

COMPLEMENTARITE DES APPROCHES ANALYTIQUE ET SIMULATOIRE POUR LE DIMENSIONNEMENT D'UN ATELIER MANUFACTURIER

Samieh MIRDAMADI, Rémi AUTEFAGE, Franck FONTANILI, Lionel DUPONT

Centre de Génie Industriel
Ecole des Mines d'Albi
Campus Jarlard
81013 ALBI

samieh.mirdamadi@enstimac.fr, remi.autefage@enstimac.fr, franck.fontanili@enstimac.fr, lionel.dupont@enstimac.fr

RESUME : *L'objectif de ce travail est de présenter une démarche faisant appel de façon complémentaire à deux approches souvent opposées pour le dimensionnement d'un atelier de production manufacturier : l'approche analytique et l'approche simulateur. La simulation de flux à événements discrets utilisée seule entraîne souvent de nombreuses expérimentations pour trouver une solution acceptable. En utilisant d'abord la résolution analytique, il est par exemple très rapide de choisir parmi plusieurs gammes celles qui permettent de respecter les délais de production et d'équilibrer le taux d'occupation des différents postes de travail. La difficulté majeure se situe dans la phase de modélisation, puisque le problème doit être décrit par un système d'équations. La résolution à l'aide d'un solveur fournit une solution optimale par rapport à l'objectif fixé. En complément, la simulation peut permettre non seulement de vérifier les résultats fournis par la résolution analytique, mais aussi d'apporter des informations complémentaires en tenant compte de la dynamique des flux et pas seulement des capacités statiques. Par ailleurs, la simulation peut prendre en compte des phénomènes aléatoires, alors que la résolution analytique s'appuie sur des données déterministes. La simulation apporte par exemple des informations plus précises sur la capacité des stocks entre les machines. Pour présenter cette démarche, nous proposons de nous appuyer sur une étude de cas d'un atelier manufacturier de type job shop.*

MOTS-CLES : *optimisation, simulation de flux à événements discrets, résolution analytique, dimensionnement.*

1. INTRODUCTION

Les objectifs de performance que l'on souhaite atteindre en production sont souvent contradictoires. Il est difficile de trouver une solution optimale qui satisfasse tous les critères de performance, comme par exemple un niveau de stock très faible, des coûts réduits et une réponse rapide à la demande du client.

Cette complexification des systèmes de production nécessite l'utilisation de méthodes d'optimisation. Ces méthodes doivent servir d'aide à la décision pour dimensionner les nouveaux équipements et déterminer les paramètres de pilotage appropriés en fonction des critères de performance souhaités. En illustrant notre propos avec l'exemple d'un atelier de type job shop, nous proposons une démarche qui utilise la complémentarité d'une résolution analytique et simulateur.

Dans un premier temps, la résolution analytique permet de cibler très rapidement des populations de solutions proches de l'optimum et de réduire le nombre de combinaisons à tester. Elle donne des renseignements statiques sur les éléments, indépendamment les uns des autres, comme par exemple la capacité de chaque

machine, la capacité des stocks d'encours, les gammes à utiliser, etc. A ce stade de l'étude, ni les phénomènes aléatoires dans le système, ni la dynamique des flux ne sont pris en compte et la solution est basée sur des données déterministes. On peut qualifier cette résolution de *statique* ou *capacitaire*.

Dans un second temps, il est nécessaire d'intégrer les interactions entre les différents éléments du système pour comprendre son comportement dynamique. La simulation de flux est un des outils capable, non seulement de vérifier la solution proposée par la résolution analytique, mais aussi de fournir les informations manquantes sur la dynamique des flux en prenant en compte des données stochastiques et des événements aléatoires.

L'un des intérêts de commencer par une résolution analytique est d'obtenir certaines données utiles à la construction du modèle et d'autres données pour alimenter les simulations proprement dites. Sans résolution analytique préalable, le nombre de solutions à tester avec la simulation peut s'avérer très important et entraîner de ce fait un grand nombre de simulations et donc un temps de résolution très long. En effet, pour les systèmes de production complexes, la simulation de flux

est un travail très fastidieux, qui monopolise des ressources tant pour sa modélisation que pour les nombreux essais de simulation qui s'en suivent. La réduction de l'espace de solutions par résolution analytique permet un gain de temps non négligeable, compte tenu de l'ampleur de la tâche, en sélectionnant *a priori* les solutions optimales par rapport à des critères et des contraintes prédéfinis.

Dans cet article, nous montrerons au travers d'un exemple tout l'intérêt du couplage ou de la complémentarité de ces deux méthodes. Dans le chapitre 2, nous présentons l'état de l'art sur les deux méthodes et leur complémentarité. Ensuite, une application sur la conception d'un atelier de type job-shop est présentée. Nous présenterons le cas d'étude que nous avons retenu. Nous détaillerons la démarche adoptée pour utiliser conjointement les résultats du modèle analytique et du modèle de simulation de flux. Enfin, nous analyserons les résultats et concluons sur l'intérêt de cette démarche.

2. ETAT DE L'ART

2.1. Les méthodes analytiques

Les méthodes analytiques cherchent à représenter une situation réelle par un modèle mathématique, puis à identifier les meilleures décisions dans le modèle en espérant qu'elles seront aussi les meilleures dans la réalité. La situation est modélisée par des contraintes, dont le but est de séparer le possible de l'impossible. Usuellement, les décisions sont évaluées à l'aide de fonctions objectifs ou économiques. Cela revient donc à trouver le minimum (ou le maximum) d'une fonction objectif parmi les solutions qui respectent les contraintes. Il existe de nombreuses méthodes qui dépendent fortement de la fonction objectif, de la nature des contraintes et du type des variables de décision (Minoux, 1983).

Pour les problèmes qui ne sont pas NP-complets, la résolution est faite par un algorithme polynomial, à condition que le polynôme soit de degré raisonnable. Certains problèmes ont de « bonnes » caractéristiques qui permettent de les résoudre à l'aide d'une formule de récurrence. Les méthodes de programmation dynamique peuvent alors éventuellement permettre de résoudre le problème avec une complexité polynomiale ou pseudo-polynomiale.

Les processus stochastiques concernent tous les problèmes aléatoires, en particulier des problèmes de fiabilité (de systèmes, de composants électroniques...) et des phénomènes d'attente. La programmation linéaire est très souvent utilisée pour résoudre des problèmes combinatoires. Elle permet de résoudre les problèmes dans lesquels les variables sont continues. Lorsqu'il y a des variables discrètes, programmation linéaire et méthodes arborescentes peuvent être combinées.

La programmation non linéaire peut aussi être utilisée. La possibilité de modéliser des contraintes ou des fonctions objectifs non linéaires offre une puissance de modélisation très importante, mais les algorithmes de résolution des programmes non linéaires sont significativement moins efficaces que ceux de la programmation linéaire. Les méthodes de type A* ou *branch and bound* sont couramment mises en œuvre pour trouver la solution exacte d'un problème de recherche opérationnelle. Pour une résolution efficace, un soin particulier est apporté au calcul des bornes supérieures ou inférieures pour la valeur de la solution. La programmation par contraintes permet de mettre en œuvre rapidement et efficacement de telles méthodes de recherche arborescente. Plusieurs bibliothèques (logiciels) d'optimisation, commerciales ou non, reposent sur cette approche (ILOG Solver, Chip, Mozart/Oz, FaCiLe). De nombreux logiciels d'optimisation de problèmes réels utilisent ainsi cette technologie.

Lorsque la solution optimale ne peut être obtenue en un temps raisonnable, on a souvent recours à des méthodes approchées de type heuristique ou métaheuristique. De nombreux travaux ont été effectués pour étudier les algorithmes et heuristiques, en particulier pour l'optimisation de l'ordonnancement, pour des jobs shops statiques et dynamiques (Gere, 1966), (Blackstone *et al.*, 1982), (Rajendran et Holthaus, 1999), (Jain et Meeran, 1999).

Un problème d'ordonnancement peut se caractériser par une série de jobs, chacun composé d'une ou plusieurs opérations. Les opérations doivent être effectuées dans un ordre déterminé sur des machines déterminées. L'objectif de l'ordonnancement est de trouver l'ordre des jobs qui minimise (ou maximise) un critère (ou plusieurs) de performance (Rajendran et Holthaus, 1999). A cause du nombre considérable de combinaisons possibles, les problèmes d'ordonnancement de job shop sont dits NP-difficiles (Jain et Meeran, 1999).

La résolution des problèmes NP-complets est coûteuse en termes de puissance de calcul, dans la mesure où pour garantir une solution optimale cela nécessite une recherche exhaustive où toutes les solutions possibles doivent être essayées et évaluées. Comme une telle recherche exhaustive prend un temps de calcul démesuré pour les problèmes plus complexes, il est courant de sacrifier l'optimalité de la solution à l'efficacité de la recherche en guidant par une heuristique la recherche de solutions et en évaluant seulement une fraction de toutes les combinaisons (Arnaout et Rabadi, 2005).

2.2. Les méthodes simulateurs

En phase de conception, par exemple pour définir les règles optimales de pilotage, le système de décision doit posséder des informations fiables, complètes et précises sur le système de production. Or, le modèle analytique n'est pas suffisant pour prendre les décisions qui

s'imposent pour concevoir ou optimiser les performances d'un atelier de fabrication (Cardin et Castagna, 2006).

La simulation permet de modéliser des systèmes sans utiliser un langage mathématique, mais plutôt en manipulant des éléments (machines, ressources, stocks, etc.) caractérisés par des états (disponible, occupé, etc.) qui vont nous permettre de représenter la réalité. La dynamique est simulée grâce à une horloge et des événements qui modifient les états de ces éléments. Quand aucun modèle analytique n'existe, la simulation fournit des résultats qui sont ensuite analysés pour permettre une optimisation. Une optimisation est donc possible, cependant elle peut nécessiter une grande quantité de simulations en modifiant à chaque fois un grand nombre de données, ce qui rend fastidieux le travail de simulation. En effet, la complexité croissante des systèmes de production fait que souvent, aucun modèle analytique satisfaisant n'existe. C'est pourquoi de nombreux travaux cherchent à coupler l'utilisation de la simulation à une résolution analytique pour faire de l'optimisation en utilisant des méthodes approchées, comme les métaheuristiques.

Les métaheuristiques sont généralement des algorithmes stochastiques itératifs, qui progressent vers un optimum global, c'est-à-dire l'extremum global d'une fonction, par échantillonnage d'une fonction objectif. Elles se comportent comme des algorithmes de recherche, tentant d'apprendre les caractéristiques d'un problème afin d'en trouver une approximation de la meilleure solution (Collet et Rennard, 2006).

Les métaheuristiques les plus classiques sont celles fondées sur la notion de parcours. Dans cette optique, l'algorithme fait évoluer une seule solution sur l'espace de recherche à chaque itération. La notion de voisinage est alors primordiale. Les plus connues dans cette classe sont le *recuit simulé* (Prudius et Andradottir, 2005), la méthode *Tabou* (Glover et Laguna, 1997) (Nowicki et Smutnicki, 1996), la recherche à *voisinage variable* (les stratégies d'évolution) (Rechenberg, 1973), la méthode *GRASP* (Feo et Resende, 1988) ou encore les *méthodes de bruitage*.

Dans cette classification, l'autre approche utilise la notion de population. La métaheuristique manipule un ensemble de solutions en parallèle, à chaque itération. On peut citer les *algorithmes génétiques* (Holland, 1975) (Goldberg, 1989), (Koza, 1992), l'optimisation par *essais particuliers* (Chong et Yoke Hean Low, 2006), les algorithmes de *colonies de fourmis*.

La recherche dans ce domaine étant très active, il est impossible de produire une liste exhaustive des différentes métaheuristiques d'optimisation. La littérature spécialisée montre un grand nombre de variantes et d'hybridations entre toutes ces méthodes, particulièrement dans le cas des algorithmes évolutionnaires pour faire de l'optimisation multi-

objectif (Deb, 2001). La littérature donne quelques exemples d'optimisation multi-objectif basée sur la simulation, par exemple dans (Eskandari *et al.*, 2005) et (Baesler et Sepúlveda, 2001). Seule une petite fraction des travaux concernent les problèmes de pilotage des opérations.

(Almeida *et al.*, 2001) utilisent une approche d'optimisation multi-objectif des paramètres de pilotage via la simulation dans une raffinerie de pétrole. La méthode qu'ils proposent est basée sur un algorithme génétique. En utilisant cette technique, les auteurs ont réussi à trouver une solution pour un problème réel de production en raffinerie avec trois objectifs : maximiser la production de diesel, maximiser la production de fuel et minimiser les coûts.

(Allaoui et Artiba, 2004) propose une méthode basée sur la combinaison du recuit simulé et de règles de dispatching pour l'optimisation multi-objectif d'un flow shop via simulation. L'indisponibilité stochastique et déterministe des machines est considérée dans la stratégie d'optimisation. Les auteurs ont appliqué cette méthode pour résoudre un problème d'ordonnement NP-difficile en optimisant les encours, le makespan et l'utilisation des ressources.

(Gupta et Sivakumar, 2002) présentent une méthode d'optimisation multi-objectif pour le pilotage de la fabrication de semiconducteurs via simulation basée sur de la programmation linéaire. La méthode proposée a été utilisée pour trouver une solution *Pareto optimale* à un problème NP-difficile d'ordonnement de tâches indépendantes sur une machine simple. Les objectifs à optimiser étaient le temps de cycle moyen et l'utilisation des machines. Un certain nombre d'études théoriques de type job shop ont été menées avec succès en utilisant cette méthode.

2.3. La combinaison des deux approches

La littérature met en évidence que la simulation est le plus souvent utilisée en premier lieu pour construire le modèle analytique. La phase d'optimisation intervient ensuite pour maximiser ou minimiser la fonction objectif.

Pourtant, il nous semble intéressant d'adopter la démarche inverse, qui consiste à construire le modèle de simulation de flux en s'aidant des résultats d'une résolution analytique.

(Hicks, 1999) a proposé une méthodologie en quatre étapes pour utiliser la simulation et l'optimisation pour la planification stratégique d'une chaîne logistique. Elle comporte d'abord une étape d'optimisation avec un solveur mixte pour dégager une structure générale du réseau qui satisfait les besoins prévus, minimise les coûts de structure et supporte les contraintes managériales prévues. Elle permet de trouver un optimum global très

rapidement. Par contre, la solution ne donne qu'une image très approximative du système réel. La deuxième étape est une étape de simulation de flux du réseau par événements discrets. L'optimisation apporte une aide dans les choix de conception, mais ne donne aucune information sur le comportement dynamique. Par contre, la simulation de flux fournit des informations sur le comportement et peut aider dans le choix des règles de pilotage. La troisième étape consiste à optimiser les règles de pilotage pour définir les meilleures. L'auteur propose ici d'utiliser l'optimisation via la simulation selon les techniques classiques que l'on trouve dans la littérature et qui ont été détaillées précédemment. Enfin, la quatrième étape consiste à vérifier la robustesse du modèle proposé en variant les cas d'utilisation.

A l'instar de (Hicks, 1999), nous proposons dans cet article une démarche qui combine les deux approches. La méthode analytique sera d'abord utilisée pour choisir, parmi plusieurs gammes possibles, celles qui permettent de satisfaire les délais. Pour alimenter la simulation, une étape intermédiaire consistera à choisir une loi de distribution d'arrivée des gammes en fonction du temps. Nous utiliserons un tableur pour générer cette séquence d'arrivée des gammes. Ensuite, la simulation prendra le relais, alimentée avec le séquencement créé dans le tableur, d'abord pour vérifier les résultats donnés par la résolution analytique en termes de capacité, puis pour vérifier le comportement dynamique en termes de flux.

3. APPLICATION

Nous allons illustrer nos propos avec un modèle d'atelier de production de type job shop et nous allons montrer la complémentarité entre les deux approches analytique et simulateur afin de déterminer les paramètres de pilotage du système étudié.

Les critères de performance habituellement utilisés dans les problèmes d'ordonnancement sont l'utilisation des postes de travail, le temps de cycle, le débit de sortie et le niveau de stock, qui sont à maximiser ou à minimiser suivant les cas. Une alternative à ces critères est le *makespan*, qui est souvent étudié par la communauté scientifique dans les problèmes d'ordonnancement de job shop. C'est d'ailleurs l'un des critères que nous avons retenu pour notre modèle et que nous allons minimiser. Rappelons qu'il s'agit de la durée entre le début de la fabrication de la première pièce jusqu'à la fin de la fabrication de la dernière pièce.

3.1. Description du job shop

Cette étude de cas correspond à un atelier de fabrication intermittente. Dans ce type de production, les ordres de fabrication sont traités par lots et chaque lot suit une gamme spécifique.

L'atelier est composé de différents postes de travail et chaque poste comprend un stock aval géré en FIFO à chaque entrée de machine.

- Le poste de travail M1 est un poste d'ébauche des pièces.
- Le poste M2 réalise la finition.
- Le poste M3 est capable d'enchaîner les opérations d'ébauche et de finition.

Il y a deux types de pièces possibles, P1 et P2 (Figure 1). Pour chaque pièce, il faut réaliser les deux opérations d'ébauche et de finition.

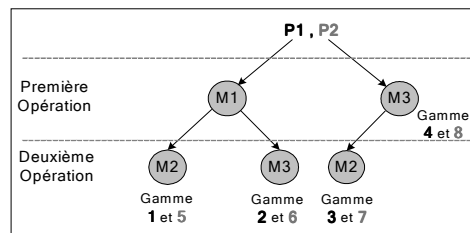


Figure 1 : Différentes gammes pour les deux pièces

Pour chaque pièce, quatre gammes sont possibles, soit au total 8 gammes avec des temps de cycle différents suivant les machines et le type de pièce (Tableau 1). Les gammes 4 et 8 réalisent les deux opérations d'ébauche et de finition sur le même poste de travail M3.

Pièce	Gamme	Opération 1		Opération 2	
		Poste	Tcyc	Poste	Tcyc
P1	G1	M1	3	M2	4
	G2	M1	3	M3	4
	G3	M3	4	M2	4
	G4	M3	8		
P2	G5	M1	5	M2	3
	G6	M1	5	M3	6
	G7	M3	3	M2	3
	G8	M3	9		

Tableau 1. Les temps de cycle des gammes (en 1/10 h)

Entre deux lots consécutifs d'une même pièce sur une même machine, il est nécessaire de réaliser un *réglage*. De la même façon, il y a un temps de *setup* pour passer d'un type de pièce à un autre (Tableau 2).

	M1	M2	M3
MTBF	400	210	196
MTTR	100	40	4
SETUP (P1-P2 ou P2-P1)	15	20	25
REGLAGE (P1-P1 ou P2-P2)	5	5	10

Tableau 2. Les lois de panne et de réglage (en 1/10 h)

Les temps moyens de bon fonctionnement (MTBF) sur chaque machine et la durée moyenne de réparation (MTTR) sont aussi indiqués dans le tableau 2.

Cet atelier travaille 8 heures par jour, 250 jours par an, soit 2000 heures/an. Les lancements se font par lots de 100 pièces. La demande annuelle prévisionnelle est respectivement de 2500 pièces P1 et 4000 pièces P2.

L'objectif principal est de satisfaire cette demande. Deux questions se posent : quel mix de gammes permet d'y arriver et comment dimensionner les aires de stockage ?

3.2. Résolution par couplage des deux approches

La démarche que nous avons adoptée se déroule en trois étapes principales (Figure 2).

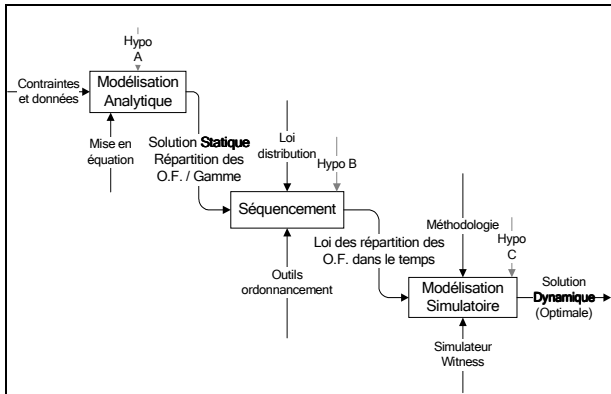


Figure 2 : Présentation de la démarche combinée

La première étape nous permet de modéliser le problème sous la forme d'équations mathématiques. La résolution analytique de ce problème mathématique donne une solution statique correspondant aux gammes à utiliser pour satisfaire la demande en pièce dans le délai imparti. Par exemple pour réaliser 25 lots de P1, le résultat de la résolution analytique donne une répartition en 6 gammes 1 et 19 gammes 2.

Avant de lancer la simulation, il faut générer au préalable une séquence d'ordres de lancement pour les commandes en respectant les résultats fournis par la résolution analytique. Cette séquence est obtenue en répartissant dans le temps les gammes de fabrication grâce à une loi de distribution aléatoire. Par exemple, pour une période de 150 jours ouvrables et la réalisation de 6 gammes P1, une répartition uniforme donnera un lancement des gammes P1 tous les 20 à 30 jours. Cette étape nécessite de compléter les hypothèses de départ, notamment pour déterminer la loi de distribution retenue ou encore choisir l'horizon de simulation (fini ou infini).

La dernière étape consiste à vérifier en dynamique les règles de pilotage de cet atelier. La simulation de flux permet cette vérification en tenant compte de l'évolution du temps. Si les résultats fournis par la simulation valident les résultats analytiques, il faudra réaliser d'autres simulations en mesurant l'influence de données stochastiques et non plus déterministes.

A l'issue de cette étape, nous pourrons appréhender les performances de notre modèle, ce qui nous permettra de définir les règles de pilotage à suivre. Si les résultats de la simulation ne satisfont pas les objectifs, alors il faut tester un ordonnancement différent.

Le processus suivant (Figure 3) illustre la démarche détaillée en trois étapes que nous avons adoptée pour cette étude de cas.

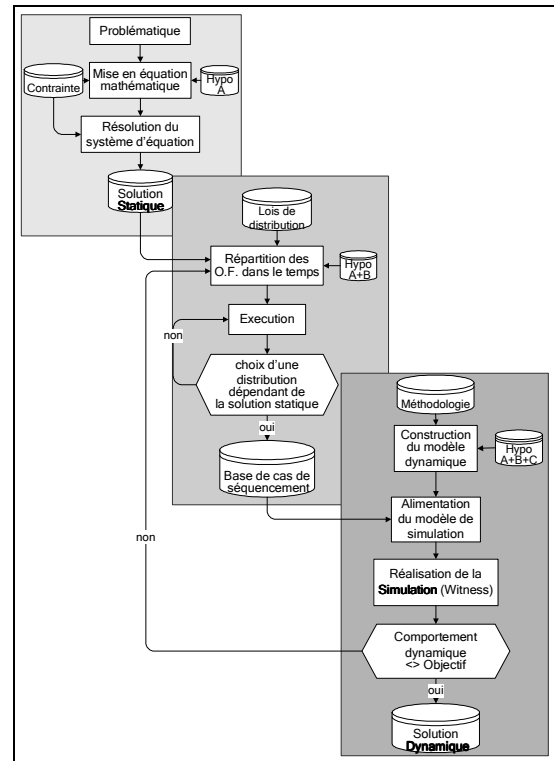


Figure 3 : Processus détaillé de complémentarité par les résolutions statiques, séquencement et dynamique

3.2.1 Modélisation analytique

Dans cette étape, l'objectif et la problématique doivent être décrits sous la forme d'équations mathématiques.

a. Le choix des hypothèses et des objectifs

Au départ, on prend l'hypothèse que nous avons 50 % de setups pour passer d'un type de pièce à un autre sur une même machine et 50 % de réglages entre deux lots consécutifs d'une même pièce.

Parmi les solutions qui respectent les contraintes, nous nous intéressons à une optimisation multi-objectif :

- Premièrement, à la minimisation du niveau de stock maximal S_{max} en aval de chaque machine défini pour éviter des ruptures d'approvisionnement dues aux aléas, tel que $S_{max} = \max S(i)$ avec $i = 1 \text{ à } 3$. Le niveau de stock est dépendant du temps total de réparation de chaque poste. Par la suite, nous détaillerons le calcul du niveau de stock dans ce cas d'étude.
- Deuxièmement, à la maximisation du temps total disponible D pour chaque machine, afin qu'il soit équitablement réparti sur les trois postes. C'est-à-dire $(\max \sum_{i=1}^3 D_{(i)})$ en tenant compte d'une répartition homogène du temps disponible ($D_2 < D_1$ et $D_3 < D_2$).

Soit $i \in \{1, \dots, 3\}$ l'indice de la machine. La fonction multi-objectif que nous cherchons donc à minimiser est :

$$\min = \frac{S_{\max}}{\sum_{i=1}^3 D_{(i)}} \quad (1)$$

La résolution doit déterminer les gammes permettant de respecter les délais parmi les huit gammes possibles et de satisfaire la demande en produits P1 et P2. Par la suite, nous décrivons les contraintes définies de l'atelier sous la forme d'équations mathématiques.

b. Description des contraintes

Dans un premier temps, nous détaillerons les différentes tâches possibles dans un temps d'ouverture d'atelier T . Pour chaque poste de travail, le temps total d'ouverture est de 250×8 h soit 20000 1/10 h. Ce temps comprend du temps de charge CP , du temps disponible D , du temps de réglage REG , du temps de setups SET et du temps de réparation REP . Soit i l'indice de la machine considérée, nous obtenons donc l'équation suivante :

$$CP_{(i)} + D_{(i)} + REG_{(i)} + SET_{(i)} = T - REP_{(i)}, \forall i \in \{1, \dots, 3\} \quad (2)$$

Il faut maintenant préciser le calcul de tous les termes de cette expression.

– Calcul du temps de charge $CP_{(i)}$

C'est le temps nécessaire pour produire l'ensemble des pièces sur chaque poste de travail. Pour le poste d'ébauche par exemple, le tableau 1 nous indique qu'il est utilisé par les gammes G1 et G2 des pièces P1 et par les gammes G5 et G6 des pièces P2. De même, les temps de cycle sont de 3/10h pour les pièces P1 et de 5/10h pour les pièces P2. Enfin, on sait que les lancements se font par lots de 100 pièces.

Soient j l'indice de la gamme, $G_{(ji)}$ le nombre de gamme j réalisées sur la machine $M(i)$ et $Tc_{(ji)}$ le temps de cycle pour réaliser la gamme $G_{(ji)}$ sur la machine $M(i)$. Ainsi, la charge totale se modélise par l'équation:

$$CP_{(i)} = 100 \times \sum_{j=1}^8 (Tc_{(ji)} \times G_{(ji)}), \forall i \in \{1, \dots, 3\} \quad (3)$$

– Calcul du temps de setup $SET_{(i)}$

C'est le temps consacré au réglage de la machine quand on change de référence de pièce. Rappelons que nous avons choisi d'équilibrer le nombre de setups et de réglages, ce qui explique le coefficient 0.5.

Soient $Ts(i)$ le temps de setup pour la machine $M(i)$, nous obtenons :

$$SET_{(i)} = 0,5 \times Ts_{(i)} \times \sum_{j=1}^8 G_{(ji)}, \forall i \in \{1, \dots, 3\} \quad (4)$$

– Calcul du temps de réglage $REG_{(i)}$

Soit $Tr(i)$ le temps de réglage pour la machine $M(i)$, nous obtenons :

$$REG_{(i)} = 0,5 \times Tr_{(i)} \times \sum_{j=1}^8 G_{(ji)}, \forall i \in \{1, \dots, 3\} \quad (5)$$

– Calcul du temps de réparation $REP_{(i)}$

Le temps de réparation est défini à partir des informations sur le MTBF, le MTTR et la charge totale de chaque machine. Le temps total de réparation pour chaque machine correspond au nombre de fois qu'une machine tombe en panne multiplié par la durée moyenne de réparation, soit :

$$REP_{(i)} = \frac{CP_{(i)}}{MTBF_{(i)}} \times MTTR_{(i)}, \forall i \in \{1, \dots, 3\} \quad (6)$$

– Calcul du niveau de stock aval $S_{(i)}$

Pour chaque machine, le stock aval doit permettre d'être correctement approvisionnée pendant le temps de réparation de la machine précédente. On obtient la relation suivante :

$$S_{(i)} = \frac{MTTR_{(i)}}{T} \times \sum_{j=1}^8 G_{(ji)}, \forall i \in \{1, \dots, 3\} \quad (7)$$

– Contraintes sur le nombre de pièces à fabriquer

Nous savons que les quatre premières gammes correspondent au produit P1, alors que les quatre dernières correspondent à P2. Nous pouvons donc écrire que :

$$\sum_{i=1}^3 \sum_{j=1}^4 G_{(ji)} = 25, \quad \sum_{i=1}^3 \sum_{j=5}^8 G_{(ji)} = 40 \quad (8)$$

c. Résolution analytique

La résolution de notre modèle analytique a été réalisée à l'aide d'un solveur *Lingo10.0* sous forme d'équations non linéaires.

Ce solveur est un programme de résolution, qui combine plusieurs algorithmes. Suivant le type d'équation (linéaire ou non-linéaire), il choisit l'algorithme le plus adapté. Notre problème est de type 'à nombres entiers à équations non linéaires', c'est-à-dire qu'il y a au moins une expression dans le modèle qui n'est pas linéaire et qu'une partie des variables sont restreintes sur l'espace des entiers N . Dans notre cas, c'est un algorithme de type 'branch and bound' qui a été utilisé.

La figure 4 présente les équations saisies dans le solveur. Les six dernières lignes de programmation linéaire (Figure 4.*) sont des contraintes supplémentaires que nous avons définies. La première ligne permet de définir la capacité maximale du stock S_{\max} , les deux suivantes permettent d'obtenir une répartition homogène du temps disponible sur les trois machines, les deux suivantes précisent le nombre de pièces P1 et P2 qu'il va falloir fabriquer dans les délais prévus et enfin la dernière indique que le nombre de gammes doit être un nombre entier.

```

! Hypothèses :
! 50% Setup 50% Réglage;
! CP i : temps coeaux sur la machines i ;
! D i : le temps restant sur la machine i ;
! G j : nombre de gamme j ;
! Objectifs :
! Maximiser le temps restant et Minimiser les stocks sécurités;
min = Smax/ (D1 + D2 + D3);
! Contraintes;;
REP1 = CP1/400*100;
REP2 = CP2/210*40;
REP3 = CP3/196*4;
CP1 = 300*G1 + 300*G2 + 500*G5 + 500*G6;
CP2 = 400*G1 + 400*G3 + 300*G5 + 300*G7;
CP3 = 400*G2 + 400*G3 + 800*G4 + 600*G6 + 300*G7+ 900*G8;
SET1 = 0.5*15*(G1 + G2 + G5 + G6);
SET2 = 0.5*20*(G1 + G3 + G5 + G7);
SET3 = 0.5*25*(G2 + G3 + G4 + G6 + G7 + G8);
REG1 = 0.5*5*(G1 + G2 + G5 + G6);
REG2 = 0.5*5*(G1 + G3 + G5 + G7);
REG3 = 0.5*10*(G2 + G3 + G4 + G6 + G7 + G8);
S1 = (100/CP1) * (G1 + G2 + G5 + G6);
S2 = (40/CP2) * (G1 + G3 + G5 + G7);
S3 = (4/CP3) * (G2 + G3 + G4 + G6 + G7 + G8);
CP1 + D1 + SET1 + REG1 = 20000 - REP1;
CP2 + D2 + SET2 + REG2 = 20000 - REP2;
CP3 + D3 + SET3 + REG3 = 20000 - REP3;
! Définir la capacité maximale du stock de sécurité Smax;
Smax>S1; Smax>S2; Smax>S3; (*)
! Répartition homogène du temps disponible sur les trois machines;
D2>D1;
D3>D2;
! Nb de pièces P1 et P2 à fabriquer dans les délais prévus;
G1+G2+G3+G4=25;
G5+G6+G7+G8=40;
! Rgin(G1);Rgin(G2);Rgin(G3);Rgin(G4);Rgin(G5);Rgin(G6);Rgin(G7);Rgin(G8)
    
```

Figure 4 : Exemple de programmation du problème sous forme d'équations mathématiques

d. Synthèse et critique des résultats analytiques

Les résultats analytiques (Figure 5) nous indiquent la meilleure combinaison possible pour minimiser notre fonction objectif. D'une part, on constate que les valeurs du temps disponible {D_i} pour chaque machine (Figure 5.i) sont approximativement équilibrées. On remarque que le temps de chargement {CP_i} est également équilibré entre les différents postes. D'autre part, cette résolution nous permet de déterminer la répartition des O.F. par gamme {G_j} pour atteindre les objectifs (Figure 5.ii). Nous éviterons ainsi de tester tous les scénarios possibles dans la simulation. Par ailleurs, on constate que les stocks en aval des postes de travail peuvent être réduits au minimum (Figure 5.iii).

Variable	Value	Reduced Cost
SMAX	0.2714286	0.000000
(i) D1	2120.000	0.000000
D2	2282.143	0.000000
D3	3174.592	0.000000
REP1	3500.000	0.000000
CP1	14000.00	0.000000
REP2	2742.857	0.000000
CP2	14400.00	0.000000
REP3	320.4082	0.000000
CP3	15700.00	0.000000
(ii) G1	6.000000	0.000000
G2	19.00000	-0.2980019E-06
G5	13.00000	0.1135547E-05
G6	0.000000	0.2365348E-05
G3	0.000000	0.000000
G7	27.00000	0.000000
G4	0.000000	-0.3635620E-06
G8	0.000000	0.1147058E-05
SET1	285.0000	0.000000
SET2	460.0000	0.000000
SET3	575.0000	0.000000
REG1	95.00000	0.000000
REG2	115.0000	0.000000
REG3	230.0000	0.000000
(iii) S1	0.2714286	0.000000
S2	0.1277778	0.000000
S3	0.1171975E-01	0.000000

Figure 5 : Résultats de la résolution analytique

En conclusion de cette résolution analytique, on peut insister sur le fait que :

- Les phénomènes aléatoires ne sont pas pris en compte, ce qui explique la capacité minimale des stocks en aval des poste de travail.
- Les événements et le comportement dynamique dans le temps ne sont pas pris en compte. Il n'est donc pas possible de mesurer les interactions entre les différents postes de travail.

3.2.2 Séquencement

Grâce à la simulation, il est possible de vérifier la dynamique de l'atelier mais nous ne pouvons pas directement utiliser les résultats de la résolution analytique pour alimenter la simulation. En effet, il manque des données liées à la répartition des gammes sur l'horizon de fabrication. Une étape intermédiaire de séquencement est nécessaire. Pour notre travail nous avons choisi d'utiliser un tableur pour générer aléatoirement ce séquencement. On répartit les gammes de fabrication obtenues dans le temps comme le montre la figure 6.

Nb Lot P1		25	Nb Lot P2		40
Taille d'un Lot P1		100	Taille d'un Lot P2		100
P1		Nb.	P2		Nb.
Gamme 1		6	Gamme 5		13
Gamme 2		19	Gamme 6		
Gamme 3			Gamme 7		27
Gamme 4			Gamme 8		
Total P1		25	Total P2		40
% P1		38%	% P2		62%
Délais Prévu		250			
Temps inter-arrivée Loi Normale					
Jours	P1		P2		
	Lancement	Gamme P1	Lancement	Gamme P2	
1	P1	2	P2	7	
2					
3					
4					
5					
6					
7			P2	7	
8					
9					
10	P1	2			
11					
12					
13					
14			P2	5	

Figure 6 : Un tableur pour générer le séquencement

Pour notre cas d'étude, nous avons choisi de créer le séquencement des lancements des ordres de fabrication :

- dans le cas déterministe, avec une répartition constante des intervalles de temps : les gammes sont toujours espacées du même nombre de jours.
- ensuite, dans le cas stochastique, avec une répartition aléatoire des intervalles de temps : les gammes sont espacées d'un nombre de jours qui suit une loi de distribution aléatoire.

Pour notre travail nous avons choisi une distribution normale comme l'illustre la figure 7.

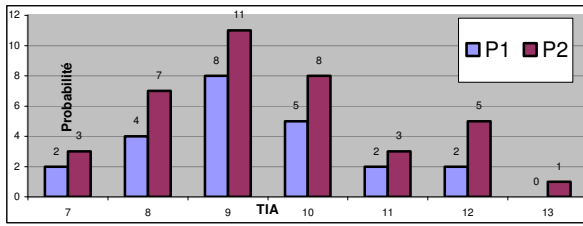


Figure 7: Intervalle de temps entre 2 gammes par la loi normale (TIA : temps inter arrivée)

Quelques hypothèses préalables ont été nécessaires. Tout d’abord, nous imposons que le premier lancement soit effectué le premier jour de la simulation. De même, nous avons choisi de ne pas lancer d’ordre de fabrication durant les dix derniers jours de simulation afin que toutes les pièces soient fabriquées dans le délai. Ainsi, la répartition des ordres de fabrication ne se fait en réalité que sur 240 jours, la simulation durant au total 250 jours.

Indépendamment pour P1 et P2, on génère une première séquence aléatoire pour définir le jour de lancement en utilisant une loi normale. Il suffit de fournir la moyenne et l’écart type souhaités qui correspond à l’intervalle de temps moyen entre deux pièces successives.

Une fois les jours de lancement choisis, on sélectionne la gamme qui sera utilisée avec une loi de probabilité basée sur les résultats analytiques. En raison du caractère aléatoire de cette répartition, il est nécessaire de lancer plusieurs exécutions successives pour obtenir une répartition de gammes identique au résultat analytique. Cette séquence permet d’alimenter correctement le modèle de simulation pour évaluer le comportement dynamique du job shop intermittent étudié.

3.2.3. Modélisation simulateur

Nous pouvons dire que le modèle de simulation de flux se compose de trois modules. Le premier module s’occupe de la gestion du temps : c’est lui qui extrait le séquencement défini pour simuler l’ordre de lancement des gammes. Le second module est un générateur de pièces. Chaque fois qu’un lancement d’ordre de fabrication est réalisé, il fournit les pièces utiles à sa réalisation aux machines concernées. Les règles de pilotage sont définies au niveau des articles pour leur indiquer quel chemin ils doivent suivre. Le troisième module concerne l’atelier physique. Il comporte les postes de travail et les stocks.

Pour vérifier comportement dynamique de l’atelier nous réaliserons successivement 3 scénario de simulation :

- a. avec des données déterministes sur un horizon fini,
- b. avec des données déterministes à horizon infini,
- c. avec des données stochastiques à horizon infini.

Ces scénario nous permettrons de vérifier la complémentarité des résultats statiques obtenus par la modélisation analytique avec les résultats dynamiques de la modélisation simulateur.

a. Simulation déterministe à horizon fini

Nous utilisons le terme *déterministe* car le modèle de simulation est alimenté par des données déterministes, et l’on ne prend pas en compte les phénomènes aléatoires dans le modèle lui-même.

Le résultat obtenu analytiquement indique le choix des gammes de production. Comme le montre le tableau 3, pour chaque produit, seules deux des quatre gammes possibles sont utilisées. Par exemple, la fabrication des 25 O.F. du produit P1 est obtenu par 6 O.F. suivant la gamme 1 et 19 O.F. suivant la gamme 2. Sachant que la taille de lot par O.F. est de 100 pièces et que les temps de cycle sont connus, on peut déterminer par calcul la charge de chaque machine.

	Nb.Lot	Gamme	Nb.Gamme	Nb.Pièces	Nb.Op.M1	Nb.Op.M2	Nb.Op.M3
P1	25	1	6	600	6 [3]	6 [4]	
		2	19	1900	19 [3]		19 [4]
		3	0			0 [4]	0 [4]
		4	0				0 [8]
P2	40	5	13	1300	13 [5]	13 [3]	
		6	0		0 [5]		0 [6]
		7	27	2700		27 [3]	27 [3]
		8	0				0 [9]
Nb. Opération					38	46	46
Tps de charge dh/100					140	144	157

Tableau 3 : La charge et le choix des gammes

La simulation doit vérifier que l’atelier peut produire les gammes définies dans les délais prévus. Tout d’abord, nous allons réaliser une *Simulation Déterministe à Horizon FINI qui est censée donner des résultats proches de la modélisation analytique*. Le terme d’horizon fini signifie que le modèle de simulation se terminera avec un modèle « vide » : plus aucune pièce ne sera présente dans l’atelier et les machines seront toutes disponibles.

La figure 8 illustre l’évolution des encours dans l’atelier sur l’horizon de simulation. Cet enregistrement fait apparaître trois parties:

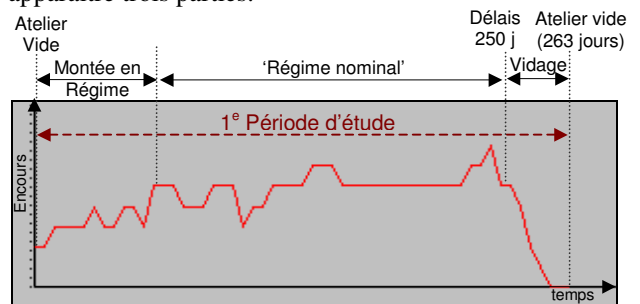


Figure 8. Niveaux des encours dans une simulation déterministe à horizon fini

- une période de *montée en régime* pendant laquelle le niveau des encours augmente,
- une période de *régime nominal* pendant laquelle le flux moyen entrant est identique au flux moyen sortant ce qui stabilise le niveau des encours,
- une période de fin production, pendant laquelle l’atelier se vide. La simulation continue jusqu’à l’obtention d’un modèle *vide à l’état final*.

On note une légère différence de 5% au niveau du *makespan* entre le modèle analytique (20000 1/10

heures) et le modèle de simulation (20984 1/10 heures). Ceci entraîne un retard de la livraison des commandes de 13 jours par rapport au délai prévu. Cette différence s'explique par le fait qu'il y a un phénomène d'enchaînement des opérations dans la simulation qui est inexistant dans le modèle analytique où les machines fonctionnent indépendamment les unes des autres. Une notion de précédence d'ordre de réalisation a été introduite, d'où l'apparition de ce décalage. Ce phénomène peut aussi s'expliquer par le fait que la ligne est vide au début de la simulation. Il y a donc de « l'inertie » avant qu'elle n'atteigne son régime permanent. Par contre, le taux d'occupation des machines est cohérent d'un modèle à l'autre, de même que le temps de charge de chaque machine (voir le tableau 4).

	Nb. Op.	Analytique (= 20000)		Simulation (=20565)	
		% Occ.	Tps Charge	% Occ.	Tps Charge
M1	38	70,00%	14000	66,72%	14000,52
M2	46	72,00%	14400	68,62%	14399,22
M3	46	78,50%	15700	74,82%	15700,22

Tableau 4 : Comparaison du taux d'occupation et du temps de charge de chaque machine

Comme les statistiques fournies par la simulation ne font pas la distinction entre réglage et setup, nous comparons les résultats issus de la simulation à la somme des temps de réglage et setup calculés analytiquement à partir des données sur le MTBF et le MTTR de chaque machine. Comme précédemment, nous constatons que les résultats sont très similaires. On observe le même phénomène quand on compare les résultats concernant le temps de réparation. Par contre, le nombre de setups et de réglages ne correspond pas du tout aux hypothèses formulées dans le modèle analytique (Tableau 5), à savoir 50 % de réglages et 50 % de setups.

	Setup		Réglage		Tps reg+setup / machine
	Nb.Setup	Tps.Setup	Nb.Réglage	Tps.Réglage	
M1	24	[15]	13	[5]	425
M2	11	[20]	34	[5]	390
M3	25	[25]	20	[10]	825
Tps totale	1205		435		

Tableau 5 : Le temps de réglage et de setup

b. Simulation déterministe à horizon infini

Dans ce cas, nous réalisons la simulation sur une période suffisamment longue de régime nominal. Ce qui signifie que nous excluons de l'étude les périodes de montée en régime et de fin de production.

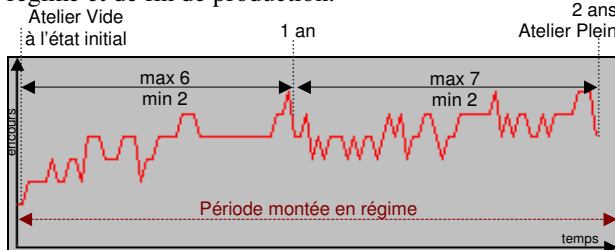


Figure 9 : Niveaux des encours dans une période de montée en régime de 2 ans (simulation à horizon infini)

Dans notre travail, nous avons choisi une période de régime nominal d'un an faisant suite à une période de montée en régime de 2 ans (=40000 dh). Les statistiques de simulation sont donc fournies sur une année de régime nominal. Le niveau des encours se stabilisent entre deux valeurs (Figure 9).

Après cette période de montée en régime, on constate que l'atelier a la capacité de réaliser les commandes de 25 produits P1 et de 40 produits P2 sur une durée de 245 jours (Figure 10). Cette durée est donc inférieure au délai prévu ce qui permet de dire que l'atelier est dans ce cas capable d'atteindre l'objectif. Ces résultats confirment ceux fournis par la modélisation analytique.

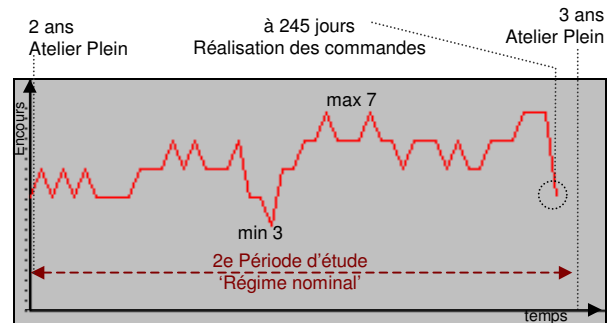


Figure 10. Niveaux des encours dans une simulation déterministe à horizon infini

Par contre, les résultats de cette simulation permettent d'établir que le niveau des stocks en aval de chaque poste de travail est différent de celui obtenu par la modélisation analytique.

Poste de travail	M1	M2	M3
Capacité maximum de stock	3	4	4

c. Simulation stochastique à horizon infini

Dans ce dernier scénario, on utilise des données *stochastiques* qui sont plus proches de la réalité. Par exemple, nous avons utilisé une *loi normale* pour la distribution des temps inter arrivées (TIA) des pièces et des temps de réglage et setup. De la même façon, nous avons utilisé une *loi erlang* pour la distribution de MTBF et une *loi gamma* pour la distribution de MTTR.

Après une période de montée en régime de 3 ans (=60000dh), on constate que l'atelier réalise les commandes de 25 produits P1 et de 40 produits P2 sur une durée de 265 jours au lieu de 250 jours. La simulation stochastique à horizon infini fait donc apparaître un retard de 15 jours par rapport à l'objectif.

Par ailleurs, comme pour les scénarios précédents, la simulation met en évidence que le niveau de stock en aval de chaque poste de travail doit être suffisant pour amortir les variations des flux. Le tableau suivant donne les capacités de stock minimum à prévoir dans cet atelier.

4. CONCLUSION

Dans ce travail, nous avons proposé une démarche complémentaire de deux approches analytique et simulateur afin de dimensionner un atelier de production manufacturier. Nous nous sommes appuyés sur un cas d'étude d'un atelier de type job shop pour illustrer cette démarche.

Les résultats des différents scénarios de simulation confirment en partie les résultats obtenus par la résolution analytique mais fournissent des résultats plus précis notamment sur la capacité minimale des stocks et le délai de réalisation.

La résolution analytique se justifie à plusieurs titres, en complément de la simulation :

- d'abord, elle permet de trouver très rapidement une solution optimale, globale ou locale ;
- ensuite, elle fournit des informations très utiles pour guider les décideurs dans leurs choix, notamment pour des problèmes de dimensionnement et d'ordonnement ;
- enfin, elle facilite la construction d'un modèle de simulation de flux.

Par ailleurs, la simulation n'en reste pas moins recommandée pour appréhender le comportement dynamique du système modélisé et valider les règles de pilotage. La simulation apporte par exemple une information plus précise sur la capacité des stocks entre les machines par rapport à la résolution analytique. Pour la conception d'un système réel, cette précision peut s'avérer déterminante à la fois sur le plan technique et sur le plan financier. Une erreur de dimensionnement de stock peut en effet avoir des conséquences fâcheuses sur les performances d'un atelier.

Enfin, la littérature met clairement en évidence les limites de l'approche analytique seule pour traiter des cas complexes. Couplée à une approche simulateur, l'approche analytique reste donc attractive pour modéliser ces cas complexes.

REFERENCES

Allaoui, H., and A. Artiba. 2004. Integrating simulation and optimization to schedule a hybrid flow shop with maintenance constraints. *Computers & Industrial Engineering*, Vol. 47, pp. 431-450.

Almeida, M. R., S. Hamacher, M. A. C. Pacheco, and M. B. R. Velasco. 2001. Applying Genetic Algorithms to the Production Scheduling of a Petroleum Refinery. MIC'2001 - 4th Metaheuristics International Conference, pp. 773-777.

Baessler, F. F., and J. A. Sepúlveda. 2001. Multi-Objective Simulation Optimization for a Cancer Treatment Center. *Proceedings of the 2005 Winter Simulation Conference*, Institute of Electrical and Electronics Engineers, Piscataway.

Blackstone, J., D. Phillips et G. Hogg, 1982. A State-of-The-Art Survey of Dispatching Rules for Manufacturing Job Shop Operations. *International Journal of Production Research*, Vol. 20, pp. 26-45.

Cardin O., Castagne P., 2006, Utilisation de la simulation proactive: Une aide au pilotage des

systèmes de production. 6^e Conférence Francophone de Modélisation et Simulation-MOSIM'06, Maroc.

Chong, C.S. and M. Yoke Hean Low, 2006. A Bee Colony Optimization Algorithm to Job Shop Scheduling. SIMTech, School of Computer Engineering, Nanyang Technological University, Winter Simulation Conference '06.

Collet, P. et J.P. Rennard, 2006. Introduction to Stochastic Optimization Algorithms. *Handbook of Research on Nature-Inspired Computing for Economics and Management*, édité par J.-P. Rennard, IDEA Group Inc, 1100 pages.

Deb, K. 2001. *Multi-objective Optimization Using Evolutionary Algorithms*, John Wiley & Son.

Eskandari, H., L. Rabelo, and M. Mollaghasemi. 2005. Multiobjective Simulation Optimization Using an Enhanced Genetic Algorithm. *Proceedings of the 2005 Winter Simulation Conference*, Institute of Electrical and Electronics Engineers, Piscataway.

Feo T.A. and M. Resende, 1988. A Probabilistic heuristic for a Computationally Difficult Set Covering Problem. *Opérations Research Letters* 8, pp. 67-71.

Gere, W. S., Jr., 1966. Heuristics in jobshop scheduling. *Management Science*, Vol. 13, No. 1, pp. 167-175.

Glover F. and M. Laguna, 1997. *Tabu search*, Kluwer Academic Publishers, Boston.

Goldberg, D., 1989. *Genetic Algorithms in Search, Optimization & Machine-Learning*, Addison-Wesley.

Gupta, A. K., and A. I. Sivakumar. 2002. Simulation based Multiobjective Schedule Optimization in Semiconductor Manufacturing, *Proceedings of the 2002 Winter Simulation Conference*, Institute of Electrical and Electronics Engineers, Piscataway.

Hicks D. A., 1999. A four step methodology for using simulation and optimization technologies in strategic supply chain planning. *Proceedings of the 1999 Winter Simulation Conference*.

Holland, J., 1975. *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Harbor.

Jain, A. S. and S. Meeran, 1999. Deterministic job shop scheduling: past, present and future. *European Journal of Operational Research*, Vol.113, pp.390.

Koza, J.R., 1992. *Genetic programming: on the programming of computers by means of natural selection*. The MIT Press, Cambridge.

Minoux, M., 1983. *Programmation mathématique tomes I et II.*, Dunod.

Nowicki, E. and C. Smutnicki, 1996. A fast taboo search algorithm for the job shop problem. *Management Science*, Vol. 42, No. 6, pp. 797-813.

Prudius, Andrei A. and S. Andradóttir, 2005. Two Simulated Annealing Algorithms for Noisy Objective Functions, *Georgia Institute of Technology, Winter Simulation Conference '05*

Rajendran, C. et O. Holthaus, 1999. A comparative study of dispatching rules in dynamic flowshops and jobshops, *European Journal of Operational Research* Vol. 116, No. 1, pp. 156-170

Rechenberg, I. 1973. *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*, Frommann-Holzboog.