



## **Rapport d'ordonnancement :** **Open-shop**

### **Fait part :**

Mickaël Charles  
Frédéric Caron  
Florian Pierrel  
Djelouah Laala  
Hamou Aldja Kahina

**Table des matières**

I. Définition du problème (open-shop).....3

II. Les heuristiques.....3

III. Les Algorithmes Génétiques.....4

IV. Modélisation du problème.....4

V. Résultats.....5

## I. Définition du problème (open-shop)

On considère le problème d'ordonnancement  $O3||\sum C_i$  (open-shop de 3 machines avec minimisation de la somme des dates de fin d'exécution des jobs).

C'est un modèle d'atelier moins contraint que le job-shop et le flow-shop, car l'ordre d'exécution des tâches d'un job  $j$  est libre, 3 machines sont disponibles pour traiter  $n$  jobs. Chaque job comprend 3 tâches. La tâche  $i$  du job  $j$  doit être traitée par la machine  $i$ , en une durée opératoire  $p_{ij}$ .  $C_i$  est la date de fin de la tâche  $i$ ;  $C_{\max}$  est la date de fin de la dernière tâche réalisée par une machine.

## II. Les heuristiques

Les heuristiques sont des algorithmes qui fournissent en temps polynomial une solution réalisable généralement bonne et proche de l'optimale, L'objectif d'une heuristique n'est pas d'obtenir un optimum global, mais seulement une « bonne » solution en un temps raisonnable pour un problème d'optimisation NP-difficile.

- **SPT (Shortest Processing Time)** on favorise parmi les tâches possibles à exécuter celle qui a la plus courte durée. On parle aussi de (*Traitement par ordre croissant du temps opératoire*).

L'idée de les traiter en premier consiste à ce qu'elles n'occupent pas beaucoup du temps mais il est possible qu'il y ait d'autres opérations qui dépendent de cette opération.

- **LPT (Longest Processing Time)** on favorise chaque fois parmi les tâches possibles celle qui a la plus longue durée. On parle aussi de (*Traitement par ordre décroissant du temps opératoire*). L'idée de les traiter en premier consiste à se débarrasser des tâches les plus difficiles le plus vite possibles.

- **MWR (Most work remaining)**

La priorité est donnée au travail dont la durée résiduelle (durée totale des opérations non traitées) est maximale.

- **Random**

Les opérations sont ordonnancées dans un ordre aléatoire.

### **III. Les Algorithmes Génétiques**

Les algorithmes génétiques sont constitués par une catégorie de programmes dont l'objectif est de résoudre un problème en reproduisant les mécanismes de la sélection naturelle. Ces algorithmes sont particulièrement adaptés à l'optimisation de problèmes combinatoires et notamment des problèmes dits NP-complets (dont le temps de calcul croît de façon non polynomiale avec la complexité du problème).

Ces algorithmes constituent parfois une alternative intéressante aux réseaux de neurones mais sont le plus souvent complémentaires. Un exemple peut être donné dans le cas d'un système à N variables (dans le cas où N est grand) et où seulement certaines combinaisons de ces variables sont pertinentes dans quelques cas particuliers.

### **IV. Modélisation du problème**

- Fonction d'optimisation :

$$\text{Min } \sum_{i=1}^n j_i \quad (\text{avec } n \text{ le nombre de job et } j_i \text{ le makespan du job } i)$$

- Les variables de décision :

$\forall (i,k) \in \{0,n\}^2$  : si la tâche i et la tâche k s'exécute sur la même machine, ou au même job, on a alors  $x_{ik}$  tel que :

$x_{ik} = 1$  si la tâche i s'exécute avant la tâche k et 0 sinon.

- Contraintes d'affectations :

$\forall x_{ik} : x_{ik} + x_{ki} = 1$  (une nécessaire et une seule tâche précède l'autre).

- Contraintes d'anti chevauchement :

$\forall x_{ik} : C_i + M * (x_{ik} - 1) \leq C_k - p_k$  (avec M une borne max, par exemple  $M = \sum p_i$  et  $C_i$  le makespan de la tâche i).

- Contraintes temporelles :

$\forall i \text{ et } \forall j \text{ tel que la tâche } i \text{ appartienne au job } j \text{ on a : } p_i \leq C_i \leq j_j \text{ et } j_j \geq 0$

## V. Résultats

**Les instances :** Dans les trois instances, on représente la durée opératoire de chaque tâche de chaque job et sur quelle machine :

1)

Machine	1	2	3
Job			
1	6	0	2
2	3	1	5
3	4	0	0
4	7	1	8

2)

Machine	1	2	3
Job			
1	4	2	7
2	1	8	6
3	12	0	0
4	3	9	0
5	0	7	3
6	4	1	11

3)

Machine	1	2	3
Job			
1	2	5	4
2	3	9	2
3	10	0	0

### Comparaison :

Instance	1	2	3
Méthode			
SPT	60	122	46
LPT	72	141	41
MWR	71	141	41
Random	60	139	46
Algo Gen	49	118	39
PLNE	49	117	39

Le tableau ci dessus présente la somme des makespan de chaque job pour chaque instance et pour chaque méthode, en exception de la méthode Random ; nous remarquons bien que la méthode génétique nous donne des résultats plus intéressants que les résultats des autres méthodes, en effet 49 est la plus petite somme pour la première instance, 118 est la plus petite pour la deuxième instance (après la PLNE) et 39 est la meilleure valeur pour la troisième instance. Ensuite les résultats de la méthode SPT sont plus intéressants sur la première et la deuxième instance, ceux de la méthode LPT sont presque pareils sur les trois machines.