

Etude de la flexibilité opérationnelle d'un flow shop par des algorithmes génétiques

Kouassi Hilaire EDI¹, Tchimou G. E N'TAKPE^{1*} et Philippe DUQUENNE²

¹ *Laboratoire de Mathématique et Informatique, UFR-SFA, Université Nangui Abrogoua, 02 BP 801 Abidjan, Côte d'Ivoire*

² *INPT/ ENSIACET/LGC/PSI/GI, 118 Route de Narbonne, 31077 Toulouse Cedex 04, France*

* Correspondance, courriel : tchimou.ntakpe@gmail.com

Resumé

Dans un récent article nous avons résolu le problème de la flexibilité opérationnelle d'un « flow shop » en prenant en compte la flexibilité à l'aide d'une méthode exacte. Ce qui a permis d'obtenir une solution optimale, mais au prix d'un temps de résolution relativement grand, trop important pour une gestion d'atelier en temps réel. Le problème à résoudre est de caractériser les leviers de flexibilité liés à la variation des durées opératoires et des dates de livraison des articles d'un atelier de production à cheminement unique afin d'assurer la flexibilité opérationnelle du système. Dans le présent travail, nous utilisons cette fois les algorithmes génétiques pour approcher la solution optimale en recherchant un temps de résolution plus court. Il s'agit dans cet article d'exposer cette résolution et de présenter une comparaison avec celle de la méthode exacte précédemment utilisée.

Mots-clés : *ordonnancement, flexibilité opérationnelle, algorithme génétique, convergence.*

Abstract

Genetic algorithms - based study of the operational flexibility of a flow shop

In a recent paper, we have investigated the operational flexibility of a flow shop, through case studies solved with the help of an exact method. This allowed to find out an optimal solution, but at the cost of a large solving time, too large for the real-time management of a workshop. The problem to solve is to characterize the flexibility levers related to the change in operating times and delivery dates for items produced in a single-path production workshop, to ensure the operational flexibility of the system. In the present work, we will use genetic algorithms to approximate the optimal solution in search for a shorter resolution time. The aim of this article are to explain the resolution process and to present a comparison with the exact method used previously.

Keywords : *scheduling, operational flexibility, genetic algorithm, convergence.*

1. Introduction

La gestion de production a pour but la recherche et la mise en œuvre d'une organisation efficace pour la réalisation d'un programme de fabrication. Elle doit respecter les méthodes de transformation des produits, les contraintes sur l'engagement des ressources et les délais de livraison, tout en respectant les indicateurs de performance définis par l'entreprise [1]. Dans ce cadre, nous avons étudié la flexibilité d'un type d'atelier en l'occurrence le flow shop avec une méthodes exactes [2]. Cela a permis de trouver des solutions avec un temps relativement long. Le présent travail consiste à mettre à profit une méthode de résolution basée sur les algorithmes génétiques pour étudier la flexibilité des durées opératoires d'opérations de production, en utilisant la variation des effectifs alloués sur les postes d'un « flow shop ». Les dates de livraison des articles produits fluctuent dans un intervalle prédéterminé. La relation fondamentale entre la charge de travail d'une opération et l'effectif nécessaire est donnée par la **Formule** suivante :

$$\text{Effectif} = \frac{\text{charge de travail (en heures)}}{\text{durée opératoire ou cycle (en heures)}} \quad (1)$$

Ce travail fait appel à la notion d'évènements associés à chaque début ou fin d'une opération sur un poste de travail. Dans cet article, nous allons dans un premier temps faire une revue des problèmes d'ordonnancement dans la littérature, en mettant l'accent sur la flexibilité opérationnelle d'un flow shop ; dans un deuxième temps, nous présenterons le modèle mathématique résultant de cette réflexion et enfin, nous proposerons une résolution basée sur les algorithmes génétiques.

2. Méthodologie

2-1. Etude fondamentale

2-1-1. Classification du problème général d'ordonnancement d'atelier

Parmi les problèmes d'ordonnancement étudiés, les problèmes d'ateliers de fabrication forment une classe importante d'applications [3 - 8]. De ces travaux, on peut dresser une typologie de ces problèmes en fonction de l'ordre d'exécution des tâches sur les machines. On distingue principalement cinq types (**Figure 1**) : l'atelier à cheminements multiples (« job shop »), l'atelier à cheminement unique (« flow shop »), objet de notre étude, l'« open shop » (une tâche peut être exécutée dans un ordre quelconque, donc pas de gammes définies), le problème à une machine et le problème à machines parallèles qui est une généralisation du problème à une machine. Cette classification est donnée sur la **Figure 1** ci-dessous.

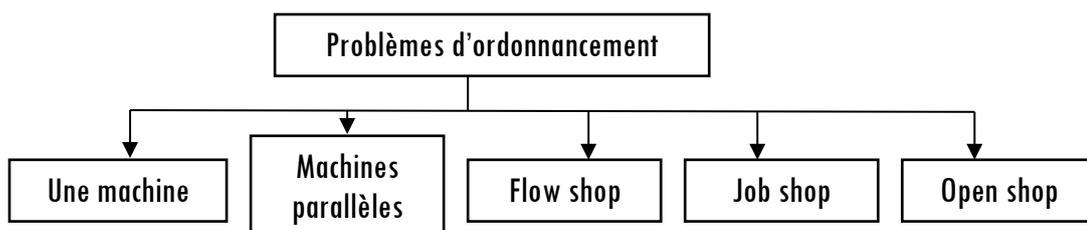


Figure 1 : Classification des problèmes d'ordonnancement classiques

De cette classification, nous allons pousser notre réflexion sur les ateliers à cheminement unique dits « flow shops » où nous allons introduire la notion de flexibilité sur les durées opératoires en fonction des effectifs alloués et la date de livraison qui se situera autour d'un intervalle défini au préalable pour chaque article. Nous exposons à présent une revue sur ce type de problème.

2-1-2. Problème de la flexibilité opérationnelle d'un flow shop

L'atelier à cheminement unique possède la caractéristique suivante : l'ordre de passage sur les différentes machines est imposé et identique pour tous les travaux notés (j). La flexibilité sur le flow shop [9 - 11] est une généralisation dans laquelle un article doit subir plusieurs étapes de transformation, chacune d'elles étant constituée de plusieurs machines identiques en parallèle (**Figure 2**). Dans ce schéma, un article subit trois opérations j1, j2 et j3 (j1 sur une machine 1, j2 sur une machine 2 et j3 sur une machine 3). Les machines M11 et M12 sont identiques et constituent la première étape, idem pour M21, M22 et M23 et la deuxième étape ainsi que pour M31 et M32 et la troisième étape. Pour fabriquer cet article, on a le choix de fait passer l'opération j1 soit sur la machine M11, soit sur la machine M12 (pas sur les deux machines) et ainsi de suite.

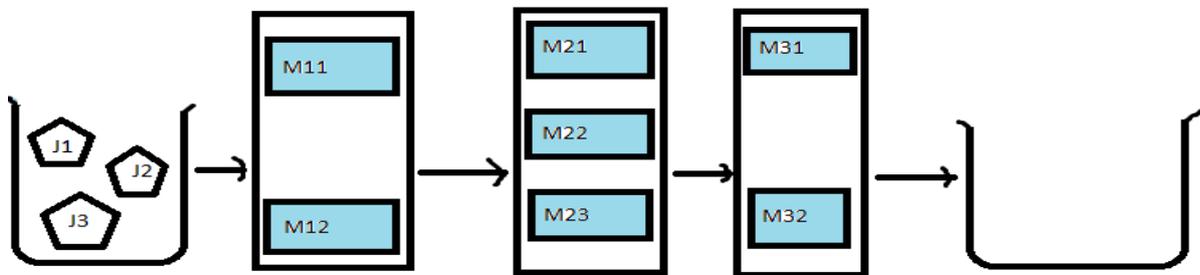


Figure 2 : Flow shop flexible

La littérature regorge d'études sur divers modèles d'ateliers ayant chacun des caractéristiques propres à son mode de fonctionnement. Les critiques portées sur le modèle de flow shop simple ont conduit de plus en plus de chercheurs à focaliser leurs recherches sur l'étude du modèle de flow shop flexible considéré comme réaliste en termes d'organisation d'ateliers de production. La plupart des études dans la littérature sont caractérisées par des durées opératoires (ou cycles) fixes [9 - 13]. De multiples travaux portent sur une configuration de flow shop flexible dans laquelle l'exécution d'une opération est précédée (resp. suivie) d'un temps de montage (resp. démontage) dépendant de la séquence des travaux (SDST : Sequence dependent Setup Time en anglais) sur la machine. Ce temps mobilise uniquement la ressource humaine et pas le poste de travail [14 - 16].

2-1-3. Méthode de résolution

Plusieurs méthodes de résolution ont été élaborées pour aborder les différents problèmes de flow shop flexible. On distingue les méthodes exactes [17], les heuristiques et les méta-heuristiques [18 - 20]. Il existe trois classes de méthodes exactes pour la résolution des problèmes d'ordonnement à savoir la procédure par séparation et évaluation (Branch & Bound), la programmation dynamique et la programmation linéaire. La méthode de séparation et évaluation est la plus utilisée parmi les méthodes exactes pour la résolution des problèmes de flow shop flexible, pour un faible nombre de postes. Compte tenu du temps d'exécution pour des problèmes de taille réduite nous nous sommes tournés vers l'utilisation des méthodes approchées par le biais des algorithmes génétiques.

2-1-4. Principe et méthode des algorithmes génétiques

Les algorithmes génétiques sont utilisés pour résoudre des problèmes d'optimisation. Cependant, leur mise en œuvre nécessite la définition des procédures et des paramètres associés [21] : un codage approprié (génotype) permet de représenter l'ensemble des variables du problème, la valeur de chaque variable de décision prenant le nom de « gène ». Un jeu complet, dans lequel toutes les variables de décision sont représentées avec pour chacune une valeur possible, est un individu : il est représenté par un vecteur dans lequel chacun des gènes nécessaire pour décrire une solution est présent : ce vecteur s'appelle un « chromosome ». On peut ainsi générer un ensemble d'individus (une « population ») aléatoirement, de manière à balayer l'ensemble de l'espace des solutions. Ces individus sont ensuite évalués : on calcule pour chacun la valeur de la fonction objectif (valeur qu'on appelle sa « force » ou son « adaptation ») ; les individus d'une population peuvent ainsi être classés en fonction de leur adaptation au problème posé. Le principe des algorithmes génétiques consiste à faire évoluer la population de génération en génération pour voir comment évolue son adaptation, en cherchant bien sûr à ce que cette adaptation s'améliore. Le passage d'une génération à une autre peut se faire :

- par croisement : on peut procéder à des échanges de gènes de deux individus « parents » de manière à créer deux individus hybrides « enfants ».
- par mutation : un individu « parent » (choisi au hasard) donne un individu « enfant » - mais au passage, un de ses gènes, tiré au hasard, aura vu sa valeur modifiée aléatoirement.
- certains individus (de préférence les plus adaptés, mais pas seulement ...) sont conservés à l'identique d'une génération à l'autre,
- enfin, d'autres individus disparaissent (de préférence les moins adaptés, mais pas seulement ...) : de manière à conserver une population constante d'une génération à l'autre, ils sont remplacés par des individus « régénérés », c'est-à-dire générés aléatoirement, comme pour la population originelle.

Pour chaque génération, on retient soigneusement la valeur de l'adaptation du ou des meilleurs individus, ainsi que la valeur moyenne de l'adaptation de l'ensemble de la population. Et bien entendu, on conserve de génération en génération la liste réactualisée à chaque itération des meilleurs chromosomes (solutions). Le processus peut prendre fin lorsqu'après un certain nombre de générations successives, la meilleure adaptation n'évolue plus ; ou bien lorsqu'après un certain nombre de générations successives, l'adaptation moyenne de la population n'évolue plus. Dans ces deux premiers cas, on considère que l'algorithme a convergé vers une solution supposée proche de l'optimum réel recherché. Le processus peut également s'arrêter après avoir « traité » un nombre maximal de générations (défini par l'utilisateur), sans qu'il ait pu converger vers une solution. Cette sécurité empêche le processus de recherche de se poursuivre indéfiniment.

2-2. Modélisation du problème

2-2-1. Les variable et les données

<p>Les indices</p>	<p>i indice pour l'article et m le nombre total d'articles sur la période, $1 \leq i \leq m$ j indice pour le poste et n le nombre total de postes, $1 \leq j \leq n$ k, o indices pour les événements et T le nombre total d'événements, $1 \leq k \leq T$ et $1 \leq o \leq T$ avec $m + n \leq T \leq 2.m.n$</p>
<p>Les variables continues</p>	<p>t_k : date d'occurrence de l'événement k, L_i^0 : date de livraison théorique de l'article i</p>
<p>Les variables discrètes</p>	<p>$N_{j,k}$: nombre de positions disponibles sur le poste j entre les événements k et $k+1$, $a_{s_{i,j,k}}$: variable binaire désignant l'affectation du début d'opération à l'événement k ($a_{s_{i,j,k}} = 1$ si l'article i arrive sur le poste j à la date de l'événement k et $= 0$ sinon), $a_{e_{i,j,k}}$: variable binaire désignant l'affectation de fin d'opération à l'événement k ($a_{e_{i,j,k}} = 1$ si l'article i quitte le poste j à la date de l'événement k et $= 0$ sinon), $\omega_{i,j}$: variable binaire matérialisant un changement de durée opératoire sur le poste j entre les articles i et $i+1$, E_k : effectif présent dans l'atelier entre les événements k et $k+1$, E^{tot} : effectif maximum présent dans l'atelier sur l'horizon de temps $E^{tot} = \max\{E_k\}$, $E_{i,j}$: effectif présent sur le poste j pour l'article i, γ_k : variable d'écart pour un événement k possible et non affecté, $c_{i,j}$: cycle de l'article i sur le poste j, $\delta_{i,j}$: date de disponibilité des éléments de l'article i pour le poste j.</p>
<p>Les données</p>	<p>c_j^{min} : valeur minimale du cycle pour le poste j, c_j^{max} : valeur maximale du cycle pour le poste j, γ : nombre d'articles sur lesquels se répercutent les perturbations, N_j^{min} : nombre minimal de positions disponibles sur le poste j (en principe égal à 0), N_j^{max} : nombre maximal de positions disponibles sur le poste j, N_j^0 : nombre de positions initialement disponibles sur le poste j (par défaut, $N_j^0 = N_j^{max}$), L_i^- : date de livraison au plus tôt pour l'article i (par défaut, $L_i^- = L_i^0 - 2$), L_i^+ : date de livraison au plus tard pour l'article i (par défaut, $L_i^+ = L_i^0 + 2$), χ_j : charge de travail sur le poste j, $v_{i,j}$: valeur de l'article i à l'entrée du poste j (en coût matière uniquement), $w_{i,j}$: valeur de la matière ajoutée à l'article i sur le poste j. il établit la relation suivante, pour assurer la cohérence des données, entre $v_{i,j}$ et $w_{i,j}$:</p> $\forall i \leq n-1, \forall j \quad v_{i+1,j} = v_{i,j} + w_{i,j}$

2-2-2. Description du problème

Nous considérons le cas d'un atelier de fabrication, constitué de n postes et devant produire m articles, répondant aux hypothèses suivantes : le volume de la demande est connu a priori, et la date de livraison L_i pour chaque article i doit appartenir à un intervalle $[L_i^-, L_i^+]$ déterminé ; tous les articles suivent la même gamme opératoire, et passent sur l'ensemble des postes, le nombre total d'opérations est donc connu ; les durées opératoires ou cycles sur chaque poste j et pour chaque article i , notées c_{ij} , sont variables ; pour chaque poste j et chaque article i , il existe un couple de données (cycle minimum, cycle maximum) entre lesquelles c_{ij} doit être compris ; chaque poste j est caractérisé par son nombre de positions N_j^{\max} , correspondant au nombre d'articles pouvant être traités en même temps à un instant donné (chaque poste est constitué d'un ou plusieurs machines identiques), leurs dates d'entrée et de et de sortie du poste étant indépendantes ; chaque article doit passer une et une seule fois sur chaque poste, en occupant une seule position (une seule machine) ; aucun stock intermédiaire n'est possible entre deux postes consécutifs dans l'atelier (politique de flux tendu). Sur chaque machine sont affectées des ressources. Dans ce travail, nous ne prenons en compte que les ressources humaines, et donc la flexibilité qu'elles apportent en termes d'horaires et d'effectifs variables.

2-2-3. Formulation mathématique

2-2-3-1. Les contraintes

Dans ce paragraphe nous exprimons les contraintes du système de production par un ensemble d'équations comme présentées ci-dessous :

- Chaque événement k (fin ou début d'opération pour un article i sur un poste j) doit se produire une et une seule fois sur l'ensemble de l'horizon.

$$\forall i, j \sum_{k=1}^T as_{i,j,k} = 1 \quad \text{et} \quad \forall i, j \sum_{k=1}^T ae_{i,j,k} = 1 \quad (2)$$

- Relation de précédence pour un article i entre deux postes (j et $j \neq l$) de la gamme : Pour qu'une opération j débute au $k^{\text{ème}}$ événement, l'événement déclenchant la fin de l'opération $j-l$ doit être affecté à un événement antérieur (donc compris entre 1 et k).

$$\forall i, \forall j > 1, \forall k > 1 \quad as_{i,j,k} \leq \sum_{o=1}^{k-1} ae_{i,j-1,o} \quad (3)$$

- Cohérence des événements de début et de fin de chaque opération : pour qu'une opération se termine au $k^{\text{ème}}$ événement, elle doit avoir débuté par une affectation à un événement antérieur (de 1 à $k-1$).

$$\forall i, \forall j > 1, \forall k > 1 \quad ae_{i,j,k} \leq \sum_{o=1}^{k-1} as_{i,j,o} \quad (4)$$

- Respect de la date de livraison échue : la mise à disposition de l'article peut avoir lieu dès la fin de la dernière opération, cette date étant définie par le membre de gauche de l'inégalité ci-dessous.

$$\forall i \sum_{k=1}^T ae_{i,n,k} \cdot t_k \leq L_i \quad \text{et} \quad \forall i \quad L_i^- \leq L_i \leq L_i^+ \quad (5)$$

- Affectation cohérente des postes : le poste dont le nombre de positions occupées initialement est connu, fait l'objet d'un bilan des actions qui débutent et qui s'achèvent lors de l'occurrence de chaque événement. La variable désignant le nombre de positions disponibles est contrainte par le nombre maximum de positions utilisables.

$$\forall j, \forall k > 1 \quad N_{j,k} = N_{j,k-1} - \sum_{i=1}^m as_{i,j,k} + \sum_{i=1}^m ae_{i,j,k} \quad \text{avec} \quad N_j^{\min} \leq N_{j,k} \leq N_j^{\max} \quad (6)$$

- Définition de la durée opératoire et des dates de réalisation : la durée de chaque opération est calculée comme la différence entre les dates de fin et de début d'opération, exprimée en fonction des dates des événements et des variables binaires d'affectation. Chaque durée est contrainte entre deux bornes. De plus, les événements sont classés par ordre chronologique

$$\forall i, \forall j \quad \sum_{k=1}^T ae_{i,j,k} \cdot t_k - \sum_{k=1}^T as_{i,j,k} \cdot t_k = c_{i,j} \quad \text{avec} \quad c_j^{\min} \leq c_{i,j} \leq c_j^{\max} \quad \text{et} \quad \forall k \geq 1 \quad t_{k-1} \leq t_k \quad (7)$$

- Relation entre l'effectif et le cycle sur chaque poste et pour chaque article : l'effectif requis pour exécuter une opération sur le poste est égale au rapport de la charge de travail par le cycle.

$$\forall i, \forall j \quad E_{i,j} \cdot c_{i,j} - \chi_j = 0 \quad (8)$$

- Détermination de l'effectif dans l'atelier : l'effectif total de l'atelier à chaque état (entre deux événements consécutifs) est calculé comme la somme des effectifs sur les postes entre les événements qui délimitent l'état. La variable E^{tot} interviendra dans la fonction coût, afin de chiffrer le coût total du travail.

$$E_0 = 0 \quad \text{et} \quad \forall k \quad E_k = E_{k-1} + \sum_{i=1}^m \sum_{j=1}^n as_{i,j,k} \cdot E_{i,j} - \sum_{i=1}^m \sum_{j=1}^n ae_{i,j,k} \cdot E_{i,j} \quad (9)$$

$$\forall k \quad 0 \leq E_k \leq E^{\text{tot}}$$

- Assignation effective des événements possibles et classement par ordre chronologique : la variable d'écart y_k prend la valeur 1 si aucune action n'est observée lors de l'événement k ; les variables d'écart sont classées par valeurs décroissantes afin de regrouper tous les événements non affectés (pour lesquels la variable d'écart est égale à 1) en début de liste. Si un événement est associé à une action, la variable d'écart prend une valeur nulle.

$$\forall k \quad 0 \leq y_k \leq 1 \quad \text{et} \quad \forall k \quad \begin{cases} \sum_{i=1}^m \sum_{j=1}^n as_{i,j,k} + \sum_{i=1}^m \sum_{j=1}^n ae_{i,j,k} + y_k \geq 1 \\ \sum_{i=1}^m \sum_{j=1}^n as_{i,j,k} + y_k \leq 1 \\ \sum_{i=1}^m \sum_{j=1}^n ae_{i,j,k} + y_k \leq 1 \end{cases} \quad \text{aussi} \quad \forall k > 1 \quad y_{k-1} \leq y_k \quad (10)$$

- Ecart minimal entre deux événements affectés : afin d'éviter que deux événements distincts ne soient associés à deux dates égales, nous définissons un écart minimal ε entre deux dates consécutives (la valeur de ε doit être petite par rapport aux cycles de production, nous avons retenu une valeur de 0,5 unités de temps). Si les événements ne sont pas affectés, ils peuvent être associés à des dates égales.

$$\forall k > 1 \quad t_k + \varepsilon \cdot (1 - y_k) \leq t_{k+1} \quad (11)$$

- Respect des dates de disponibilité : une opération ne peut commencer que lorsque les éléments matériels nécessaires (fournitures, approvisionnements) à sa réalisation sont disponibles sur le site de l'atelier. De plus, l'hypothèse d'une alimentation à débit constant permet de calculer l'ensemble des dates d'alimentation pour tous les articles et pour tous les postes à partir de la première date ($\delta_{1,1}$), en utilisant la durée entre deux approvisionnements dans l'atelier, noté α , et la durée entre l'approvisionnement de deux postes consécutifs, noté β . La date $\delta_{1,j}$ intervient dans le terme de la fonction objectif ; elle prendra donc la valeur maximale permettant de respecter la contrainte, afin de minimiser le coût total.

$$\forall i, \forall j \quad \sum_{k=1}^T a_{i,j,k} \cdot t_k \geq \delta_{i,j} \quad \text{avec} \quad \delta_{i,j} = \delta_{1,1} + \alpha \cdot (i-1) + \beta \cdot (j-1) \quad (12)$$

2-2-3-2. La fonction objectif à minimiser

La fonction objectif à optimiser est la somme de plusieurs coûts, exprimés en unités monétaires :

- F_1 : coûts liés au stockage amont (matières premières et approvisionnement des postes) ;
- F_2 : coûts liés aux en-cours et stocks intermédiaires ;
- F_3 : coûts liés au stockage de produits finis ;
- F_4 : pénalités de retard ;
- F_5 : coût du travail effectif ;
- F_6 : coût de l'inactivité ;
- F_7 : coût des perturbations.

Notre objectif est d'obtenir une solution optimale, non contrainte par des limites sociales, et de mesurer les effets de perturbations. En désignant par τ le taux d'actualisation, les coûts imputés à l'immobilisation d'un capital V durant une période d sont de la forme : $V \cdot [(1+\tau)^d - 1]$. Lorsque cette durée d est courte, nous pouvons nous livrer à l'approximation suivante : $V \cdot [(1+\tau)^d - 1] \approx V \cdot \tau \cdot d$, et lorsque τ est petit, l'approximation $\tau = \tau'$ est acceptable.

2-2-3-2-1. Les stocks de matière avant chaque poste (F1)

Les coûts liés à l'immobilisation de la matière première et des articles stockés avant leur entrée dans l'atelier sont calculés en fonction de la valeur de chacun des éléments, sur la durée comprise entre leur date de disponibilité et leur date d'entrée sur le poste correspondant.

$$\begin{aligned}
 F_1 &= \tau \cdot \sum_{i=1}^m \left(\sum_{j=1}^n w_{i,j} \cdot (d_{i,j} - \delta_{i,j}) + v_{i,0} \cdot (d_{i,1} - \delta_{i,1}) \right) \\
 &= \tau \cdot \sum_{i=1}^m \left(\sum_{j=1}^n w_{i,j} \cdot \left(\sum_{k=1}^T as_{i,j,k} \cdot t_k - (\delta_{i,1} + \alpha \cdot (i-1) + \beta \cdot (j-1)) \right) + v_{i,0} \cdot \left(\sum_{k=1}^T as_{i,1,k} \cdot t_k - (\delta_{i,1} + \alpha \cdot (i-1)) \right) \right)
 \end{aligned}
 \tag{13}$$

avec : d_{ij} : date de début de l'opération j pour l'article i ; $v_{i,0}$: valeur de l'article i avant le début de l'opération 1.

2-2-3-2-2. Les articles en cours de production et les stocks intermédiaires (F2)

Les encours sont calculés, article par article, entre la date d'entrée de l'article dans l'atelier ($d_{i,1}$) et sa date de mise à disposition ($f_{i,n}$), en prenant en compte l'évolution de sa valeur. Soit f_{ij} , la date de fin de l'opération j pour le poste i , on a :

$$\begin{aligned}
 F_2 &= \tau \cdot \left(\sum_{i=1}^m \sum_{j=1}^n (f_{i,j} - d_{i,j}) \cdot v_{i,j} + \sum_{i=1}^m \sum_{j=1}^{n-1} (d_{i,j+1} - f_{i,j}) \cdot v_{i,j} \right) \\
 &= \tau \cdot \left(\sum_{i=1}^m \left(\sum_{j=1}^{n-1} (d_{i,j+1} - d_{i,j}) \cdot v_{i,j} + (f_{i,n} - d_{i,n}) \cdot v_{i,n} \right) \right) \\
 &= \tau \cdot \left(\sum_{i=1}^m \left(\sum_{j=1}^{n-1} v_{i,j} \sum_{k=1}^T (as_{i,j+1,k} - as_{i,j,k}) \cdot t_k + v_{i,n} \cdot \sum_{k=1}^T (ae_{i,n,k} - as_{i,n,k}) \cdot t_k \right) \right)
 \end{aligned}
 \tag{14}$$

2-2-3-2-3. Stockage des produits finis avant la livraison (F3)

Le stockage des produits finis s'étale sur la période comprise entre la date de mise à disposition de l'article et sa date de livraison (L_i), au cours de laquelle la valeur n'évolue plus, et le risque d'obsolescence sera considéré comme négligeable.

$$F_3 = \tau \cdot \sum_{i=1}^m (L_i - f_{i,n}) \cdot v_{i,n} = \tau \cdot \sum_{i=1}^m \left(L_i - \sum_{k=1}^T ae_{i,n,k} \cdot t_k \right) \cdot v_{i,n}
 \tag{15}$$

2-2-3-2-4. Pénalités de retard (F4)

La fonction $P_i(t)$ est la loi de pénalité applicable pour un retard de livraison de l'article i ; elle peut varier selon le client. Elle est a priori croissante au sens strict.

$$F_4 = \int_0^{f_i - L_i} P_i(t) dt
 \tag{16}$$

2-2-3-2-5. Coût du travail (F5) et de l'inactivité (F6)

Les composantes permettant de chiffrer le coût du travail réellement effectué F_5 et l'inactivité F_6 , et peuvent être évaluées ensemble ou indépendamment l'une de l'autre. En effet, la somme de ces deux termes correspond à la rémunération du nombre maximum d'opérateurs entre le début et la fin du plan de production ; on peut ainsi écrire, en notant U le coût du travail par unité de temps :

$$F_5 + F_6 = (f_{m,n} - d_{1,1}) \cdot E^{tot} \cdot U = E^{tot} \cdot U \cdot \sum_{k=1}^T t_k \cdot (ae_{m,n,k} - as_{1,1,k})
 \tag{17}$$

Si l'on souhaite dissocier activité et inactivité, par exemple en leur affectant des coûts différents, notés respectivement U et S (si l'on considère que l'inactivité n'est pas totalement perdue mais peut être utilisée à d'autres activités, telles que formation, action d'amélioration, etc.), on obtient alors :

$$F_5 = U \cdot \sum_{k=1}^{T-1} E_k \cdot (t_{k+1} - t_k) \quad (18)$$

$$F_6 = S \cdot \sum_{k=1}^{T-1} (E^{tot} - E_k) \cdot (t_{k+1} - t_k) \quad (19)$$

2-2-3-2-6. Coût des perturbations (F7)

Chaque changement de cycle (durée opératoire) sur un poste entraîne une modification de la composition de l'équipe qui lui est affectée ; des perturbations sont ainsi générées selon le sens et l'amplitude de la variation. La diminution d'un cycle n'a pas la même influence sur le coût qu'une augmentation de même amplitude. Pour un poste j donné, le coût d'un changement de cycle entre deux articles consécutifs aura la forme suivante (où γ est un coefficient pondérateur caractérisant l'importance donnée aux perturbations et f désigne le coût de la perturbation entre les articles i et $i+1$) :

si $c_{i+1,j} > c_{i,j}$, alors

$$f_7 = U \cdot \gamma \cdot \chi_j \cdot \left(1 - \frac{c_{i,j}}{c_{i+1,j}} \right), \quad \text{et si } c_{i,j} > c_{i+1,j}, \text{ alors } f_7 = U \cdot \gamma \cdot \chi_j \cdot \left(1 - \frac{c_{i+1,j}}{c_{i,j}} \right)$$

et nous retiendrons finalement :

$$F_7 = U \cdot \gamma \cdot \sum_{i=1}^{m-1} \sum_{j=1}^n \chi_j \cdot \left(1 - \omega_{i,j} \cdot \frac{c_{i+1,j}}{c_{i,j}} - (1 - \omega_{i,j}) \cdot \frac{c_{i,j}}{c_{i+1,j}} \right) \quad (20)$$

2-3. Choix et méthodes de résolution

La modélisation aboutit à un problème de programmation non-linéaire en variables mixtes, sa résolution nécessite l'utilisation d'un code capable de traiter l'arborescence des solutions (capacité des algorithmes, temps de calcul, etc.). Les fonctions que nous devons optimiser ne présentent pas les propriétés mathématiques requises pour l'utilisation de méthodes classiques, efficaces et rigoureuses (linéarité, continuité, dérivabilité, etc.). Nous nous sommes donc orientés vers des métaheuristiques, parmi lesquelles nous avons choisi d'utiliser les algorithmes génétiques qui fournissent une réponse efficace au traitement de l'aspect fortement combinatoire de ce type de problèmes, et font partie des méthodes les mieux adaptées lorsque la fonction n'est ni continue, ni dérivable. Ils présentent l'avantage de traiter simultanément un grand nombre de solutions potentielles (l'ensemble des *chromosomes* d'une génération) qui explorent aléatoirement l'ensemble de l'espace des solutions possibles, et permettent généralement d'obtenir une réponse de bonne qualité, voisine d'un optimum démontré.

2-4. Résolution du problème

Nous envisageons une approche séquentielle, caractérisée par la décomposition du problème en sous-problèmes, optimisant chacun un jeu de variables en se fondant sur une fonction objectif appropriée.

Cette décomposition conduit à la résolution séparée : d'une part, du problème de l'optimisation des dates de livraison (on désignera par « date de livraison » la date d'achèvement du programme d'activités étudié), et d'autre part, du problème d'optimisation des durées opératoires.

2-4-1. Décomposition en sous-problèmes : Algorithme général de résolution

Les ensembles de variables distinctes (dates de livraison et cycles de production), précédemment mis en évidence, permettent de décomposer le problème en sous-problèmes : chacun d'entre eux correspond à une optimisation locale de chaque groupe de variables. La procédure est la suivante :

- optimisation des dates de livraison, les cycles étant constants et fixés, en minimisant les perturbations ;
- optimisation des cycles, en utilisant comme données les dates de livraison obtenues à l'étape 1, et en minimisant la fonction coût (les dates de livraison sont alors fixées) ;
- si la solution après l'étape 2 diffère de la solution avant celle-ci (en quoi diffère-t-elle si l'étape 1 se focalise sur des dates de livraison qui sont respectées par l'étape 2 ?), retourner à l'étape 1, sinon arrêter, puisque aucune amélioration n'a été possible.

2-4-2. Résolution successive des sous problèmes

2-4-2-1. Détermination des dates de livraison optimales

La première étape de la résolution est la détermination de la date de livraison $L(i)$ optimale pour chaque ordre de fabrication (programme d'activités) i , à l'intérieur de l'intervalle de tolérance défini avec le client. Les valeurs des cycles sont alors des données du problème (égales pour chaque poste à la valeur minimale du cycle autorisé), elles ne seront donc pas modifiées au cours de cette étape. Le nombre de variables $L(i)$ est égal au nombre d'ordre de fabrications à traiter, et les valeurs discrètes autorisées pour $L(i)$ sont les entiers appartenant à l'intervalle de tolérance pour la date de livraison de l'article i . On peut ainsi calculer la planification au plus tard en fonction de ces dates, et en déduire le plan de charge des effectifs : la durée de passage retenue étant la durée minimale pour chacun des postes, il en résulte les plans de charge correspondant à une mobilisation maximale des effectifs. La démarche de détermination de la planification et du plan de charge des effectifs suit l'algorithme suivant :

« $m =$ nombre d'articles

$n =$ nombre de postes

$Eff(i,j) =$ effectif sur le poste j pour l'article i

$E(k) =$ effectif total pour le jour k

$T =$ durée de l'horizon de temps

Pour $i \leftarrow m$ à 1 par pas de -1 faire

$f(i,n) \leftarrow L(i)$

$L(i)$: date de livraison de l'article i

Pour $j \leftarrow n$ à 2 par pas de -1 faire

$d(i,j) \leftarrow f(i,j) - c(i,j)$

$c(i,j)$: cycle de l'article i sur le poste j

$f(i,j-1) \leftarrow d(i,j)$

```

    fin pour
     $d(i,1) \leftarrow f(i,1) - c(i,1)$ 
  fin pour
  Pour  $i \leftarrow 1$  à  $m$  faire
    Pour  $j \leftarrow 1$  à  $n$  faire
      de  $k \leftarrow d(i,j)$  à  $f(i,j)$  faire
         $E(k) \leftarrow E(k) + \text{Eff}(i,j)$ 
        où  $\text{Eff}(i,j) = \chi_{ij} / c_{ij}$ 
      fin faire
    fin faire
  fin faire »

```

Si nous appelons G la fonction d'évaluation correspondant aux perturbations du plan de charge, on peut alors en déterminer la valeur, puis la pénaliser si la solution proposée entraîne la violation de contraintes d'occupation des postes (càd si deux livraisons L_i et L_{i+1} sont prévues à un intervalle inférieur au plus long des cycles de l'article i , M_i) :

```

«  $G \leftarrow 0$ 
  Pour  $k \leftarrow 1$  à  $T-1$  faire
     $G \leftarrow G + |E(k+1) - E(k)|$ 
  fin pour

  Pour  $i \leftarrow 1$  à  $m-1$  faire
    si  $L_{i+1} - L_i < M_i$ , alors
       $G \leftarrow G + K * (L_i - L_{i+1} + M_i)$ 
    fin si
  fin pour »

```

Le coefficient K ci-dessus est un coefficient de pénalisation ; il permet de dégrader les solutions violant cette contrainte, sans toutefois les éliminer totalement : il est essentiel lors de l'utilisation d'algorithmes génétiques de préserver, au moins durant les premières générations, des solutions « décevantes » afin de conserver une certaine diversité dans la population [22], gage d'exhaustivité dans l'exploration du domaine des solutions. La meilleure solution obtenue est alors utilisée pour le calcul des cycles associés à chaque poste et à chaque ordre de fabrication.

2-4-2-2. Modulation des durées opératoires

On considère maintenant que les dates de livraison obtenues lors de l'étape précédente sont figées, et l'on fait varier la durée opératoire $c(i,j)$ sur chaque poste et pour chaque ordre de fabrication. Le nombre de ces variables $c(i,j)$ est alors égal au produit du nombre d'articles par le nombre de postes de travail. Les valeurs discrètes pouvant être prises par $c(i,j)$ sont des données du problème. On calcule tout d'abord la planification, en respectant cette fois les contraintes de capacité :

```

«  $f(m, n) \leftarrow L(n)$ 
  POUR  $j \leftarrow n$  à 2 par pas de -1 FAIRE
     $d(m, j) \leftarrow f(m, j) - c(m, j)$ 

```

```

    f(m,j-1) ← d(m, j)
  FIN POUR
  d(m,1) ← f(m,1) - c(m,1)
  POUR i ← m-1 à 1 par pas de -1 FAIRE
    f(i, n) ← min(L(i), Li+1 - Mi)
    POUR j ← n à 2 par pas de -1 FAIRE
      d(i, j) = f(i, j) - c(i, j)
      f(i,j-1) = d(i, j)           (pas de stockage intermédiaire)
    FIN POUR
    d(i,1) ← f(i,1) - c(i,1)
  FIN POUR
  POUR i ← 1 à m FAIRE
    POUR j ← 1 à n FAIRE
      POUR k ← d(i, j) à f(i, j) FAIRE
        E(k) ← E(k) + Eff(i, j)
      FIN POUR
    FIN POUR
  FIN POUR »
  
```

On cherche alors à minimiser la fonction coût F. La valeur de F pour chaque chromosome est calculée après avoir déterminé la planification et le plan de charge, en pénalisant les solutions pour lesquelles le niveau maximal autorisé pour les effectifs est dépassé. De plus, lorsqu'il existe des contraintes sur la disponibilité des matières en stock avant chaque poste, les solutions violant ces contraintes sont pénalisées, proportionnellement à l'ampleur de cette violation. En appliquant séquentiellement les principes développés ci-dessus, il est possible de résoudre le problème. La **Figure 3** ci-dessous (où l'opérateur « > » signifie « meilleure que ») présente le principe de fonctionnement de l'algorithme.

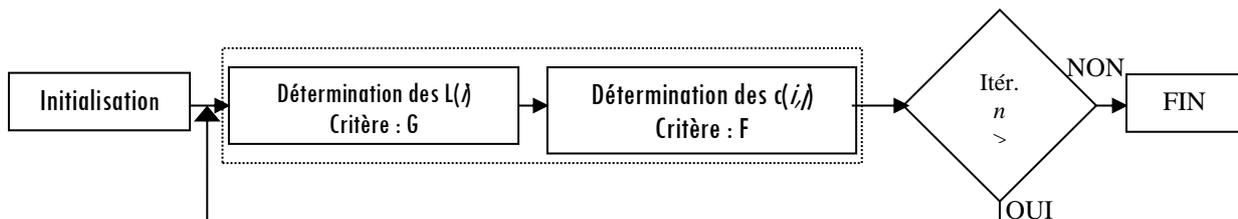


Figure 3 : Organigramme des itérations

Après la phase d'initialisation, qui voit la constitution de la génération initiale, la répétition des itérations, constituées de la détermination successive des dates de livraison et des durées opératoires, se poursuit tant que le critère F est amélioré. Si celui-ci n'est pas amélioré par la dernière itération, l'algorithme se termine en retenant la meilleure solution obtenue.

3. Résultats et discussion

3-1. Données du problème

La formulation décrite est appliquée à la résolution d'un exemple ; l'atelier étudié est constitué de deux postes de travail, P1 et P2, sur lesquels doivent être traités dix articles, suivant une gamme identique. Chaque article doit d'abord passer sur le poste P1 avant de passer sur le poste P2 ; de plus, chaque poste ne compte qu'une seule position. L'ensemble des pièces devant être intégrées dans l'atelier arrivent dans le stock amont selon un principe « à débit constant » ; le rythme d'arrivée (intervalle entre deux arrivées) est connu *a priori*, mais la date d'arrivée du premier lot dans le stock ($\delta_{1,1}$) est une variable, qui détermine l'arrivée de tous les autres lots. Les valeurs discrètes autorisées pour les cycles des postes P1 et P2 sont au nombre de deux, et sont présentées dans le **Tableau 1**, ainsi que les charges de travail sur chaque poste (en hommes.unités de temps) :

Tableau 1 : Cycles et charge de travail par poste

	Poste P1	Poste P2
Cycle minimum (en u. t.) c_j^{\min}	6	5
Cycle maximum (en u. t.) c_j^{\max}	8	8
Charge de travail (en h. u. t.)	48	60

De plus, l'effectif total disponible sur l'atelier ne peut en aucun cas dépasser 15 opérateurs, qui est donc une borne supérieure pour L^{tot} . Les intervalles autorisés pour la livraison de chacun des articles sont énumérés dans le **Tableau 2** ci-dessous.

Tableau 2 : Date de livraison théorique et intervalles de livraison pour chaque article

N° de l'article	L_i	L_i^p	L_i^+
1	26	28	30
2	35	37	39
3	38	40	42
4	55	57	59
5	65	67	69
6	73	75	77
7	78	80	82
8	87	89	91
9	93	95	97
10	101	103	105

3-2. Interprétation

pour l'exemple considéré, la solution est obtenue en trois itérations « utiles » (la quatrième ne servant qu'à confirmer que la solution ne peut pas être améliorée). Les résultats intermédiaires sont donnés dans les **Tableaux 3 à 8** ci-dessous et sur les **Figures 4 à 6** (diagramme de Gantt), où les flèches en pointillés représentent le stockage avant livraison.

3-2-1. * Itération 1

Tableau 3 : Solution intermédiaire, itération 1

N° de l'article	Dates de livraison	Durée poste P1	Durée poste P2
1	27	8	8
2	36	8	8
3	42	8	8
4	55	8	8
5	66	8	8
6	73	8	8
7	79	8	8
8	90	8	8
9	96	8	8
10	103	8	5

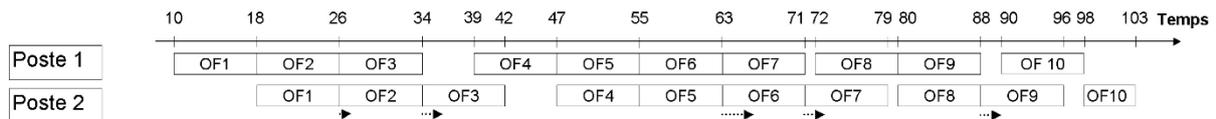


Figure 4 : Diagramme de Gantt pour l'itération 1

Tableau 4 : Fonction coût, itération 1

F ₁	F ₂	F ₃	F ₄	F ₅	F ₆	F ₇	F
3 150	18 750	1 500	0	2 224	344	180	26 156

La première itération permet l'obtention d'une solution satisfaisant les contraintes sur l'occupation des postes, ainsi que sur le niveau des opérateurs disponibles. Les durées opératoires sont constantes, à l'exception du dernier article sur le dernier poste (*Tableau 3*); le terme F7 (perturbations) a donc une valeur très faible (*Tableau 4*). En contrepartie, certains articles seront stockés entre la fin du dernier poste et leur livraison ; le terme F3 prend alors une valeur élevée, par rapport au stockage avant entrée dans l'atelier de production (F1).

3-2-2. * Itération 2

Tableau 5 : Solution intermédiaire, itération 2

N° de l'article	Dates de livraison	Durée poste P1	Durée poste P2
1	26	8	8
2	35	8	8
3	42	8	5
4	57	6	8
5	65	8	8
6	73	8	8
7	81	8	8
8	89	8	8
9	97	8	8
10	104	8	5

Tableau 6 : Fonction coût, itération 2

F ₁	F ₂	F ₃	F ₄	F ₅	F ₆	F ₇	F
4 080	18 120	0	0	2 224	380	732	25 536

La deuxième itération permet l'obtention d'une solution meilleure que la précédente (*Tableau 6*). L'amélioration porte notamment sur le terme F₃, dans la mesure où l'on n'observe plus de stockage de produits finis. De plus, les encours sont moins importants, puisque la modulation des durées opératoires (*Tableau 6*) permet d'utiliser le cycle minimum sur les postes 1 et 2, respectivement pour les articles 4 et 3 ; ceci correspond, comme dans la solution optimale obtenue dans [1], à une « fermeture » successive de chacun de ces postes, plutôt qu'à un travail simultané, comme dans la solution à débit constant. Le diagramme de Gantt correspondant à l'itération 2 est présenté en *Figure 5*.

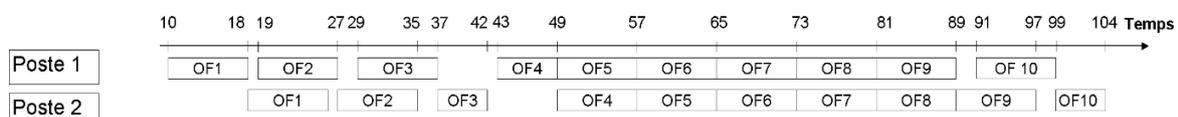


Figure 5 : Diagramme de Gantt de l'itération 2

3-2-3. * Itération 3

Tableau 7 : Solution, itération 3

N° de l'article	Dates de livraison	Durée poste P1	Durée poste P2
1	27	8	8
2	35	8	8
3	42	8	5
4	56	6	8
5	65	8	8
6	73	8	8
7	81	8	8
8	89	8	8
9	97	8	8
10	104	8	5

Tableau 8 : Fonction coût, itération 3

F ₁	F ₂	F ₃	F ₄	F ₅	F ₆	F ₇	F
4 080	18 120	0	0	2 224	352	732	25 508

Les durées opératoires (*Tableau 7*) ne varient pas par rapport à la solution précédente. Cependant, une légère modification de la date de livraison pour l'article 1 (retardé de 1 unité de temps) permet de réduire la durée totale de la période travaillée ; or, celle-ci intervient dans le calcul des termes F₅ (coût du travail) et F₆ (coût de l'inactivité), ce qui permet d'améliorer la fonction coût. Le diagramme de Gantt de cette solution est présentée sur la *Figure 6*.

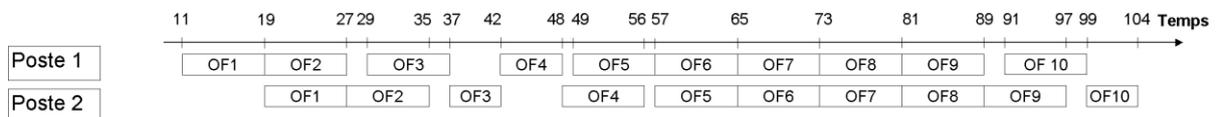


Figure 6 : Diagramme de Gantt de l'itération 3

La troisième et dernière itération donne la solution qu'on ne peut plus améliorer ; l'utilisation d'algorithmes génétiques sur le problème considéré permet donc l'obtention d'une solution intéressante ; cependant, ce résultat n'est pas systématique : en effet, ce type de métaheuristiques ne présente aucune garantie quant à l'obtention de la solution optimale. Il est donc courant de n'obtenir que des solutions sous-optimales, ceci dépendant bien sûr de l'occurrence du problème traité. Notamment, si le calendrier de livraison, ou les intervalles de tolérance, étaient modifiés. Néanmoins, la rapidité des calculs (une itération prenant environ 30 secondes) permet de lancer successivement plusieurs « boucles de résolution », afin de confronter l'ensemble des résultats obtenus et de retenir la meilleure solution.

4. Conclusion

L'utilisation d'algorithmes génétiques pour la résolution du problème envisagé précédemment permet l'obtention d'une solution optimale par rapport à la fonction coût, sur l'instance considérée. Il est cependant probable que d'autres jeux de données pour le même problème ne permettront pas d'obtenir la solution optimale, mais seulement des solutions sous-optimales ; néanmoins, la bonne qualité des résultats permet de penser que les solutions seront satisfaisantes. De plus, la rapidité des calculs permet de réaliser plusieurs cycles d'optimisation, afin de se placer dans les meilleures conditions d'obtention de la solution optimale, en utilisant ainsi les propriétés de convergence stochastique des algorithmes génétiques.

5. Références

- [1] - A. EL AWADY, K. H. EDI and P. DUQUENNE, "Flexible resources allocation techniques : characteristics and modelling", *Int. J. Operational Research*, Vol. 14, No 2 (2012) 221- 254.
- [2] - Kouassi Hilaire EDI et Philippe DUQUENNE «Un modèle à événements pour étudier la flexibilité opérationnelle d'un flow shop flexible». *Afrique Science*, Vol.11, N°1 (2015) 59-73, 1 janvier 2015, <http://www.afriquescience.info/document.php?id=4218>. ISSN 1813-548X.
- [3] - L. FRANCHINI, « Aide à la décision pour la gestion des opérateurs en production : modélisation, Planification et Evaluation » , Thèse de doctorat de l'Institut National Polytechnique de Toulouse, (2000).
- [4] - E. VIDAL, « Stratégie de planification pour la flexibilité opérationnelle d'un atelier à cheminement unique », Thèse de doctorat, Institut National Polytechnique de Toulouse, (2000).
- [5] - R. L. DANIELS, J-B. MAZOLLA, D. SHI, « Flow shop scheduling with partial resource flexibility » , *Management science*, 50(1) (2004) 658-669.
- [6] - A. HAÏT et P. BAPTISTE, « Ordonnancement d'atelier et ressources humaines : affectation des opérateurs dans un flowshop », *Journal Européen des systèmes automatisés*, (2005).
- [7] - P. BAPTISTE, V. GIARD, A. HAÏT et F. SOUMIS, « Gestion de production et ressources humaines - Méthodes de planification dans les systèmes productifs », Presses Internationales Polytechnique, Montréal, Juin (2005).
- [8] - C. ARTIGUES, M. GENDREAU and L-M. ROUSEAU, « A flexible model and hybrid exact method for integrated employee timetabling and production scheduling » , *Proceedings of the 6th international conference on Practice and theory of automated timetabling*, Brno, Czech Republic, (2006) 64-81.
- [9] - R. LOGENDRAN, S. CARSON and E. Hanson, « Group scheduling in flexible flow shops », *International Journal of Production Economics*, 96(2) (2005) 143-155.
- [10] - R. LOGENDRAN, P. SZOEKE and F. BERNARD, « Sequence-dependent group scheduling problems in flexible flow shops », *International Journal of Production Economics*, (2006).
- [11] - D. ALISANTOSO, LP. KHOO and PY JIANG, « An immune algorithm approach to the scheduling of a flexible PCB flow shop », *Int J Adv Manuf Tech* 22(11-12) (2003) 819-827.
- [12] - C. SRISKANDARAJAH and S. P. SETHI, « cheduling algorithms for flexible flow shops: worst and average case performance » , *European Journal of Operational Research.*, (1989).
- [13] - R. LOGENDRAN, S. CARSON. and E. HANSON, « Group scheduling in flexible flow shops », *International Journal of Production Economics*, 96(2), pp. 143-155, (2005).
- [14] - E. MARY and R. G. ASKIN, « Scheduling flexible flow lines with sequence dependent set-up times », *European Journal of Operational Research*, 159 : 66-82, (2003).
- [15] - M. E. KURZ, M. RUNKLE and S. PEHLIVAN, « Comparing problembased-search and random keys genetic algorithms for the SDST FFL makespan scheduling problem », working paper, (2005).

- [16] - M. ZANDIEH, S. M. T. FATEMI GHOMI and S. M. MOATTAR HUSSEINI, « An immune algorithm approach to hybrid flow shops scheduling with sequence-dependent setup times », *Appl Math Comput* 180, 111- 127, doi :10.1016/j.amc.2005.11.136, (2006).
- [17] - J. BLAZEWICZ, K. H. ECKER, E. PESCH, G. SCHMIDT and J. WEGLARZ, « Handbook of scheduling from theory to applications ». Berlin, Heidelberg: Springer, (2007).
- [18] - G. J. KYPARISIS and C. KOULAMAS, « Flexible flow shop scheduling with uniform parallel machines », *European Journal of Operational Research* 168 (3), 985-997, (2006).
- [19] - T-P. HONG, P-Y. HUANG, G. HORNG and C-L. WANG, « Three Algorithms for Flexible Flow-shop Scheduling », *American Journal of Applied Sciences* 4 (11): 887-895, ISSN 1546-9239, (2007).
- [20] - O. SHAHVARI, N. SALMASI and R. LOGENDRAN, « A Meta-Heuristic Algorithm for Flexible Flow Shop Sequence Dependent Group Scheduling Problem », *Proceedings of the International Conference on Value Chain Sustainability*, (2009).
- [21] - D. E. GOLDBERG, « Real-coded Genetic Algorithms, virtual alphabets and blocking », *Complex Systems*, P. 139-167, (1991).
- [22] - D. DELAHAYE, « Optimisation de la sectorisation de l'espace aérien par algorithmes génétiques », *Thèse de doctorat, Ecole Nationale Supérieure de l'Aéronautique et de l'Espace, Toulouse, France*, (1995).