

# Agile Approach for Joint Scheduling of Production and Maintenance in Flow Shop

Chaouqi M.

Laboratoire LISER ENSEM,  
Km 7 BP 8118 Route El Jadida  
Casablanca, Maroc

Benhra J.

Laboratoire LISER ENSEM,  
Km 7 BP 8118 Route El  
Jadida Casablanca, Maroc

Zakari A.

Laboratoire LITA,  
Université de Lorraine, 57000  
Metz, France

## ABSTRACT:

Modeling and optimizing joint production-maintenance functions is a part of complex system's representation domain. The exact methods are spread for their simplicity and efficiency. The use of Meta heuristics for optimization problems is the aim of several researches in various domains, enabling us to obtain fast optimal solutions.

The focus of this study is resolving of the scheduling problem using some heuristics. To achieve this goal, an informatics application has been developed which simulate Johnson's algorithm applied in the scheduling of the production for makespan minimization as well as the optimal insertion of maintenance tasks in the elaborated schedule. Simulation was implemented for one or several machines.

## General Terms

Algorithms, performance and reliability.

## Keywords

Scheduling, production, maintenance, algorithm, Johnson's algorithm generalized, Genetic Algorithm, flow shop makespan.

## 1. INTRODUCTION:

Maintenance and production are two functions often opposable, carrying through the same objective: insuring an industrial company's continuity.

On the one hand the maintenance department tries to maintain the means of production in good condition by repairing unforeseen stoppages, e.g. preventive maintenance.

On the other hand the production department tries to meet the deadlines imposed by customers by avoiding inevitable stoppages.

Schedules maintenance/production are often independently planed thus leading to conflicts between both departments to possible conflicts.

This problem has become a subject at several studies and research. In this paper, a general approach is proposed which allows for a flexible and agile scheduling thus meeting the needs of both departments.

## 2. STATE OF THE ART:

There are few works which concern the joint scheduling of production and maintenance. The studies found on this subject are the ones of Lee [1] who tried to minimize the makespan Cmax in the case of a flow shop workshop with two machines which undergo a single period of unavailability and preemptive maintenance tasks.

Two years later the author extended his study to machines which can undergo several periods of unavailability [2]. In the case of non preemptive tasks, he developed a heuristics, applied when the first machine undergoes unavailability.

Allaoui and al. [3] focused on the minimization of Cmax of a flow shop workshop with two machines in case the first machine undergoes a single period of unavailability. They were interested in the cases of the preemptive and non preemptive tasks. The authors proposed a dynamic programming algorithm to solve this problem.

Aggoune [4] studied the Cmax minimization problem with periods of unavailability in a flow shop workshop with several machines.

Kaabi and al [5] studied the case of flow shop permutations where the maintenance periods must be made in a predefined interval. This study has the advantage of decreasing the conflict between both departments, but it favors maintenance.

Ruiz and al [6] proposed a joint optimization scheduling of production and maintenance for the flow shop permutation case to reduce the run time. The authors used models of reliability to determine the periods of maintenance in the planning.

In [5], the fitness function is a balanced sum of two criteria: the total production delay and maintenance advances delays. Chaouqi and al. [7] used the same fitness function, however to optimize it, they used a hybridization of two heuristics; the optimization by an ants colony and the Taboo method in the job shop case.

## 3. MODELING AND SIMULATION:

### 3.1. Organization in production:

In this part, a set of algorithms which take into account these scheduling problems in various cases has been proposed. [8]

#### 3.1.1. Case of one machine:

##### 3.1.1.1. According to increasing operating Time:

This method allows minimizing average completion time to be minimized.

$$\bar{A} = \frac{1}{n} \sum_{i=1}^n A_i = \frac{1}{n} \sum_{i=1}^n (n - j + 1) T_i$$

With  $T_i$  is the task,  $i$  as the operating duration on the machine, and  $n$  as the total number of tasks.

If  $T_1 \leq T_2 \leq \dots \leq T_n$ , then the average completion time will be minimal, thus the algebraic delay will be minimized..

For example, on one production center, six jobs of production are executed such as:

**Table 1. Application example**

Task i	1	2	3	4	5	6
Operating duration $T_i$	50	30	10	100	70	20
Completion duration $A_i$	50	80	90	190	260	280
$\bar{A}$	158,33					
Execution Ordrer	1	2	3	4	5	6
Task i	3	6	2	1	5	4
$T_i$	10	20	30	50	70	100
$A_i$	10	30	60	110	180	280
$\bar{A}$	111,67					

**3.1.1.2. Use of the balanced TOM's rule:**

The scheduling is made according to the priority of the task, i.e. according to its important order.

The completion average time is:

$$\bar{A} = \frac{1}{n} \sum_{i=1}^n \alpha_i T_i$$

With  $\alpha_i$  the coefficient of level-headedness.

In that case the planning is made according to an increasing order of  $\frac{T_i}{\alpha_i}$  :

$$\frac{T_1}{\alpha_1} \leq \frac{T_2}{\alpha_2} \leq \dots \leq \frac{T_n}{\alpha_n}$$

The following table summarizes an application of the TOM's rule in the previous example:

**Table 2 Application example of TOM's rule**

Task i	1	2	3	4	5	6
Operaing duration $T_i$	50	30	10	100	70	20
Coefficient of level-headedness $\alpha_i$	1	2	1	3	2	4
ratio $T_i/\alpha_i$	50	15	10	33,3	35	5
Increasing Order $T_i/\alpha_i$	6	3	2	4	5	1
$\bar{A}$	158,33					
Execution Order	1	2	3	4	5	6
Task i	6	3	2	4	5	1
$T_i$	20	10	30	100	70	50
$\alpha_i$	4	1	2	3	2	1
$\sum_{i=1}^n \alpha_i T_i$	80	90	150	450	590	640
$\bar{A}$	333,33					

**3.1.1.3. According to the rule of lower delivery date (Jackson rule):**

This rule allows minimizing the true delay by scheduling in increasing order dates:  $d_1 \leq d_2 \leq \dots \leq d_n$  with  $d_j$  as the delivery date required for the job  $j$ .

Compared with TOM's rule which reduces the average delay, Jackson's rule minimizes the maximum delay.

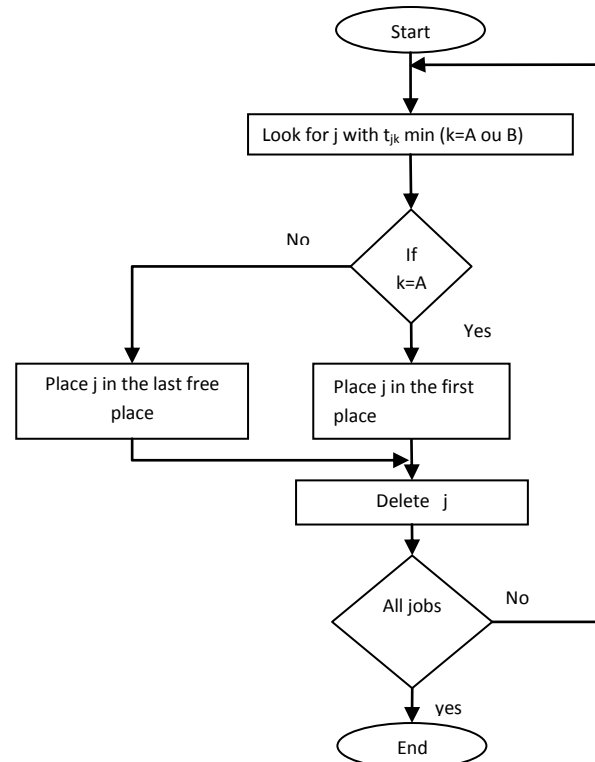
**3.1.1.4. Scheduling by rule of the minimal margin:**

In this case the production tasks are scheduled in increasing order margins  $d_j - T_j$ . This method allows maximizing the lowest delay.

**3.1.2. Case of n tasks to execute on two machines:**

**3.1.2.1. Identical Running order on machines (A then B):**

In that case the Johnson's algorithm can be used:



**Figure 1: Johnson's algorithm**

**3.1.2.2. Case of the non-uniqueness of the passage order on both machines {A; B}:**

In that case the Jackson's algorithm associated with the Johnson's algorithm can be applied:

**Step 1:**

Distribute the initial set in 4 partitions:

{A}: All the jobs to be executed on A only

{B}: all the jobs to execute on B only

{A-B}: all the jobs to execute on A then B

{B-A}: all the jobs to be executed on B then A

**Step 2:**

Apply the algorithm of Johnson to {A-B} = Sequence 1;

Apply the algorithm of Johnson to {B-A} = Sequence 2;

Follow any order on {A} = Sequence 3;

Follow any order on {B} = Sequence 4.

**Step 3:**

- On A, follow sequences: 1, 3 then 2

- On B, follow sequences: 2, 4 then 1

**3.1.3. Case of n jobs on m machines execution:**

**3.1.3.1. Any Running order (Job-shop):**

In this case an heuristic based on the busiest Machine's schedule can be used:

**Step 1:** determine the bottleneck;

**Step 2:** Determine the durations of executions of the tasks on machines upstream and downstream it of the bottleneck:

- Before bottleneck\_ =  $\sum$ duration of a various tasks upstream
- After bottleneck =  $\sum$  duration of a various tasks Downstream

**Step 3: use of TOM's rule**

However there have been several works elaborated on this subject. For example those of Zhang and al [9] and Zhuo and al [10] who used meta heuristics optimization by ants colony to solve this kind of problem.

### 3.1.3.2. Identical Running order (Flow shop):

In that case Johnson's generalized algorithm can be applied.

## 3.2. Maintenance scheduling:

The scheduling function in maintenance is to pilot three different elements: needs, resources and constraints. [11]

A good maintenance schedule is based on a set of indicators:

- MTTF (Mean Time to Failure) it is the average time before failure. This indicator is calculated for the first failure or for the disposable parts;
- MTBF (Mean Time before Failure) it is the average time between two failures. In other literature it is defined as the average as the average time of smooth running. It is bound to the laws of reliability by the rate of failure;
- MUT (Mean up Time) it is the time of availability;
- MDT (Mean down Time) Time of unavailability;

MTTR (Mean Time to Repair) It is the average time of repair before starting.

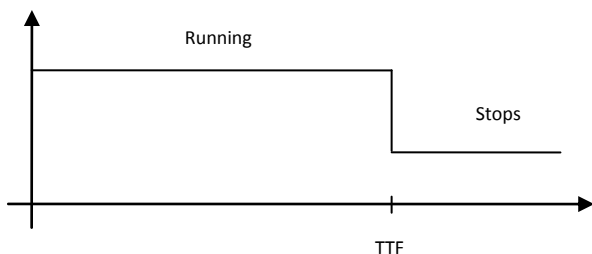


Figure 2: Case of a non-repairable system [13]

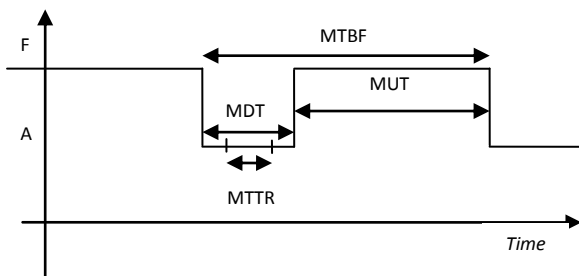


Figure 3: Case of a repairable system [13]

In maintenance, the availability is defined as being the capacity of a device to be able to achieve its required function in given conditions and in a given time interval supposing that the necessary supply is assured.

The availability is often associated with two running factors: reliability and maintainability.

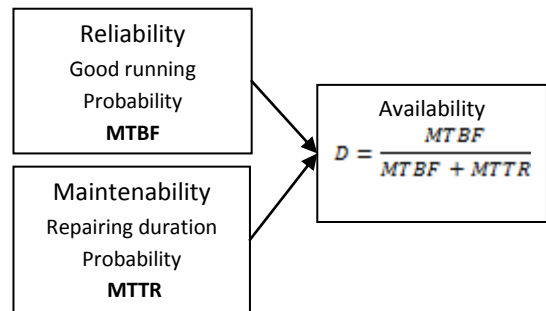


Figure 4: Reliability, maintainability and availability relations

The relation who defines the availability is:

$$D = \frac{MTBF}{MTBF + MTTR}$$

As in [13]:

$$\text{The rate of failure by: } \lambda = \frac{1}{MTBF}$$

$$\text{The rate of repair by: } \mu = \frac{1}{MTTR}$$

The formula of the availability becomes then:

$$D = \frac{\mu}{\lambda + \mu}$$

The failure ratio and the repairing ratio are two similar parameters. They are often considered by mathematical models as in the Weibull model for failure ratio estimation.

Increasing availability then consists in reducing stoppage time and the time to solve it.

The following graph represents the empirical costs curves engendered by maintenance, the production stops and failures, according to stoppages time [12]:

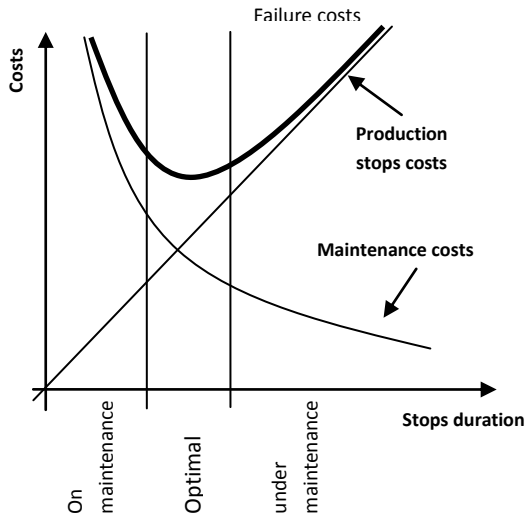


Figure 5: costs graph according to a month stops hours

Stoppage frequency has a big influence on failure costs. Thus the stoppage time must be taken into account to increase availability and costs reduction.

To simplify this study the equality  $MTTR=MDT$  is supposed true. Then :

- $T_{max}=MTTF$  (resp.  $T_{max}=MTBF$ ) the latest date of the first intervention (resp. after the first intervention),
- $T_{min} = MTTF - \varepsilon$  (resp.  $T_{min} = MTBF - \varepsilon$ ) the earliest date as of the first intervention (resp. after the first intervention),
- $\varepsilon$  depends on the various involved costs and on the tolerated default risk ratio [14],
- The interventions will be more frequent than normal thus engendering higher maintenance costs. The same occurs if a task after  $T_{max}$  is postponed because there will be too many repairs, thus raising the maintenance costs (see figure 5),
- $[T_{mini} , T_{maxi}]$  is defined as the optimal interval for every maintenance task  $M_i$ ,
- Another assumption in the modelization part using genetic algorithms, is that each machine  $m_i$  requires only a single maintenance task  $M_i$  which must be executed completely to maintain it in good condition.
- Machines are a part of a flow shop workshop.

### 3.3. Joint scheduling of maintenance and production:

The model defined in this study relies on three objectives; the increase of the machines availability, reduction failure costs and the minimization of the total time of production tasks.

These three objectives can be modeled by the following three fitness functions:

$$f1 = \text{Min} \left( \sum_{k=1}^n \max(0, c_{max_k}) \right) = \text{Min} (\text{Max}(c_{ij}))$$

$$f2 = \text{Min} \left( \sum_{k=1}^m \text{Cost}_i \right)$$

$$f3 = \text{Max}(\prod_{k=1}^m D_i(t)) = \text{Min}(1 - \prod_{k=1}^m \bar{D}_i(t)) \quad [15]$$

With  $D_i(t)$  is the availability and  $\bar{D}_i(t)$  is the unavailability of the machine  $m_i$ .

The fitness global function is the following one:

$$F = \alpha f1 + \beta f2 + \rho f3$$

The coefficients  $\alpha$ ,  $\beta$  and  $\rho$  are coefficients of level-headedness attributed to three fitness functions according to their degree of importance in this optimization problem.

Considering the complexity of the problem, first Johnson's generalized algorithm has been used to minimize the Makespan then a model by using genetic algorithms was established to insert maintenance tasks on the production's schedule while minimizing the failure costs, maximizing the availability of the system and keeping the Makespan optimal. Finally an algorithm has been realized based on an intuitive method to simplify the insertion of maintenance tasks into the production schedule.

#### 3.3.1. Elaboration of the production schedule:

A better solution which minimizes  $f1$  using Johnson's generalized algorithm is determined; Then The Gantt diagram according to the chosen configuration is established, and the beginning and the ending dates of execution are calculated as well as the margin before each product and on each machine.

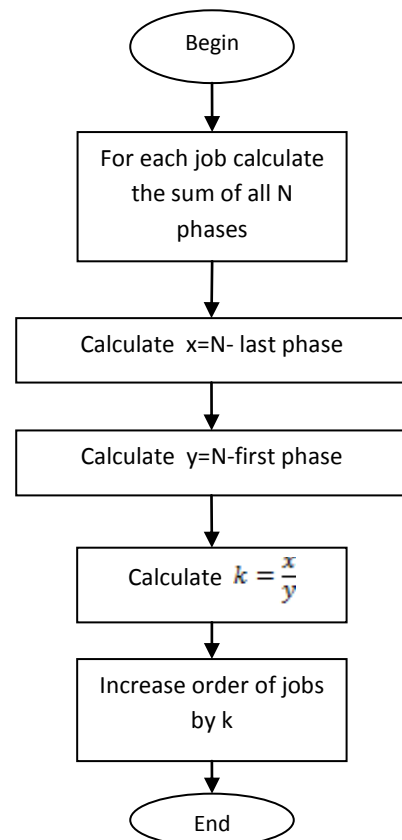


Figure 6: Johnson's generalized algorithm

Example of application of Johnson's generalized algorithm:

**Table 3. Application Example of the Johnson's generalized algorithm**

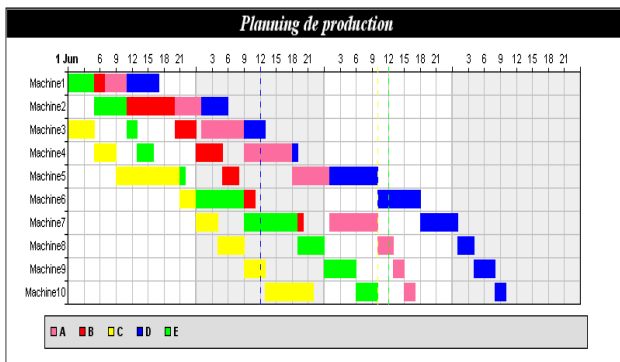
Machine	Task's execution time				
	A	B	C	D	E
M1	4	2	0	6	5
M2	5	9	0	5	6
M3	8	4	5	4	2
M4	9	5	4	1	3
M5	7	3	12	9	1
M6	0	2	3	8	9
M7	9	1	4	7	10
M8	3	0	5	3	5
M9	2	0	4	4	6
M10	2	0	9	2	4
Total	49	26	46	49	51
x	47	25	37	47	47
y	45	24	41	43	46
x/y	1,044	1,042	0,902	1,093	1,022

According to the Johnson's generalized algorithm, the sequence which minimizes Cmax is the following one:

C-E-B-A-D

	Marge avant produit				
	C	E	B	A	D
M1		0	0	0	0
M2		5	0	0	0
M3	0	6	7	1	0
M4	5	4	8	4	0
M5	9	0	7	10	0
M6	21	0	0		23
M7	24	5	0	5	8
M8	28	10		10	12
M9	33	11		7	13
M10	37	8		5	15

As hypothesis the workshop is opened 24 hours a day, 7 days a week. The GANTT chart associated with the previous example is:



**Figure 7: GANTT's chart**

An optimal schedule of production by using Johnson's generalized algorithm is obtained.

### 3.3.2. Genetic algorithm Modeling:

Meta heuristics are methods which can supply problems requiring a mattering time to be solved by determinist methods. Among these Meta heuristics, genetic algorithms is able to find optimal solutions in a reasonable amount of time based on the evaluation of several generations by crossing and mutating them.

In this part a modeling of the maintenance tasks insertion problem in the production schedule elaborated by Johnson's generalized algorithm, using the genetic algorithm has been realized.

#### Fitness function:

The fitness function to evaluate individuals was defined as follows:

$$F = \alpha f_1 + \beta f_2 + \rho f_3$$

#### Individual:

The determination of the problem variables represents an essential phase for its resolution. The variables chosen are sequenced by the positions of tasks maintenance with regard to the positions of production jobs.

Let  $p_{ij}$ : be compound position «The maintenance task  $M_i$  runs after the job  $j$  and before job  $j+1$ »

$i \in \{1, \dots, m\}$  and  $j \in \{0, 1, \dots, n\}$

$j=0$  corresponds to a fictitious job placed in the production schedule before the first job (It can be the last job of a former production cycle).

In genetic algorithms a solution is characterized by individuals. In this study an individual is considered as being the sequence of  $m$  positions of maintenance tasks on  $m$  machines in the order with regard to jobs to be executed.

The initial population of the individuals is noted  $P$ .

Example of solution for  $n=7$  and  $m=4$

P13	P21	P32	P40
-----	-----	-----	-----

#### Coding:

In genetic algorithms an individual is a member of a population, and is represented by one or several chromosomes.

Every chromosome includes a succession of several genes. They represent the parameters of the possible solutions and all the values are chromosomes that must be codified differently. [16]

A binary coding was used in that case, which is the most common coding used. The binary coding consists in representing every gene by a long integer, chromosomes are represented by an array of genes and the individuals are represented by an array of chromosomes.

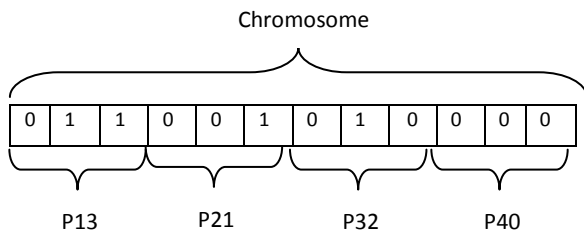
Every parameter  $P_{ij}$  corresponds to a gene  $i$  represented by the integer  $j$  coded in binary.

In the previous example  $n=7$ ;  $m=4$ . The parameter  $P_{13}$  of the previous solution corresponds to the first gene represented by the integer 3 coded 011.

Then the individual:

P13	P21	P 32	P40
-----	-----	------	-----

Is represented by the following chromosome:



**Selection:**

This selection allowed us to determine  $N/2$  individuals of the population  $P$  ( $|P| = N$ ) which will be duplicated in the population  $P'$  and from other individuals called sons will be generated by means of a crossing operator. [16]

The biased lottery wheel is the most commonly known operator of selection. This method consists in increasing the individual's chance of selection, is the most adapted to this problem. So an individual having a higher performance will have more chances of being selected.

The similarity to a biased lottery wheel is to attribute a sector having a proportional angle in its degree of fitness adaptation to every individual.

The wheel turns  $N/2$  times and the chosen individual corresponds to the sector indicated by a cursor when the wheel stopped. This step is redone  $N/2$  times.

**Crossover:**

The crossover allows for the generation of the chromosomes which inherit partial characteristics of the parents. It applies a recombination of the information present in the first population.

This operator was applied after the selection of the parents. The number of the selected parents is  $N/2$ , and then those parents were distributed in  $N/4$  couples from which  $N/2$  sons were reproduced.

The number of the crossing points can go up to  $L-1$  ( $L$  is the length of the chromosome), the higher number is the higher amount of information exchanged is, thus providing much more diversity in the new population  $P'$ .

Every chromosome finds itself then divided into a number of segments. Later every segment of the parent 1 is exchanged with his counterpart of the parent 2 according to a crossing probability.

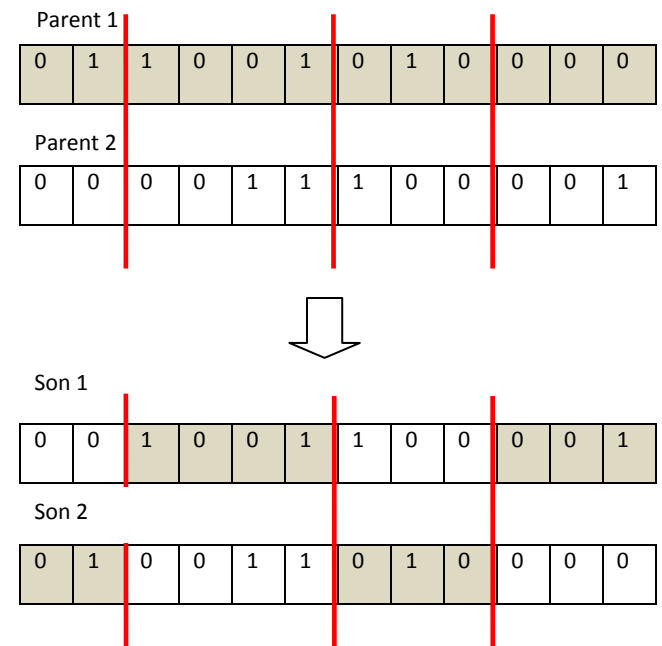
In the previous example with  $n=7$  and  $m=4$ . Both individuals selected by the population are:

P13	P21	P 32	P40
-----	-----	------	-----

And

P10	P23	P 34	P41
-----	-----	------	-----

The corresponding chromosomes are respectively:



**Figure 8: Crossover operator**

Both sons obtained after crossing of both parents represent following both individuals:

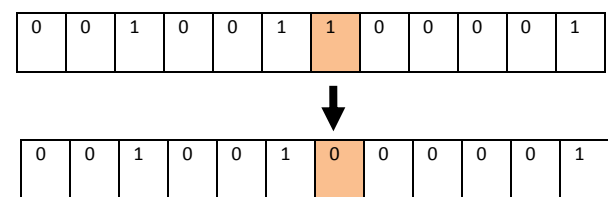
P11	P21	P 32	P41
-----	-----	------	-----

And

P12	P23	P 32	P40
-----	-----	------	-----

**Mutation:**

The mutation consists in exchanging the chromosome value of a gene with a very low probability  $p_m$ . In particular,  $p_m$  can be set as the reverse of  $L$  length of chromosome:  $p_m=1/L$ .



The new individual generated after mutation is:

P11	P21	P 30	P41
-----	-----	------	-----

**Figure 9: Crossover Operator**

**Replacement Operator:**

This operator role is to replace a portion of the individuals in the first population  $P$  by downward individuals in the population  $P'$ . The number of individuals introduced into the population  $P$  as well as the replaced number of individuals is called gap generation.

Two types of operators of replacement can be distinguished:

- The still replacement,
- The elitist replacement.

### 3.3.3. Intuitive method:

In this part an intuitive method was used to simplify the insertion of maintenance tasks in the production schedule elaborated by Johnson’s generalized algorithm. The algorithm proposed here consists in comparing the task duration and the margins which are situated before the products to be processed on the same machine:

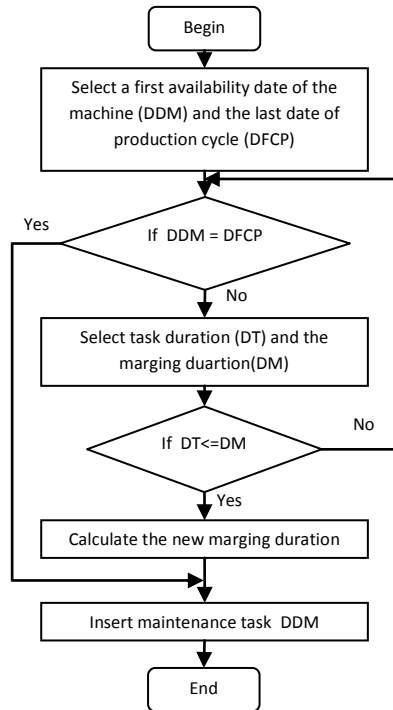


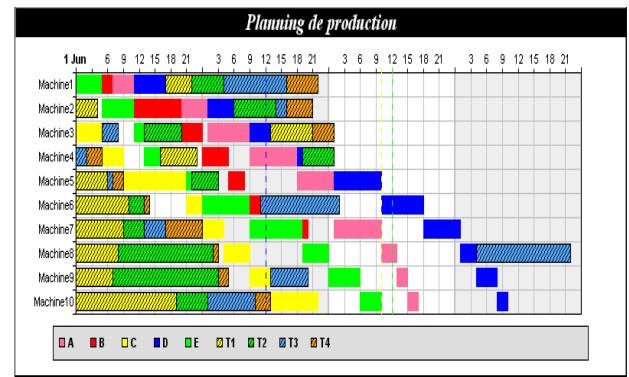
Figure 10: Insertion algorithm

Looking again at the Gantt chart elaborated in the previous example, as well as the following maintenance tasks:

Table 4. Durations of execution of maintenance tasks on every machine

Machine	Maintenance tasks			
	T1	T2	T3	T4
M1	5	6	12	6
M2	4	8	2	5
M3	8	7	3	4
M4	7	6	2	3
M5	6	5	1	2
M6	10	3	15	1
M7	9	4	4	7
M8	8	18	18	1
M9	7	20	7	2
M10	19	6	9	3

To simulate this insertion an application is developed. The Gantt schedule after insertion becomes:



A solution to the problem of insertion is gotten in a simple way using this method.

## 4. CONCLUSION:

The use of Meta heuristics for optimization problems is the aim of several researches in various domains, enabling us to obtain fast optimal solutions. The exact methods are spread for their simplicity and efficiency. In this work a set of methods used in scheduling was illustrated, then a model based on cooperation of three methods was proposed: the Johnson’s generalized algorithm for production planning on the one hand, and the genetic algorithms as well as an intuitive method for the insertion of maintenance tasks on the other hand.

The insertion of a maintenance task is made after the end of a production operation situated inside the interval of tolerance. The best configuration is the one which reduces the fixed fitness function.

Considering the complexity of the problem, this study was limited to the insertion modeling problem, by using genetic algorithms. On the other hand, an application was developed to simulate Johnson’s generalized algorithm applied in the scheduling of the production for Makespan minimization as well as the optimal insertion of maintenance tasks in the elaborated schedule.

As a conclusion, The Johnson's generalized algorithm allows to find an optimal solution quickly and that a flexible exploitable schedule is obtained with the proposed intuitive method.

## 5. REFERENCES

- [1] C.Y. Lee, 1997. Minimizing the makespan in two machines flowshop scheduling problem with an availability constraint. *Operations Research Letters*, 20:pp.129–139.
- [2] C.Y. Lee, 1999. Two-machine flowshop scheduling problem with availability constraints. *European Journal of Operational research*, 114: pp.420–429.
- [3] H. Allaoui, A. Artiba, F. Riane, and S.E. Elmaghraby, 2003. On the two machines flow shop with availability constraints. In *Proceedings for the International Conference on Industrial Engineering*
- [4] R. Aggoune. 2001. Minimizing the makespan for the flow shop scheduling problem with availability constraints. In *Operational Research Peripatetic Post-Graduate Programme, ORP3, Paris*.
- [5] J. Kaabi, C. Varnier, and N. Zerhouni. *Odonancement de la production et de la maintenance : cas d'un atelier*

- de type flow shop a deux machines. APIIJESA, 37: pp. 641–660, décembre 2003.
- [6] F. Hnaien, F. Yalaoui, Optimisation bi-objectif d'ordonnancement et de maintenance d'un atelier *flow-shop*, 8e Conférence Internationale de MODélisation et SIMulation - MOSIM'10, 2010
- [7] Mohsine Chaouqi, Jamal Benhra, Yamine Eloudari 2011, Simultaneous control of production and maintenance through meta-heuristics' hybridization, International workshop on Information Technologies and Communication.
- [8] Vincent GIARD, 2003. Gestion de la production et des flux, 3e édition, Economica
- [9] Jun Zhang, Xiaomin Hu, X. Tan, J.H. Zhong and Q. Huang, Implementation of an Ant Colony Optimization technique for job shop scheduling problem, Transactions of the Institute of Measurement and Control 28, 1 (2006) pp. 93-108
- [10] Xiao-Lan Zhuo, Jun ZHANG, MIEEEE and Wei-neng Cheng, A new pheromone design in ACS for solving JSP, 2007 IEEE Congress on Evolutionary Computation (CEC 2007)  
<http://www.ee.cityu.edu.hk/~jzhang/papers/cec07zxl.pdf>
- [11] Michel Villain, Organisation et méthodes de maintenance, V3.0 – 2003, Chapitre 12 – pp : 57-71.
- [12] François Monchy Jean-Pierre Vernier, MAINTENANCE Méthodes et organisations, 3<sup>ème</sup> édition, l'USINE NOUVELLE DUNOD, Chap. 7 : Les fonctions ordonnancement, logistique et réalisation, page : 303.
- [13] LYONNET Patrick , La maintenance. Mathématiques & méthodes, 3<sup>ème</sup> édition, 1992, éditions TEC&DOC
- [14] M. Bennour, C. Bloch, N. Zerhouni, Modélisation intégrée des activités de maintenance et de production, 3e Conférence Francophone de MODélisation et SIMulation "Conception, Analyse et Gestion des Systèmes Industriels" MOSIM'01, 2001  
<http://www1.utt.fr/mosim01/pdf/ARTICLE-194.pdf>
- [15] F. HNAIEN, Optimisation bi-objectif d'ordonnancement et de maintenance d'un atelier flow-shop, 8e Conférence Internationale de MODélisation et SIMulation - MOSIM'10, « Evaluation et optimisation des systèmes innovants de production de biens et de services »
- [16] Souquet Amédée, Radet Francois-Gérard, algorithmes génétiques TE de fin d'année, 21/06/2004  
<http://deptinfo.unice.fr/twiki/pub/Linfo/PlanningDesSoutenances20032004/Radet-Souquet.pdf>