsy.2011.08.004.

[2] Pechmann et I. Schöler, « Optimizing Energy Costs by Intelligent Production Scheduling », in Glocalized Solutions for Sustainability in Manufacturing, J. Hesselbach et C. Herrmann, Éd., Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, p. 293-298. doi: 10.1007/978-3-642-19692-8_51.

[3] M. Tigane, M. Dahane, et M. Boudhar, « Multiobjective approach for deteriorating jobs scheduling for a sustainable manufacturing system », The International Journal of Advanced Manufacturing Technology, nov. 2018, doi: 10.1007/s00170-018-3043-1.

[4] Bachman et A. Janiak, « Scheduling jobs with position-dependent processing times », Journal of the

Operational Research Society, vol. 55, no 3, p. 257-264, mars 2004, doi: 10.1057/palgrave.jors.2601689.

[5] J. Kaabi et Y. Harrath, « Scheduling on uniform parallel machines with periodic unavailability constraints », International Journal of Production Research, vol. 57, no 1, p. 216-227, janv. 2019, doi: 10.1080/00207543.2018.1471242.

[6] H.-J. Kim et J.-H. Lee, « Scheduling uniform parallel dedicated machines with job splitting, sequence-dependent setup times, and multiple servers », Computers & Operations Research, vol. 126, p. 105115, févr. 2021, doi: 10.1016/j.cor.2020.105115.

[7] D. Chung et B. S. Kim, « A hybrid genetic algorithm with two-stage dispatching heuristic for a machine scheduling problem with step-deteriorating jobs and rate-modifying activities », Computers & Industrial Engineering, vol. 98, p. 113-124, août 2016, doi: 10.1016/j.cie.2016.05.028.

[8] S. Browne et U. Yechiali, « SCHEDULING DETERIORATING JOBS ON A SINGLE PROCESSOR », new york, USA, mars 1989.

[9] Alidaee et N. Womer, « Scheduling with time dependent processing times: Review and extensions », Journal of the Operational Research Society, Mississipi USA, 1999.

[10] X. Wu, P. Guo, Y. Wang, et Y. Wang, « Decomposition approaches for parallel machine scheduling of step-deteriorating jobs to minimize total tardiness and energy consumption », Complex Intell. Syst., vol. 8, no 2, p. 1339-1354, avr. 2022, doi: 10.1007/s40747-021-00601-9.

[11] M. M. Mazdeh, F. Zaerpour, A. Zareei, et A. Hajinezhad, « Parallel machines scheduling to minimize job tardiness and machine deteriorating cost with deteriorating jobs », Applied Mathematical Modelling, vol. 34, no 6, p. 1498-1510, juin 2010, doi: 10.1016/j.apm.2009.08.023.

[12] X. Huang et M.-Z. Wang, « Parallel identical machines scheduling with deteriorating jobs and total absolute differences penalties », Applied Mathematical Modelling, vol. 35, no 3, p. 1349-1353, mars 2011, doi: 10.1016/j.apm.2010.09.013.

[13] M. Liu, F. Zheng, S. Wang, et Y. Xu, « Approximation algorithms for parallel machine scheduling with linear deterioration », Theoretical Computer Science, vol. 497, p. 108-111, juill. 2013, doi: 10.1016/j.tcs.2012.01.020.

[14] Y. Ouazene et F. Yalaoui, « Identical parallel machine scheduling with time-dependent processing times », Theoretical Computer Science, déc. 2017, doi: 10.1016/j.tcs.2017.12.001.

[15] J. Ruiz-Torres, G. Paletta, et E. Pérez, « Parallel machine scheduling to minimize the makespan with sequence dependent deteriorating effects », Computers & Operations Research, vol. 40, no 8, p. 2051-2061, août 2013, doi: 10.1016/j.cor.2013.02.018.

[16] S. Norelhouda, B. Faycal, et S. Lamia, « Parallel machines scheduling problem with skilled operators in a potery handicraft firm », in 2020 IEEE 13th International Colloquium of Logistics and Supply Chain Management (LOGISTIQUA), Fez, Morocco: IEEE, déc. 2020, p. 1-7. doi: 10.1109/LOGISTIQUA49782.2020.9353923.

[17] C.-Y. Lee et V. J. Leon, « Machine scheduling with a rate-modifying activity », European Journal of Operational Research, vol. 128, no 1, p. 119-128, janv. 2001, doi: 10.1016/S0377-2217(99)00066-1.

[18] G. Mosheiov et A. Sarig, « Scheduling a maintenance activity and due-window assignment on a single machine », Computers & Operations Research, vol. 36, no 9, p. 2541-2545, sept. 2009, doi: 10.1016/j.cor.2008.10.007.

[19] C.-L. Zhao, H.-Y. Tang, et C.-D. Cheng, « Two-parallel machines scheduling with rate-modifying activities to minimize total completion time », European Journal of Operational Research, vol. 198, no 1, p. 354-357, oct. 2009, doi: 10.1016/j.ejor.2008.08.012.

[20] Zhao et H. Tang, « Single machine scheduling with general job-dependent aging effect and maintenance activities to minimize makespan », Applied Mathematical Modelling, vol. 34, no 3, p. 837-841, mars 2010, doi: 10.1016/j.apm.2009.07.002.

[21] K. Rustogi et V. A. Strusevich, « Single machine scheduling with general positional deterioration and rate-modifying maintenance », Omega, vol. 40, no 6, p. 791-804, déc. 2012, doi: 10.1016/j.omega.2011.12.007.

[22] M. Ghaleb, S. Taghipour, M. Sharifi, et H. Zolfagharinia, « Integrated production and maintenance scheduling for a single degrading machine with deterioration-based failures », Computers & Industrial Engineering, vol. 143, p. 106432, mai 2020, doi: 10.1016/j.cie.2020.106432.

[23] D.-L. Yang, T. C. E. Cheng, et S.-J. Yang, « Parallel-machine scheduling with controllable processing times and rate-modifying activities to minimise total cost involving total completion time and job compressions », International Journal of Production

Research, vol. 52, no 4, p. 1133-1141, févr. 2014, doi: 10.1080/00207543.2013.841330.

[24] C.-J. Hsu, T. C. E. Cheng, et D.-L. Yang, « Unrelated parallel-machine scheduling with rate-modifying activities to minimize the total completion time », Information Sciences, vol. 181, no 20, p. 4799-4803, oct. 2011, doi: 10.1016/j.ins.2011.06.010.

[25] M. Geurtsen, J. Adan, et A. Akçay, « Integrated maintenance and production scheduling for unrelated parallel machiness with setup times », Flex Serv Manuf J, oct. 2023, doi: 10.1007/s10696-023-09511-z.

[26] T. C. E. Cheng, C.-J. Hsu, et D.-L. Yang, « Unrelated parallel-machine scheduling with deteriorating maintenance activities », Computers & Industrial Engineering, vol. 60, no 4, p. 602-605, mai 2011, doi: 10.1016/j.cie.2010.12.017.

[27] S.-J. Yang et D.-L. Yang, « Minimizing the makespan on single-machine scheduling with aging effect and variable maintenance activities », Omega, vol. 38, no 6, p. 528-533, déc. 2010, doi: 10.1016/j.omega.2010.01.003.

[28] K. Rustogi et V. A. Strusevich, « Single machine scheduling with a generalized job-dependent cumulative effect », Journal of Scheduling, vol. 20, no 6, p. 583-592, déc. 2017, doi: 10.1007/s10951-016-0497-6.

[29] N. Sekkal et F. Belkaid, « A multi-objective simulated annealing to solve an identical parallel machine scheduling problem with deterioration effect and resources consumption constraints », J Comb Optim, vol. 40, no 3, p. 660-696, oct. 2020, doi: 10.1007/s10878-020-00607-y.