Management and Production Engineering Review
Volume 11 • Number 4 • December 2020 • pp. 138–148
DOI: 10.24425/mper.2020.136128

A GENETIC ALGORITHM AND B&B ALGORITHM FOR INTEGRATED PRODUCTION SCHEDULING, PREVENTIVE AND CORRECTIVE MAINTENANCE TO SAVE ENERGY

Sadiqi Assia[1], El Abbassi Ikram[2], El Barkany Abdellah[1], Darcherif Moumen[2], El Biyaali Ahmed[1]

[1] *Mechanical Engineering Laboratory, Faculty of Science and Techniques -USMBA- Route d'Imouzzer – Fez, Morocco*
[2] *ECAM-EPMI, Research Laboratory in Industrial Eco-innovation and Energetics Quartz-Lab, Cergy-Pontoise, France*

Corresponding author:
Sadiqi Assia
Mechanical Engineering Laboratory
Faculty of Science and Techniques Sidi Mohammed Ben Abdellah University
B.P. 2202 – Route d'Imouzzer – Fez, Morocco
phone: +212 605820229
e-mail: sadiqiassia@gmail.com

Received: 4 July 2019
Accepted: 26 October 2020

Abstract
The rapid global economic development of the world economy depends on the availability of substantial energy and resources, which is why in recent years a large share of non-renewable energy resources has attracted interest in energy control. In addition, inappropriate use of energy resources raises the serious problem of inadequate emissions of greenhouse effect gases, with major impact on the environment and climate. On the other hand, it is important to ensure efficient energy consumption in order to stimulate economic development and preserve the environment. As scheduling conflicts in the different workshops are closely associated with energy consumption. However, we find in the literature only a brief work strictly focused on two directions of research: the scheduling with PM and the scheduling with energy. Moreover, our objective is to combine both aspects and directions of in-depth research in a single machine. In this context, this article addresses the problem of integrated scheduling of production, preventive maintenance (PM) and corrective maintenance (CM) jobs in a single machine. The objective of this article is to minimize total energy consumption under the constraints of system robustness and stability. A common model for the integration of preventive maintenance (PM) in production scheduling is proposed, where the sequence of production tasks, as well as the preventive maintenance (PM) periods and the expected times for completion of the tasks are established simultaneously; this makes the theory put into practice more efficient. On the basis of the exact Branch and Bound method integrated on the CPLEX solver and the genetic algorithm (GA) solved in the Python software, the performance of the proposed integer binary mixed programming model is tested and evaluated. Indeed, after numerically experimenting with various parameters of the problem, the B&B algorithm works relatively satisfactorily and provides accurate results compared to the GA algorithm. A comparative study of the results proved that the model developed was sufficiently efficient.

Keywords
Scheduling, maintenance, Genetic Algorithm, Branch and Bound, MILP, modeling, optimization, CPLEX, Python.

Introduction

Nowadays, as a result of the accelerated progress of the economy, population growth and globalization, the energy production sector is experiencing a permanent and rapid 30-year increase of 56% between 2010 and 2040 [1]. The industrial sector, including manufacturing, still accounts for nearly 50 percent of total global energy consumption. More specifically, total energy consumption in the industrial sector will increase from 58.9 PWh in 2010 to 90.4 PWh in 2040 [1]. As for China, the industrial world consumes about 69.44% of the total energy consumption in 2014, of which 82.9% comes from the manufacturing sector [2]. The total energy consumption in China's manufacturing sector is 295,686.4 units in 2014, with 10,000 tons of standard coal equivalent (SCE) per unit [3].

One of the main causes of environmental pollution is the rapid growth of energy expenditure due to the increasing amount of greenhouse pollutants, especially carbon dioxide (CO2) [4]. For example, in the United States, the rate of greenhouse gas emissions is almost 28% [5], while in Germany, it is in the range of 18–20% [6] and energy consumption in the manufacturing sector accounts for at least 26% of total CO2 emissions in China [7], especially in energy-intensive industries such as mold, chemicals, glass and petroleum products. For example, the ferrous metal smelting and pressing industries (i.e. mainly for injection molding and stamping processes) account for about 28.8% of the total energy consumption of the manufacturing sector in 2014 [2]. Therefore, there is a very urgent need to improve the efficiency of energy use in the manufacturing sector, to reduce energy consumption and to enhance the reduction of greenhouse gas emissions [8–10]. In addition, producing economically and in a timely manner is growing more and more important in today's competitive business and global environments. Against this background, it is essential for many manufacturers that production is optimized through efficient and operationally stabilized planning. Conventional scheduling documentation requires the assumption that machines are accessible at all times. Nevertheless, these machines and equipment are often inaccessible from the planning phase for various factors [11, 12, 20], especially failures and maintenance schedules in traditional industrial environments. All of the above scheduling problems are complicated by this availability consideration.

The production environment in continuous planning is still susceptible to degradation depending on utilization, reducing machine reliability and affecting the stability of the machine and equipment system. The effectiveness of predictive maintainability in the industry of today extends to the preventive maintenance of machines and systems. In order to maintain a machine, it must be serviced after it has been in permanent use for a certain amount of time. Accordingly, an extended programming horizon must include a number of maintenance periods [13].

Considering that both production and maintenance tasks require machine uptime, PM tasks should be scheduled at precisely the same time with a view to enhancing overall system productivity and performance. Other than programmed PM, a deterministic aspect affecting machine utilization, some unexpected downtime (e.g., failures, job reversal, incoming or changing deadlines) can quickly disrupt the schedule. Many production processes in many manufacturing facilities have an initial production calendar established to guide workshop operations during a specific period of downtime. Once an unanticipated interruption arrives, all or part of the original calendar schedule is adjusted accordingly to maintain the practicability and performance of the originally proposed calendar. When the planning period comes to a close, a schedule effectively implemented in the program at the workshop is referred to as the accomplished schedule [14]. When considering the interruption of machine failures in this paper, the interruption of machine failures is considered.

It is a question of robustness and stability [14]:

- To be robust, it is necessary to have a stable performance when something unforeseen occurs.
- It is stable a program whose realized schedule does not deviate substantially from the initial one because of disruptions and revisions.

The contribution of this paper is to address the problem of integrated production scheduling and preventive and corrective maintenance to minimize the total energy consumption using several exact and approximate methods: the B&B Exact Method and the GA approximate method to provide results of a comparative study of the two methods.

The discussion in the paper as described below. In the next section, a detailed description and modeling of the problem under consideration and formulate an improved MILP model and the results of the computation of the problem using the two methods of resolution are provided. Finally, the conclusion is drawn.

Problem description and modelling

The purpose of this section is to propose a solution to the problem of fully scheduling production, preventive (PM) and corrective (CM) maintenance tasks in the single-machine workshop. In order to minimize the total energy consumption.

By adding energy constraints on the existing model of the literature that does not address the energy aspect in their study [13, 20]. In this context, a common model for integrating production scheduling and preventive and corrective maintenance to optimize total energy consumption is proposed. Our model takes into account the different machine downtime constraints and performance measures mentioned above. The indices, parameters and variables used in this formulation are given in Table 1.

Table 1
The notation used to formulate the problem.

Indices:	
i, j	Indices of jobs
Sets:	
J	Set of jobs; $J = \{J_1, J_2, ..., J_n\}$
Parametres:	
n	Number of jobs
T_j	Processing time of job j
P_j	Processing power of job j
P_0	The common power, which is consumed by auxiliary equipment and facilities
$C_{\max}$	The makespan, which equals to the maximum completion time of all jobs
PM	Time required to perform PM on the machine
CM	Time required to repair the machine of machine
B	Shape parameter of failure function of machine
α	Scale parameter of failure function of machine
PEC	The processing energy consumption of job j
CEC	The common energy consumption of auxiliary facilities in the workshop
TEC	The total energy consumption in the workshop
Decision variables:	
$x_{i,j}$	$\begin{cases} 1 & \text{if job } j\text{-th performed is job } i \\ 0 & \text{Otherwise} \end{cases}$
Y_i	$\begin{cases} 1 & \text{if PM is performed prior to the } i\text{-th job} \\ 0 & \text{Otherwise} \end{cases}$
j_i	The i-th job in the sequence
T_i	The processing time of the i-th job
N_i	The number of breakdowns from the start time of j_i to the finish time of j_i, which is a discrete random variable
C_i	The initially planned completion time of the i-th job
C_i^r	The realized completion time of the i-th job

Problem assumptions and formulation

This article analyzes the problem of scheduling n jobs $\{j_1, j_2, ..., j_n\}$ executed on a single machine:

- they are available at time zero and do not allow any preemption,
- let's assume that when the machine processing the jobs is affected by a breakdown to schedule preventive and corrective maintenance periods in the model.

Since the frequency of malfunctions can be predicted for a given solution, the B&B and GA methods have been adopted in this document to manage this uncertainty factor and minimize total energy consumption. First, an initial plan is established at the beginning of the scheduling horizon. A complete solution can be subdivided into three parts (X, Y, T)

- first, the list of work sequences X,
- the second is the matrix of the PM, Y positions,
- the third is the time frame for the i-th matrix of T jobs.

The first example. It is thus possible to program in a single machine the realization of 4 jobs. For this problem, the Gantt diagram of the initial scheduling where the list of succession of jobs is defined by $\mathbf{X} = \{1, 2, 3, 4\}$, where the matrix of PM positions corresponds to $\mathbf{Y} = \{0, 0, 0, 1\}$, on the basis of a Gantt diagram and where the duration of the i-th matrix of jobs p is defined by $\mathbf{T} = \{T_1, T_2, T_3, T_4\}$

- $x_{i,j}$: designates the j-th job performed is the job i,
- Y_i: means that a PM is performed just before the i-th job performed in the machine. On the other hand, a PM is performed before the idle time,
- T_i: means the processing time of the i-th job.

In order to absorb the unexpected uncertainty of moving to the line, a dynamic schedule of downtime is inserted in the calendar. An example of downtime in the calendar is shown in Fig. 1, some downtimes do not appear in the calendar if the unplanned downtime is completely avoided.

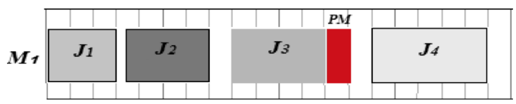

Fig. 1. Gant chart of Initial plan.

In other words, the expected beginning of the works is not completely determined by the schedule, and is severely affected by the entered downtimes. After the initial design has been drawn up at the commencement, job sequences and PM Positions will not be changed for the complete period of the design horizon.

The start times of the realized activities will be changed, nevertheless. For instance, when a breakdown takes place during J_3 as indicated in Fig. 2, a CM is executed. The operations J_2 may be delayed. And the operations J_3 and PM will not be delayed because of the occurrence of downtime. This rescheduling approach is quite feasible in practice since the initial schedule is used as a planning basis for the planning of such external activities as changing tools and procurement of equipment. If the sequence of completed jobs is different from the sequence initially scheduled, for example, many problems will occur in the material procurement system, because the originally prepared equipment has to be taken out of production and newly acquired equipment has to be inserted. In addition, if the corresponding equipment is not ready, a job cannot be started.

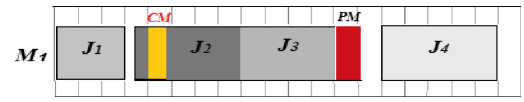

Fig. 2. Gantt chart of Realized Schedule one CM period.

The 2nd example. From the first sample the identical information given in the first example will be taken, however, this again with a CM value corresponding to j_3, as described in Fig. 4.

In order to absorb the fluctuating and unpredictable instability of the right, periods of inactivity are foreseen. As shown in Fig. 3, there are periods of inactivity that are not included in programs if unexpected failures are neglected, as shown in the first example. The sequence of jobs and the locations of the maintenance periods will not change after the initial plan has been drawn up throughout the scheduling period. However, for the activities, the realized start times will be changed. Thus, if a break occurs during J_3 as shown in Fig. 4, then a CM occurs during J_3 is executed. Tasks J_3 and PM must be postponed. In addition, task J_4 is not postponed due to the presence of idle time or rest time. In practice, this rescheduling is very reasonable since the initial schedule is used as a basis for scheduling externally related activities, such as tool changes and procurement of supplies. For example, when the sequence of tasks performed is different from the one initially envisaged, this leads to many dysfunctions in production management, particularly with regard to the procurement of materials, since previously prepared materials must be taken out of the production line and new materials must be introduced. Similarly, it is impossible to start a job if the equipment concerned is not ready, as shown in the first example.

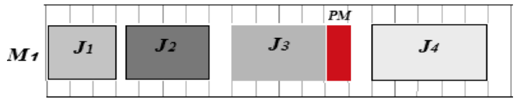

Fig. 3. Gant chart of initial plan.

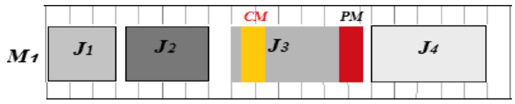

Fig. 4. Gantt chart of realized schedule one CM period.

Objective function

The target function minimizes as much as possible the total energy expenditure or consumption (TEC) of the single-machine workshop, i.e., equipment energy expenditure and general energy expenditure [16–18]. In this context, the term "total energy consumption" is therefore the sum of the total process energy expenditure and the current energy expenditure, estimated according to the following equation:

$$\mathrm{TEC} = \mathrm{PEC} + \mathrm{CEC}. \tag{1}$$

Energy consumption of the production process:

$$\mathrm{PEC} = \sum_{i=1}^{n} \sum_{j=1}^{n} P_j.T_j.x_{i,j}. \tag{2}$$

Collective energy expenditure:

$$\mathrm{CEC} = P_0.C_{\max}. \tag{3}$$

In this context, the concept of Energy Demand Management (TEC) is defined as the total process energy demand and current energy consumption, determined as follows:

$$\mathrm{TEC} = \sum_{i=1}^{n} \sum_{j=1}^{n} P_j.T_j.x_{i,j} + P_0.C_{\max}. \tag{4}$$

Within this context, it is expected to minimize the total energy expenditure:

$$\textbf{Fct Obj} = \mathrm{TEC}, \tag{5}$$

$$\textbf{Fct Obj} = \mathrm{PEC} + \mathrm{CEC}, \tag{6}$$

$$\textbf{Fct Obj} = \sum_{i=1}^{n} \sum_{j=1}^{n} P_j.T_j.x_{i,j} + P_0.C_{\max}. \tag{7}$$

Modelling of the problem

To definitively establish within this approach that the Mixed Integer Linear Programming Model:

$$\textbf{Minimize}\text{Fct Obj}, \tag{8}$$

S.t

$$C_1 \geq \text{PM}.Y_1 + T_1, \tag{9}$$

$$C_i \geq C_{i-1} + \text{CM}.Y_i + T_i, \quad i = 2, 3, ..., n, \tag{10}$$

$$T_j = \sum_{i=1}^{n} x_{i,j}.T_i, \quad j = 1, 2, ..., n, \tag{11}$$

$$C_1^r = \text{PM}.Y_1 + T_1 + \text{CM}.N_1, \tag{12}$$

$$C_i^r = C_{i-1}^r + \text{PM}.Y_i + T_i + \text{CM}.N_i, \quad i = 2, 3, ..., n, \tag{13}$$

$$\text{Robustness} = E.\left(\sum_{i=1}^{n} C_i^r\right), \tag{14}$$

$$\text{Stability} = E.\left(\sum_{i=1}^{n} |C_i - C_i^r|\right), \tag{15}$$

$$C_{\max} = C_n^r, \quad i = 1, 2, ..., n, \tag{16}$$

$$\sum_{j=1}^{n} x_{i,j} = 1, \quad i = 1, 2, ..., n, \tag{17}$$

$$\sum_{i=1}^{n} x_{i,j} = 1, \quad j = 1, 2, ..., n, \tag{18}$$

$$x_{i,j} \text{ binary}; \; i = 1, 2, ..., n, \quad j = 1, 2, ..., n. \tag{19}$$

The target function (8) focuses on minimizing all the energy consumed (TEC). The constraints (9) and (10) build the relationship from task i-th to $(i+1)$th job. Constraint (11) indicates processing time for task $(i+1)$. Mates (12) and (13) specify the real processing time of i-th job.

Contrasts (14) and (15) set constraints for System Resilience (R) and Stability (S), separately. In terms of robustness, the real concern is the achieved scheduling behavior performance as opposed to the initial scheduling organization performance. Therefore, an expected scheduling efficiency of system E. $\left(\sum_{i=1}^{n} C_i^r\right)$ is used to evaluate the reliability of an existing system. For stability, the actual planning must deviate from the original bare minimal scheduling. Therefore, the sum of absolute differences in absolute job completion timings is used as a stable measure E. $\left(\sum_{i=1}^{n} |C_i - C_i^r|\right)$.

Constraint (16) describes the job execution rate. Constraints (17) and (18) guarantee only one position for the sequence and that a single job can be positioned at only one position in the succession, and only one job can be positioned at each position, respectively. Constraints (19) and (20) establish the binary limitations for $x_{i,j}$ and Y_i separately.

Branch and bound algorithm

Among all popular approaches to solve programming problems with integers, the branch-and-bound algorithms are proposed. In general, this algorithm essentially follows two phases. In the first one, it consists in separating a series of problems into subsets; whereas in the second phase, it consists in evaluating the solutions of a subset by valuing the best solution of this subset. The research process stops when there are no more parts left to explore in the solution search area, and the best solution is then recognized as the best current solution [19]. This procedure is considered as a downward derivation, as shown in Fig. 5:

- the branch-and-bound methods are techniques based on a "smart" inventory of admissible choices for a problem of combinatorial automation;
- concept: Demonstrate the solution's optimization by dividing the solution space;
- "Split and rule";
- linear programming with integer numbers: using all the linear programming power to determine good limits;
- linear expansion of a linear program is called linear relaxation in program generated by removing the numbers in the table;
- completeness constraints on variables.

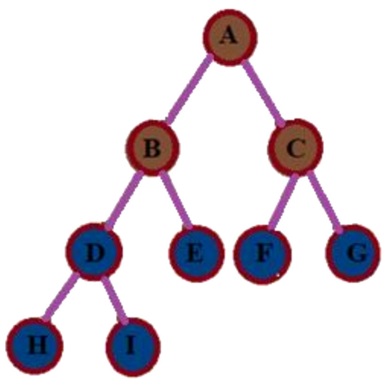

Fig. 5. Branch and bound procedure.

Approximate resolution through (AG)

Genetic algorithms (AGs) represent methods of stochastic research that enable to overcome a large variety of problems in the field of combinatorial algorithms. They are based on the genetic biological mechanism of breeding and selection. AGs are designed to provide the sustainability of the best and most prospective instincts, enabling to search successfully for a next higher generation of genes with a superior cost performance. They are very efficient in their simplicity, known performance in the discovery process and effective even for problems of an ever increasing complexity. Following the basic concept of AGs, a population of solutions is simulated to simulate the development process. Starting from the basic principle of AGs, the change process starts with a genotype consisting of one or several individual chromosomes, where each individual is equipped with a chromosome genotype. The chromosomes are made up of a set of elements, known as genes, which may assume several properties, referred to as "allegations". In such so-called evolutionary algorithms, it is necessary to use the following three fundamental operators: selection, which eliminates the solutions that are unlikely to be the most promising. In addition, to implement a genetic algorithm, it is necessary to have four pieces of data that practically correspond to the size of the population, the probability of crossing, the probability of mutation and the total number of generations.

Procedure of genetic algorithms

Genetic algorithm begins with a stage referred to as generation in which an initial *Pop_Size* population of individuals is created. For each individual generated, an individual fitness function is computed in order to define the adaptation level of the selection process. The individuals progress through the crossing application with a probability P_c. Afterwards; the resulting children are inverted at the gene level at a probability of P_m mutation. These three evolution stages allow with a big possibility to generate a new population that is better than the preceding population, as shown in Figs 6 and 7.

Start

Generate the initial population

Compute fitness

Repeat

- Selection
- Crossover
- Mutation
- Compute fitness

Until population has converged

Stop

Fig. 6. Pseudocode of General Genetic

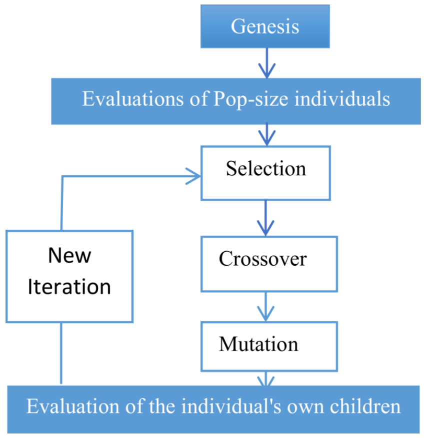

Fig. 7. General Genetic Algorithm Mechanism.

Each new generation, the populations increase and a cycle are repeated as long as the assessment estimates that a solution is not optimal already. The general process of genetic investigation is illustrated in Fig. 7. The principal actors will be discussed, in more detail, in subsequent parts of this document.

Computational results

This section is devoted to the exact analysis of the performance of the linear model corresponding to the system under study. However, the mathematical model represents a linear programming for which an exact method such as Branch and bound integrated in the CPLEX commercial solver of linear programming, it is necessary to qualify the complexity and optimality state of the problem considered. In order to evaluate the computational power of the proposed model, the problem should be examined in its most difficult cases; in fact, 30 different instances for each fixed dimension of n jobs in single machine have been generated. Moreover, in the experimental design adopted for the problem considered, the operating times were generated uniformly in the interval [1, 100]. According to [21–24], the generation of operating times between 1 and 100 is based on two reasons: The first reason is related to the historical uniformity, that is to say that the majority of the research work relating to the linear model calculation tests, including scheduling problems, generate the operating times from a uniform distribution in the interval [1, 100]; The second reason is related to the fact that it is preferable to use data representative of the real problems of scheduling since the generation of small intervals will certainly lead very easily to optimal solutions. These solutions will not necessarily be realistic because of the inadequate conclusions that may result. Table 2 reports the average computation times obtained using the CPLEX 12.6 software to find the optimal total energy consumption of the integrated scheduling problems of production and maintenance jobs. To gain more insight into the capability of the above model, comparative studies of average computation times between small and large problems are performed as shown in Figs 8–10. For all instances, the common power P_0 is set to 8. The processing powers P are derived from the uniform distribution [2, 8]. The preventive maintenance time of the PM machine and the corrective maintenance time of the CM machine are generated randomly from the set $\{5, 9, 12\}$ and $\{9, 12, 15\}$ respectively.

Tables 3 show the average gap between GA and B&B in CPLEX, calculated as

$$\frac{(\text{TEC}\,(\text{GA}) - \text{TEC}\,(\text{B\& B}))}{\text{TEC}\,(\text{B\& B})} \times 100\%.$$

TEC (B& B) is the optimal solution or lower bound found by CPLEX in 3600 seconds. The HIIGA was run 10 times for each instance, and TEC (GA) is the average value of solutions.

All MILP formulations are modeled using IBM ILOG CPLEX12.6 and the OPL language. The 30 instances are resolved on an HP 4300U notebook with an Intel Core i5 Duo processor clocked at 2.50 GHz and 8 GB of RAM. The time limit is 300 seconds. In other words, the analyzes are completed after 300 seconds. If no optimal solution is obtained within 300 seconds, the best current solution is returned.

Based on the results of the literature search, no article considers the integrated planning of production, preventive maintenance (PM) and corrective maintenance (CM) tasks in a single machine with minimal total energy consumption. For this reason, a comparative study with a second method approached from the genetic algorithm approach is approached, in order to concretize the results and generate the best solution concerning our problem in order to minimize the total energy consumption.

Table 2
Summary of calculation results.

Jobs		PM	CM	T	P_0	P	$C_{\max}$	TEC	Time [s]
5	Inst. 1	5	9	[10,50[	8	[2,4[	110	1140	0,14
	Inst. 2	9	12	[50,70[	8	[4,6[	270	3350	0,17
	Inst. 3	12	15	[70,100]	8	[6,8]	410	5960	0,18
7	Inst. 1	5	9	[10,50[	8	[2,4[	150	1520	0,14
	Inst. 2	9	12	[50,70[	8	[4,6[	380	4720	0,16
	Inst. 3	12	15	[70,100]	8	[6,8]	600	8620	0,19
10	Inst. 1	5	9	[10,50[	8	[2,4[	230	2340	0,17
	Inst. 2	9	12	[50,70[	8	[4,6[	550	6810	0,2
	Inst. 3	12	15	[70,100]	8	[6,8]	840	11980	0,21
15	Inst. 1	5	9	[10,50[	8	[2,4[	350	3540	0,17
	Inst. 2	9	12	[50,70[	8	[4,6[	830	10270	0,17
	Inst. 3	12	15	[70,100]	8	[6,8]	1270	18000	0,21
20	Inst. 1	5	9	[10,50[	8	[2,4[	470	4740	0,18
	Inst. 2	9	12	[50,70[	8	[4,6[	1110	13730	0,19
	Inst. 3	12	15	[70,100]	8	[6,8]	1700	24020	0,23
40	Inst. 1	5	9	[10,50[	8	[2,4[	940	9480	0,40
	Inst. 2	9	12	[50,70[	8	[4,6[	2220	27460	0,45
	Inst. 3	12	15	[70,100]	8	[6,8]	3400	48040	0,49
80	Inst. 1	5	9	[10,50[	8	[2,4[	1880	18960	10,5
	Inst. 2	9	12	[50,70[	8	[4,6[	4440	54920	11,52
	Inst. 3	12	15	[70,100]	8	[6,8]	6800	96080	12,61
100	Inst. 1	5	9	[10,50[	8	[2,4[	2350	23700	27,96
	Inst. 2	9	12	[50,70[	8	[4,6[	5550	68650	39,76
	Inst. 3	12	15	[70,100]	8	[6,8]	8500	120100	49,87
200	Inst. 1	5	9	[10,50[	8	[2,4[	4700	47400	146,87
	Inst. 2	9	12	[50,70[	8	[4,6[	11100	137299	149,05
	Inst. 3	12	15	[70,100]	8	[6,8]	17000	240199	158,1
400	Inst. 1	5	9	[10,50[	8	[2,4[	Out of memory		245,87
	Inst. 2	9	12	[50,70[	8	[4,6[			249,04
	Inst. 3	12	15	[70,100]	8	[6,8]			258,05

Table 3
Average gap between GA and B&B.

Jobs		PM	CM	T	P_0	P	TEC(GA)	TEC(B&B)	Gap [%]
5	Inst. 1	5	9	[10,50[	8	[2,4[	1148	1140	0,7017
	Inst. 2	9	12	[50,70[	8	[4,6[	3364	3350	0,4179
	Inst. 3	12	15	[70,100]	8	[6,8]	5972	5960	0,2013
7	Inst. 1	5	9	[10,50[	8	[2,4[	1532	1520	0,7894
	Inst. 2	9	12	[50,70[	8	[4,6[	4732	4720	0,1617
	Inst. 3	12	15	[70,100]	8	[6,8]	8633	8620	0,1508
10	Inst. 1	5	9	[10,50[	8	[2,4[	2341	2340	0,0427
	Inst. 2	9	12	[50,70[	8	[4,6[	6822	6810	0,1762
	Inst. 3	12	15	[70,100]	8	[6,8]	11998	11980	0,1502
15	Inst. 1	5	9	[10,50[	8	[2,4[	3543	3540	0,0847
	Inst. 2	9	12	[50,70[	8	[4,6[	10274	10270	0,0389
	Inst. 3	12	15	[70,100]	8	[6,8]	18010	18000	0,0555
20	Inst. 1	5	9	[10,50[	8	[2,4[	4745	4740	0,1054
	Inst. 2	9	12	[50,70[	8	[4,6[	13760	13730	0,2184
	Inst. 3	12	15	[70,100]	8	[6,8]	24067	24020	0,1956
40	Inst. 1	5	9	[10,50[	8	[2,4[	9499	9480	0,2004
	Inst. 2	9	12	[50,70[	8	[4,6[	27477	27460	0,0619
	Inst. 3	12	15	[70,100]	8	[6,8]	48053	48040	0,0270
80	Inst. 1	5	9	[10,50[	8	[2,4[	18973	18960	0,0685
	Inst. 2	9	12	[50,70[	8	[4,6[	54931	54920	0,0200
	Inst. 3	12	15	[70,100]	8	[6,8]	96098	96080	0,0187
100	Inst. 1	5	9	[10,50[	8	[2,4[	23721	23700	0,0886
	Inst. 2	9	12	[50,70[	8	[4,6[	68683	68650	0,0480
	Inst. 3	12	15	[70,100]	8	[6,8]	120210	120100	0,0915
200	Inst. 1	5	9	[10,50[	8	[2,4[	47428	47400	0,0590
	Inst. 2	9	12	[50,70[	8	[4,6[	137319	137299	0,0145
	Inst. 3	12	15	[70,100]	8	[6,8]	240238	240199	0,0162

The best solution can be found in a very short time thanks to our MILP models. Even better, in a reasonable time. However, the MILPs models require an excessive amount of time for a big problem. Therefore, a genetic algorithm approach (GA) has been used to solve large problems.

Compared to the various problems treated, the average calculation times are relatively reasonable for small problems. As soon as the size of the problem is larger. Calculation times are becoming more important, as shown in Fig. 8.

As shown in Figs 9 and 10, the average calculation times are relatively reasonable (less than 30 seconds) when the problem is less than 20 jobs. As soon as the size exceeds 20 jobs, computing times become very important. For example, for the third instance of the problem (100 jobs) with the value of the preventive maintenance time PM is 12, the corrective maintenance time CM is 15, the task processing times have been generated uniformly in the interval [70, 100] in this case, the processing powers P are derived from the uniform distribution [6, 8], followed by the common power is 8, for this case the results are obtained for a realization time of 49.87 seconds. Similarly, for the third instance of the problem (400 jobs), the completion time is 258.05 seconds. To further illustrate the impact of TEC on $C_{\max}$, a comparative study of the evolution of total energy consumption and Makespan as a function of average execution times is established for small and large problems as shown in Fig. 11. In addition, because the TEC values are much larger than the $C_{\max}$ values, a better illustration of the $C_{\max}$ values is shown in Fig. 12.

From Fig. 11, it can be seen that the variation in energy consumption depends on the type of workshop studied and the number of jobs to be performed. And as shown in Fig. 11, if the processing power of the jobs is increased, a considerable increase of the consumed energy is considered, if for example the change from the first to the second case for the problem to (80 jobs), For P is determined from the uniform distribution [2, 4], the obtained TEC value is 18960 and the second case for the processing power of all jobs P is determined from the uniform distribu-

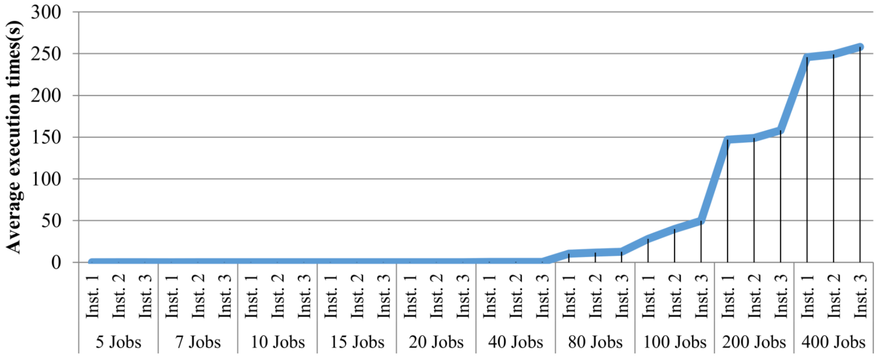

Fig. 8. Average execution times for different instances.

Fig. 9. Comparison of average execution times for SP.

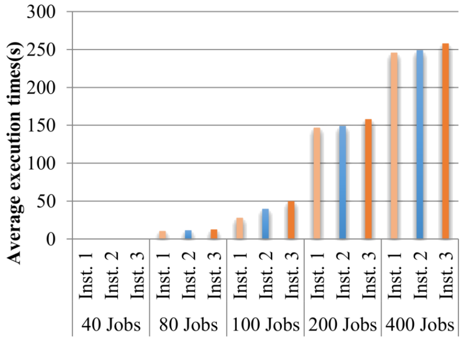

Fig. 10. Comparison of average execution times for LP.

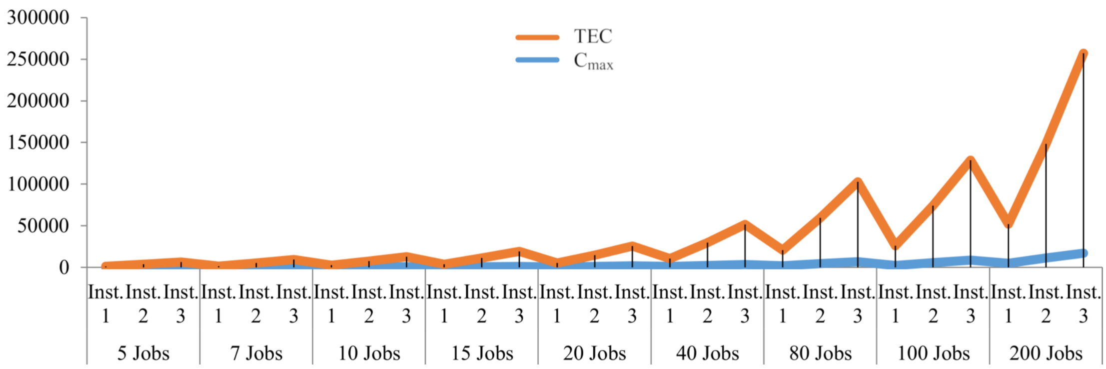

Fig. 11. Evolution of the total energy consumption and the Makespan depending on the instances.

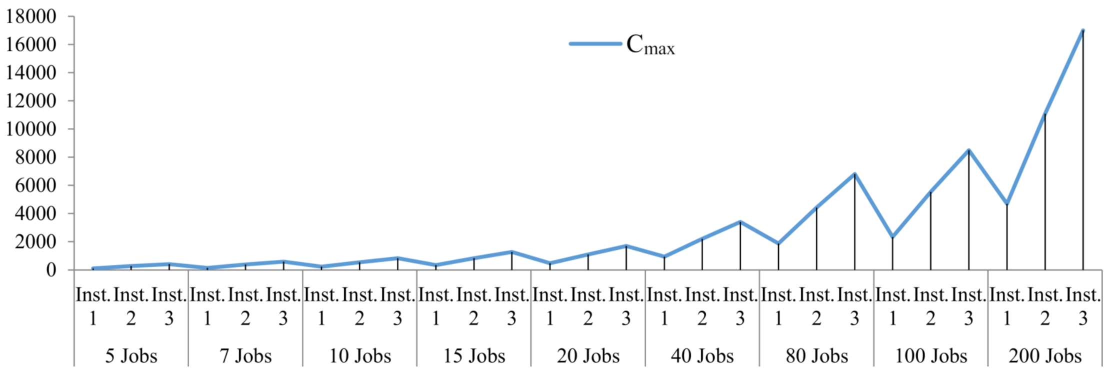

Fig. 12. Evolution of the Makespan depending on the instances.

tion [4, 6], the TEC value is 54920 which confirms all that is shown. With regard to the $C_{\max}$, the values provided in Table 2 and Figs 11 and 12 shows that increasing the final completion time of all jobs results in a significant increase in TEC. As an example for the second instance of these problems at (5 jobs) (10 jobs) (10 jobs) (20 jobs) (80 jobs) and (200 jobs) the completion time of all jobs $C_{\max}$ is equal to (270, 550, 1110, 4440 and 11100); respectively and the TEC follows this variation (3350, 6810, 13730, 54920 and 137299); respectively as shown in Table 2.

Conclusion

This article addresses the problem of integrated scheduling of production, preventive maintenance (PM) and corrective maintenance (CM) jobs in a single machine. The objective of this article is to minimize total energy consumption under the constraints of system robustness and stability. A propose a common model for the integration of preventive maintenance (PM) in production scheduling, where the sequence of production tasks is considered, as well as the preventive maintenance (PM) periods and the expected times for completion of the jobs are established simultaneously; this makes the theory put into practice more efficient. On the basis of the exact Branch and Bound method integrated on the CPLEX solver and the genetic algorithm (GA) solved in the Python software, the performance of the proposed integer binary mixed programming model is tested and evaluated. Indeed, after numerically experimenting with various parameters of the problem, the B&B algorithm works relatively satisfactorily and provides accurate results compared to the GA algorithm. A comparative study of the results proved that the model developed was sufficiently efficient. Despite this, the performance is considered tolerable since the suggested exact solution requires less than 258.05 seconds of computation time. Furthermore, in the future, studies may focus on extending the problem to different types of machine environments, such as the flow shop and the job shop using multi-objective optimization approaches.

References

[1] EIA, 2013, International Energy Outlook 2013 (online), www.eia.gov/forecasts/ieo/pdf/0484(2013).pdf, accessed on 5 May 2018.

[2] CSY, 2016, China Statistical Yearbook 2015 (online), www.stats.gov.cn/tjsj/ndsj/2016/indexeh.htm, accessed on 5 May 2018.

[3] Wang S., Wang X., Yu J., Ma S., Liu M., *Bi-objective identical parallel machine scheduling to minimize total energy consumption and makespan*, Journal of Cleaner Production (2018), doi: 10.1016/j.jclepro.2018.05.056.

[4] Ding J., Song S., Zhang R., Chiong R., Wu C., *Parallel machine scheduling under time-of-use electricity prices: new models and optimization approaches*, IEEE Transactions on Automation Science and Engineering, 13, 2, 1138–1154, 2016.

[5] Mouzon G., *Operational methods and models for minimisation of energy consumption in a manufacturing environment*, Ph.D. thesis, Wichita State University, Wichita, the United States of America, 2008.

[6] Luo H., Du B., Huang G., Chen H., Li X., *Hybrid flow shop scheduling considering machine electricity consumption cost*, International Journal of Production Economics, 146, 423–439, 2013.

[7] Liu Y., Dong H., Lohse N., Petrovic S., Gindy N., *An investigation into minimizing total energy consumption and total weighted tardiness in job shops*, Journal of Cleaner Production, 65, 87–96, 2014.

[8] Gahm C., Denz F., Dirr M., Tuma A., *Energy – efficient scheduling in manufacturing companies: a review and research framework*, European Journal of Operational Research, 248, 3, 744–757, 2016.

[9] Giret A., Trentesaux D., Prabhu V., *Sustainability in manufacturing operations scheduling: a state of the art review*, Journal of Manufacturing Systems, 37, 1, 126–140, 2015.

[10] Merkert L., Harjunkoski I., Isaksson A., Saynevirta S., Saarela A., Sand G., *Scheduling and energy-industrial challenges and opportunities*, Computers and Chemical Engineering, 72, 183–198, 2015.

[11] Sadiqi A. et al., *Joint scheduling of jobs and variable maintenance activities in the flowshop sequencing problems: review, classification and opportunities*, International Journal of Engineering Research in Africa, 39, 170–190, 2018.

[12] Sadiqi A., El Abbassi I., El Barkany A., El Biyaali A., *Comparative analysis the simultaneous scheduling problems of production and maintenance activities*, April 2018 Indexed SCOPUS, IEEE Xplore, 10.1109. LOGISTIQUA.2018.8428283.

[13] Zhiqiang Lu, Weiwei Cui, Xiaole Han, *Integrated production and preventive maintenance scheduling for a single machine with failure uncertainty*, Computers & Industrial Engineering, 80, 236–244, 2015.

[14] Goren S., Sabuncuoglu I., *Robustness and stability measures for scheduling single-machine environment*, IIE Transactions, 40, 66–83, 2008.

[15] Weinstein L., Chung C.H., *Integrating maintenance and production decisions in a hierarchical production planning environment*, Computers & Operations Research, 26, 1059–1074, 1999.

[16] Lin W., Yu D.Y. Zhang C., Liu X., Zhang S., Tian Y., Liu S., Xie Z., *A multi objective teaching-learning-based optimization algorithm to scheduling in turning processes for minimizing makespan and carbon footprint*, J. Clean. Prod., 101, 337–347, 2015.

[17] Zhang H., Deng Z., Fu Y., Lv L., Yan C., *A process parameters optimization method of multi-pass dry milling for high efficiency, low energy and low carbon emissions*, J. Clean. Prod., 148, 174–184, 2017.

[18] Meng L., Zhang C., Shao X., Ren Y., Ren C., *Mathematical modelling and optimisation of energy conscious hybrid flow shop scheduling problem with unrelated parallel machines*, Int. J. Prod. Res., 1–27, 2018.

[19] Baker K.R., Keller B., *Solving the single-machine sequencing problem using integer programming*, Computers & Industrial Engineering, 59, 4, 730–735, 2010.

[20] Sadiqi A., El Abbassi I., El Barkany A., El Biyaali A., *Non-permutation flow shop scheduling problems with unavailability constraints to Minimize Total Energy Consumption*, April 2019, 10.1109/ICOA.2019.8727649.

[21] Clausen J., *Branch and bound algorithms – Principles and examples*, Department of Computer Science, University of Copenhagen, 1–30, 1999.

[22] Assia S., El Abbassi I., El Barkany A., Darcherif M., El Biyaali A., *Green Scheduling of Jobs and Flexible Periods of Maintenance in a Two-Machine Flowshop to Minimize Makespan, a Measure of Service Level and Total Energy Consumption*, Advances in Operations Research, 1–9, 2020, doi: 10.1155/2020/9732563.

[23] Sadiqi A., El Abbassi I., El Barkany A., Darcherif M., El Biyaali A., *Speed scaling technique integrated in scheduling of production and maintenance under energy constraints using genetic algorithms*, E3S Web Conf. 170 01029, 2020, doi: 10.1051/e3sconf/202017001029.

[24] Sadiqi A., El Abbassi I., El Barkany A., Darcherif M., El Biyaali A., *Optimizing electricity costs during integrated scheduling of jobs and stochastic preventive maintenance under time-of-use electricity tariffs*, Management and Production Engineering Review, 10, 123–132, 2019, doi: 10.24425/mper.2019.131452.