HAL
open science

OPTIMISATION DES SERVICES DE MAINTENANCE ET DE PRODUCTION SOUS DES CONTRAINTES DE RESSOURCES NON-RENOUVELABLES-UNE APPROCHE BASÉE SUR LES ALGORITHMES GÉNÉTIQUES

Fayçal Belkaid, Mohammed Dahane, Zaki Sari, Abdelhakim Khatab

To cite this version:

Fayçal Belkaid, Mohammed Dahane, Zaki Sari, Abdelhakim Khatab. OPTIMISATION DES SERVICES DE MAINTENANCE ET DE PRODUCTION SOUS DES CONTRAINTES DE RESSOURCES NON-RENOUVELABLES-UNE APPROCHE BASÉE SUR LES ALGORITHMES GÉNÉTIQUES . MOSIM 2014, 10ème Conférence Francophone de Modélisation, Optimisation et Simulation, Nov 2014, Nancy, France. hal-01166640

HAL Id: hal-01166640

https://hal.science/hal-01166640v1

Submitted on 23 Jun 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

OPTIMISATION DES SERVICES DE MAINTENANCE ET DE PRODUCTION SOUS DES CONTRAINTES DE RESSOURCES NON-RENOUVELABLES : UNE APPROCHE BASÉE SUR LES ALGORITHMES GENETIQUES

F. BELKAID[1,2], M. DAHANE[3]

MELT/LGIPM
[1]Université de Tlemcen/[2]Université de Lorraine
[3]Ecole Nationale d'Ingénieurs de Metz
faycal.belkaid@mail.univ-tlemcen.dz, dahane@enim.fr

Z. SARI[1], A. KHATAB[3]

MELT/LGIPM
[1]Université de Tlemcen
[3]Ecole Nationale d'Ingénieurs de Metz
z_sari@mail.univ-tlemcen.dz, khatab@enim.fr

RESUME : *Cet article considère un problème de planification de la maintenance et d'ordonnancement sur un système de machine parallèles soumis à des contraintes de ressources renouvelables et consommables. L'objectif est d'intégrer les services de maintenance préventive avec les actions de production sous des contraintes de ressources non-renouvelables. Une métaheuristique est présentée pour résoudre le problème d'ordonnancement des tâches à effectuer et celui des actions de maintenance préventive à réaliser sur les différentes machines. L'approche proposée vise à minimiser le makespan à travers la minimisation des indisponibilités des machines. Le modèle présenté dans cette étude consiste à choisir simultanément, la machine la plus appropriée pour chaque opération, à déterminer la séquence des tâches assignées à la même machine et à définir les instants de réalisation de la maintenance préventive.*

MOTS-CLES : *Intégration de la production et la maintenance, ressources consommables, ordonnancement de la production, maintenance préventive, fiabilité, algorithme génétique.*

1 INTRODUCTION

L'objectif majeur des industriels est d'exploiter efficacement leurs systèmes de productions afin d'améliorer leur réactivité et leur productivité en utilisant toutes les ressources disponibles d'une façon optimale dans des délais minimums. Pour cela, il est nécessaire, d'une part, d'élaborer des stratégies de maintenance afin de garantir le niveau de fiabilité requis, d'améliorer l'exploitation et d'assurer la disponibilité des machines et d'autre part, de développer des méthodologies d'ordonnancement de la production pour exploiter au mieux les disponibilités des machines, garantir les niveaux exigés de la qualité et respecter les délais demandés. Dans ce contexte, les stratégies de maintenance et de production, ont un impact direct sur les performances de l'entreprise.

L'ordonnancement de la production consiste à déterminer la séquence de passage d'un certain nombre d'opérations à exécuter sur différentes machines. Il s'agit de prévoir le travail à réaliser, de façon à coordonner l'utilisation des ressources renouvelables et non renouvelables afin de satisfaire les différentes contraintes et d'optimiser un ou plusieurs critères de performances. L'ordonnancement sur machines parallèles est un problème qui reste assez fréquent dans plusieurs industries. Il consiste à affecter les tâches aux machines et à déterminer la séquence des travaux assignés à la même machine. Il représente un domaine de recherche important. Cette importance est encore amplifiée par la variété de configurations complexes qui peuvent être réduites à ce type de problème.

Dans la littérature relative aux problèmes d'ordonnancement, une partie importante des travaux suppose la disponibilité totale des ressources (Xu et al., 2008), (Berrichi et al., 2009), mais ce n'est pas toujours le cas, car les ressources peuvent être consommées et peuvent être indisponibles pour diverses raisons (Belkaid et al., 2013a), (ex, une quantité de matière première est consommée après l'exécution de chaque tâche) et par conséquent, le manque de ressources perturbe et affecte l'efficacité de l'ordonnancement.

Pour ce qui concerne la maintenance, elle vise à assurer les niveaux nécessaires de fiabilité, disponibilité, qualité,… Ainsi, une stratégie de maintenance a comme but d'assurer la pérennité des systèmes de production, de diminuer les pannes imprévues et de minimiser les couts réparation, de révision et de vérification des équipements matériels. Cependant, plusieurs études menées précédemment dans le domaine de l'ordonnancement supposent souvent que les systèmes de production des entreprises manufacturières sont toujours disponibles et ignore l'aspect maintenance ce qui augmente la probabilité de pannes des machines. Dans ce contexte, afin d'assurer une disponibilité optimale des équipements (et du système de production d'une façon générale), il est nécessaire de déterminer la stratégie de production en tenant compte la planification des dates d'intervention des actions de maintenance. En effet, et malgré la relation d'interdépendance entre stratégie de production et de maintenance, les deux activités sont généralement planifiées et exécutés séparément ce qui conduit à un conflit dans la prise de décisions. Par conséquent, il est primor-

dial de coordonner les services de la production et ceux de la maintenance (Nahmias, 2004), afin de satisfaire les différentes demandes des clients, minimiser les couts de maintenance et maximiser la productivité de l'entreprise.

Dans ce sens, nous proposons un modèle intégré qui prend en considération le service de maintenance et celui de production sous les contraintes des ressources non-renouvelables ; afin d'augmenter la disponibilité du système de production et de maximiser la productivité en exploitant efficacement les ressources disponibles (renouvelables et consommables).

La prochaine section sera dédiée à une revue de la littérature sur le problème étudié, ensuite, la section 3 décrira la modélisation de l'approche intégrée d'ordonnancement de la production et de la planification de la maintenance préventive. La section 4 sera consacrée à la présentation de la méthode de résolution. La section 5 présentera les différentes simulations effectuées et les résultats obtenus. Finalement, la section 6 sera dédiée à une conclusion et recommandation pour quelques travaux futurs.

2 REVUE DE LA LITTÉRATURE

Ces dernières décennies, quelques auteurs se sont intéressés à étudier des problèmes d'ordonnancement sur machines parallèles avec des ressources consommables et à l'étude de leurs complexité : Lenstra (Lenstra et al., 1977) et Sethi (Sethi, 1977) montrent que l'ordonnancement sur machines parallèles est NP-difficiles, et cela, même dans le cas de deux-machines. En outre, Carlier et Rinnooy Kan (Carlier et Rinnooy Kan, 1982) montrent que pour les problèmes à une seule machine avec ressources consommables, les problèmes deviennent NP-difficiles au sens fort quand les durées des jobs ne sont pas identiques. De plus, Blazewicz et al. (Blazewicz et al., 1983) confirment que la quasi-totalité des problèmes sur machines parallèles avec contraintes de ressources non-renouvelables sont NP-difficiles sauf pour quelques cas particuliers qui apparaissent lorsque le nombre de machines est égal à deux où ils peuvent être résolus en un temps polynomial.

L'implication croissante des machines parallèles avec ressources consommables dans plusieurs domaines explique l'intérêt grandissant des chercheurs pour ce problème. Cela peut être prouvé à travers l'état de l'art effectué par Edis et al. (Edis et al., 2013) sur les machines parallèles avec ressources additionnelles, dont lequel il fournit des éléments de référence de la littérature. Daniels et al. (Daniels et al. 1997) (Daniels et al., 1999) proposent une métaheuristique basée sur la recherche tabou pour minimiser le makespan sur machines parallèles avec des ressources non-renouvelable. Kai Li et al. (Kai et al., 2011) proposent un algorithme de recuit simulé afin de minimiser le makespan pour un problème d'ordonnancement sur machines parallèles identiques avec un temps de traitement contrôlable. Ils supposent que les temps de traitement de chaque tâche sont des fonctions linéaires décroissantes de la ressource consommée. Lee et al. (Lee et al., 2013) considère un problème d'ordonnancement pour assurer la logistique d'urgence de fourniture des services médicaux dans les zones affectées après une catastrophe. Ils montrent que si la quantité de ressource consommable disponible à l'instant zéro est suffisante, alors le problème est équivalent à un problème d'ordonnancement sur machines parallèles avec des dates de disponibilité et des séquences dépendant de temps de traitement; et comme objectif la minimisation du retard total pondéré. Belkaid et al. (Belkaid et al., 2013a) proposent un algorithme génétique et une heuristique qui consiste à combiner les temps de traitement et la consommation des ressources non-renouvelables pour l'ordonnancement sur machines parallèles avec des ressources consommables. Ils supposent que l'arrivée de chaque composant est représentée par une courbe sous forme d'escaliers. .Comme extension, ils développent un modèle mathématique linéaire en nombres entiers et un algorithme génétique hybride pour minimiser le makespan (Belkaid et al., 2013b).

Malgré la richesse de la littérature dans le domaine de l'ordonnancement de la production et la maintenance des systèmes de production, de nombreux travaux de recherche ne tiennent pas en compte l'impact des ressources non-renouvelables sur la disponibilité et les performances du système. Une meilleure gestion des ressources peut offrir de nombreux avantages, tels que l'amélioration de taux de production et la minimisation du nombre de travaux en cours de production.

D'un autre côté, nous notons que la plupart des études qui s'intéressent à l'aspect production ignorent généralement le service de maintenance ou considèrent que les périodes de maintenance sont déjà planifiés (Liao et al., 2007), (Dahane et al., 2010). En d'autres termes, l'ordonnancement de production est effectué en fonction d'un plan de maintenance établi préalablement.

Dans ce contexte, quelques études ont été consacrées à l'optimisation des deux fonctions relatives à la maintenance et la production, mais séparément. La majorité des études d'ordonnancement qui prennent en compte la maintenance supposent que l'intervalle de temps des actions de maintenance préventive ainsi que leur nombre sont connus à l'avance. Cependant, dans les modèles intégrés, les actions de maintenance préventive doivent être considérées comme des variables de décision, ce qui est de même pour les actions de production. Ces deux démarches doivent être ordonnancées conjointement et simultanément. Quelques travaux ont été réalisés dans ce sens :

Lee (Lee, 1996) et Schmidt (Schmidt, 2000) analysent cette approche en étudiant différentes contraintes et différents modèles d'ateliers de production. Ils s'intéressent aussi aux problèmes dans lesquels les dates d'exécution des tâches de maintenance ne sont pas fixées à l'avance. Dans cette approche, la maintenance est sou-

vent prioritaire par rapport à la production. Kaabi et al. (Kaabi et al., 2002) proposent différentes heuristiques pour la maintenance et l'ordonnancement sur une seule machine afin de minimiser le retard maximal. Les auteurs supposent que les tâches de maintenance sont effectuées dans un intervalle fixé à l'avance. Kaabi et al. (Kaabi et al., 2003) considèrent un problème de production et de maintenance dans un système de type flow shop et supposent que les dates d'interventions de la maintenance varient dans un intervalle déterminé à l'avance. Dans leur étude ils présentent un algorithme génétique pour la résolution du problème étudié. Cassady and Kutanoglu (Cassady et Kutanoglu, 2003) formulent un modèle intégré pour minimiser le retard total pondéré de production sur une seule machine. Ils proposent une méthode exacte de résolution qui s'avère inefficace pour les problèmes de taille industrielle. Ruiz et al. (Ruiz et al., 2007) introduisent un modèle intégré pour résoudre un problème de type flow shop où les périodes de maintenance sont fixés afin de conserver un niveau minimum de fiabilité pour chaque machine. Le niveau de fiabilité est considéré comme une contrainte. Leurs objectifs est de minimiser le makespan. Xu et al. (Xu et al., 2008) considèrent un problème intégré dans un environnement de machines parallèles dans le quelles activités de maintenance sont quasi-périodiques. Les auteurs considèrent le makespan comme objectif mais ils défavorisent l'aspect production dans leurs recherches.

Berrichi et al. (Berrichi et al., 2009) proposent un modèle intégrée de la maintenance préventive et d'ordonnancement de la production sur des machines parallèles. Ils utilisent un modèle de fiabilité pour prendre en compte l'aspect de la maintenance. Ils proposent deux algorithmes génétiques pour optimiser deux critères qui sont : la minimisation du temps d'exécution total pour la partie de la production et la minimisation de l'indisponibilité du système pour la partie de la maintenance. Berrichi et al. (Berrichi et al., 2010) traitent le même problème décrit précédemment, ils proposent, cette fois-ci, une métaheuristique basée sur les colonies de fourmis pour minimiser le makespan pour l'aspect production et pour optimiser l'indisponibilité du système relative à l'aspect maintenance. Moradi et al. (Moradi et al., 2011) considèrent un problème intégré de type job shop flexible avec une activité de maintenance préventive. Ils proposent un algorithme génétique pour la minimisation du makespan pour la partie de la production et la minimisation de l'indisponibilité du système pour la partie de maintenance. Berrichi et al. (Berrichi et Yalaoui, 2013) proposent une extension de la problématique étudié par Berrichi et al. (Berrichi et al., 2009) en considérant une approche de colonies de fourmis pour minimiser le retard maximal (pour le service de production) et l'indisponibilité du système (pour le service de maintenance).

Malgré le nombre importants d'études menées dans le domaine de l'ordonnancement de la production et la maintenance des systèmes de production, de nombreuses d'entre elles ne tiennent pas en compte, l'impact des ressources non-renouvelables sur la disponibilité et les performances du système. Une bonne gestion des ressources peut offrir de nombreux avantages, comme l'amélioration de taux de production et la minimisation de nombre de travaux en cours de production.

Ce papier étend les travaux menés dans la littérature et notamment celui de Berrichi et al. (Berrichi et al., 2009), en proposant une stratégie visant à intégrer la gestion des actions de maintenance préventive avec celle des actions de production tout en tenant compte des contraintes de ressources consommables. La stratégie adaptée est celle de Berrichi et al. (Berrichi et al., 2009) qui stipule qu'il est possible de retarder ou d'avancer une tâche de maintenance préventive. En plus, le modèle proposé prend en compte la fiabilité des machines pour la réalisation de la maintenance préventive. Cet article propose une approche basée sur une métaheuristique pour résoudre le problème relatif de l'intégration de l'ordonnancement sur machines parallèles avec des ressources non-renouvelables et les activités de maintenance préventive, afin de minimiser le makespan et les indisponibilités des machines.

3 MODÉLISATION DE L'APPROCHE INTÉGRÉE

Cette section décrit séparément la définition du problème d'ordonnancement de la production et celui du problème de planification de la maintenance préventive, puis le modèle intégré proposé.

En ce qui concerne la production, dans la majorité des cas, un problème d'ordonnancement peut être représenté comme un problème d'optimisation en considérant des objectifs à atteindre afin de pouvoir juger la pertinence d'un ordonnancement et mesurer les performances du système tout en satisfaisant ses contraintes du point de vue exploitation. Dans ce contexte, plusieurs chercheurs se sont intéressés à la minimisation du Makespan pour les problèmes d'ordonnancement sur machines parallèles. Ce critère qui fait l'objet de notre travail, permet d'équilibrer la charge entre les machines, d'offrir généralement un taux élevé de leurs utilisations et de maximiser la productivité du système. Le problème considéré consiste à ordonnancer un ensemble de n tâches sur m machines parallèles identiques avec la présence de ressources non-renouvelables.

Le système étudié possède les caractéristiques suivantes : L'atelier contient plusieurs machines parallèles identiques. Chaque tâche j est caractérisée par un temps de traitement p_j et elle peut être exécutée sur une machine i, lorsque tous les composants nécessaires sont disponibles. Le processus de production pourra être lancé si tôt qu'une tâche peut être exécutée. Chaque tâche consomme k composants au début de son exécution. Les composants sont livrés par les fournisseurs à des instants différents et l'arrivée de chaque composant est représen-

tée par une courbe sous forme d'escalier. La préemption n'est pas autorisée, en outre, les machines sont disponibles à l'instant initial $t = 0$, et ne peuvent traiter qu'une seule tâche à la fois. Enfin, chaque machine possède une file d'attente d'entrée avec une capacité illimitée et les tâches seront dirigées vers la file d'attente qui comporte le moins de tâches (voir la figure 1).

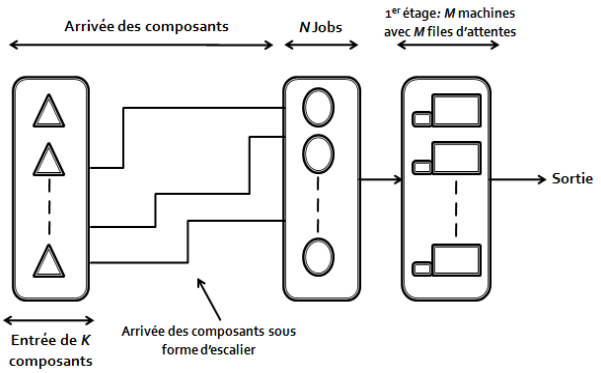

Figure 1 : Représentation du problème d'ordonnancement de la production (Belkaid et al., 2013a)

Les variables utilisées dans le modèle du problème étudié sont présentées comme suit :

- n : le nombre de tâches
- m : le nombre de machines
- c : le nombre de composants
- T_1 : l'instant de la première arrivée des composants
- T_{last} : l'instant de la dernière arrivée des composants
- j : l'indice de la tâche, où $j=1,...,n$
- i : l'indice de la machine, où $i=1,...,m$
- k : l'indice de la ressource, où $i=1,...,c$
- t : l'indice d'arrivé des composants, où $t= T_1,..., T_{last}$
- n_i : le nombre de tâches affectées à la machine i
- p : la position de la tâche dans une machine, où $p=1,...n_i$
- p_j : le temps opératoire de la tâche j
- d_{i0} : la date de début de l'ordonnancement
- d_{ip} : la date de début de traitement du job sur la machine i en position p
- p_{ip} : la somme des temps de traitement des jobs jusqu'à atteindre la machine i en position p
- f_{ip} : la date de fin de traitement du job sur la machine i en position p
- c_{jk} : la quantité de composant k consommé par le job j
- c_{ipk} : le nombre de composant k que le job qui est sur la machine i en position p consomme
- A_{tk} : le nombre total de composant k arrivé jusqu'à l'instant t
- Z : est un nombre positif très grand
- C_{max} : le makespan
- X_{jip} et Y_{ipt} : les variables binaires

$$Min\ C_{max} \tag{1}$$

Sous les contraintes suivantes :

$$\sum_{j=1}^{n} X_{jip} = 1 \quad \forall\, i = 1,2,\dots,m \quad \forall\, p = 1,2,\dots,n_i \tag{2}$$

$$\sum_{i=1}^{m}\sum_{p=1}^{n_i} X_{jip} = 1 \quad \forall\, j = 1,2,\dots,n \tag{3}$$

A ce stade, la fonction objectif consiste à minimiser le makespan pour la partie production (avant de considérer la partie maintenance). La famille des contraintes (2) et (3) assurent qu'il y a seulement une tâche affectée sur la machine i et à la position p et que chaque tâche est affectée seulement une fois sur ces machines.

$$p_{ip} = \sum_{j=1}^{n} X_{jip}\, p_j \quad \forall\, i = 1,2,\dots,m \quad \forall\, p = 1,2,\dots,n_i \tag{4}$$

$$d_{i0} = 0 \quad \forall\, i = 1,2,\dots,m \tag{5}$$

$$d_{i(p-1)} + p_{ip} \le f_{ip} \quad \forall\, i = 1,2,\dots,m \quad \forall\, p = 1,2,\dots,n_i \tag{6}$$

$$d_{ip} = f_{ip} - p_{ip} \quad \forall\, i = 1,2,\dots,m \quad \forall\, p = 1,2,\dots,n_i \tag{7}$$

La contrainte (4) calcule la durée d'opération de la tâche qui est en position p sur la machine i. La contrainte (5) assure que le début de l'ordonnancement est à l'instant 0. La contrainte (6) a pour objectif le calcul des temps de fin de traitement des jobs à chaque position p. La contrainte (7) calcule la date de début du job qui est sur la machine i *et* en position p.

$$c_{ipk} = \sum_{j=1}^{n} X_{jip}\, c_{jk} \quad \forall\, i = 1,2,\dots,m \quad \forall\, p = 1,2,\dots,n_i \quad \forall\, k = 1,2,\dots,c \tag{8}$$

$$\sum_{i=1}^{v}\sum_{p=1}^{w} c_{ipk} \le \sum_{t=T_1}^{T_{last}} A_{tk} * Y_{iwt} \quad \forall\, v = 1,2,\dots,m \quad \forall\, w = 1,2,\dots,n_i \quad \forall\, k = 1,2,\dots,c \tag{9}$$

Les contraintes relatives aux ressources consommables peuvent être résumées comme suit : La contrainte (8) permet de calculer le nombre de composants consommés par la tâche qui est en position p sur la machine i. La contrainte (9) vérifie que la quantité consommée par un job en position p est inférieure ou égale au nombre qu'on dispose et cela pour chaque ressource. De plus elle assure que la date de début de traitement du job sur la machine i en position p est supérieure à l'arrivée du composant.

$$Z * (Y_{ipt} - 1) \le d_{ip} - t \;\forall\, v = 1,2,\dots,m \;\forall\, t = t_1,\dots,T_{last} \tag{10}$$

$$C_{max} \ge \max_{i} f_{ip} \quad \forall\, i = 1,2,\dots,m \quad \forall\, p = 1,2,\dots,n_i \tag{11}$$

La contrainte (10) met le lien entre la variable Y_{ipt} et l'instant de début de traitement du job en position p. Elle a pour objectif de conserver sa linéarité du modèle mathématique. La contrainte (11) décrit le makespan, il est égal au maximum des dates de fin de traitement des derniers jobs sur les machines.

$$X_{jip} = \begin{cases} 1 \text{ si le job j est en position p sur la machine i} \\ 0\ sinon \end{cases} \tag{12}$$

$$Y_{ipt} = \begin{cases} 1\ si\ d_{ip} \ge t \\ 0\ sinon \end{cases} \tag{13}$$

La contrainte (12) indique que la variable binaire X_{jip} est égale à 1, si la tâche *j* est en position *p* sur la machine *i* et 0 sinon. La contrainte (13) est une variable binaire, elle est égale à 1 si la date de début de traitement du job sur la machine i en position est supérieur à l'instant d'arrivé des composants et 0 sinon.

Pour la partie maintenance, notre étude consiste à proposer un plan de maintenance préventive systématique (Berrichi et al., 2009). Cette phase de maintenance permet d'augmenter la disponibilité d'un système en préservant les outils de production dans de bonnes conditions de fonctionnement et de diminuer les coûts en réduisant l'occurrence des défaillances imprévues. La disponibilité est définie comme : "la probabilité qu'un système (ou un composant) accomplit sa fonction requise à un moment donné ou sur une période déterminée" (Ebeling, 1997). Par conséquent, l'augmentation de la disponibilité d'un matériel consiste à la diminution des nombres de ses arrêts et à la réduction du temps nécessaire pour la résolution des causes de ces arrêts. Mais la difficulté de cette stratégie de maintenance réside dans la détermination des dates d'intervention pour chaque machine.

Dans le reste du papier nous utilisons les paramètres suivants nécessaires à la définition de la stratégie de maintenance:

- λ_i : le taux de défaillance d'une machine *i*
- μ_i : le taux de réparation d'une machine *i*
- $A_i(t')$: la disponibilité d'une machine i à l'instant *t'*
- $A_s(t')$: la disponibilité du système à l'instant *t'*
- $\overline{A_s}(t')$: l'indisponibilité du système à l'instant *t'*
- $Z_{it'}$: la variable binaire
- D_{ri}: la date prévue pour effectuer une tâche de maintenance préventive sur une machine *i*
- mp_{ri}: le temps nécessaire pour effectuer une tâche de maintenance préventive sur une machine *i*
- nb_i : le nombre d'interventions de maintenance préventive sur une machine *i*

L'expression de la disponibilité d'une machine *i* dépend de son taux de défaillance λ_i et taux de réparation μ_i. Dans la suite de ce travail, nous considérons que ces deux paramètres sont constants. Nous supposons également que chaque action de maintenance remet le système à neuf "as good as new". Analytiquement, après une maintenance préventive, la fonction du taux de défaillance et le même que celui d'une machine neuve. Ceci stipule que la machine peut avoir que deux états : un état de fonctionnement ou un état d'arrêt ; sa fonction d'état peut être illustrée comme suit :

$$Z_{it'} = \begin{cases} 1 \text{si la machine i fonctionne à l'instant t'} \\ 0 \text{ si la machine i est en panne à l'instant t'} \end{cases} \quad (14)$$

Par conséquent, la disponibilité $A_i(t')$ d'une machine *i*, à partir de l'instant initial *t'=0,* jusqu'à l'instant *t'* tout en supposant que le temps avant défaillance (respectivement temps de réparation) est représenté par une distribution de probabilité exponentielle ayant comme paramètre le taux de défaillance λ_i (resp. taux de réparation paramètre μ_i) est donnée par l'expression suivante (Ebeling et al., 1997):

$$A_i(t') = \frac{\mu_i}{\lambda_i+\mu_i} + \frac{\lambda_i}{\lambda_i+\mu_i} exp[-(\lambda_i + \mu_i)t] \quad (15)$$

L'indisponibilité $\overline{A_i}(t')$ d'une machine *i*, à partir de l'instant initial *t'=0,* jusqu'à l'instant *t'* peut être calculée comme suit :

$$\overline{A_i}(t') = 1 - A_i(t') \quad (16)$$

Puisque le système étudié à une structure parallèle, possédant *m* machines parallèles identiques indépendants, ayant chacune une fonction de disponibilité $A_i(t')$, ceci implique que la disponibilité du système $A_s(t')$ à l'instant *t'* est donnée par l'expression suivante (Villemeur, 1991) (Ebeling, 1997):

$$A_s(t') = 1 - \prod_{i=1}^{m} [1 - A_i(t')] \quad (17)$$

Donc, l'indisponibilité du système $\overline{A_s}(t')$, durant les opérations de maintenances peut être représentée comme suit :

$$\overline{A_s}(t') = 1 - A_s(t') = \prod_{i=1}^{m} [1 - A_i(t')] \quad (18)$$

L'objectif est d'intégrer les services de maintenance et de production sous des contraintes de ressources non-renouvelables dans un seul modèle, ces derniers doivent collaborer ensemble pour atteindre un objectif commun, celui de la maximisation de la productivité du système. Cette stratégie de maintenance intégrée consiste à choisir dans un premier temps la machine la plus appropriée pour chaque opération et de déterminer dans un second temps la séquence des tâches assignées à la même machine tout en déterminant les instants d'interventions de la maintenance préventive afin de minimiser l'indisponibilité du système ce qui entraine la minimisation du makespan. En effet, il faut effectuer un nombre d'intervention optimal de maintenance préventive afin de diminuer l'indisponibilité du système tout en minimisant le makespan et en respectant les différentes contraintes relatives aux ressources, aux machines,.... Notons que le nombre d'intervention de la maintenance préventive n'est pas fixé à l'avance.

La nouvelle fonction objectif, sous les contraintes définies précédemment, consiste à minimiser le nouveau makespan en insérant les différentes actions de maintenance préventive sur chaque machine et à minimiser les indisponibilités des machines (voir l'exemple dans la section suivante).

4 RÉSOLUTION DU PROBLÈME

Les algorithmes génétiques sont des méthodes de recherche stochastiques conçues pour explorer des espaces de problèmes complexes afin de guider le processus de recherche et de trouver des solutions optimales en utili-

sant un minimum d'informations sur le problème. Contrairement à d'autres techniques d'optimisation, les algorithmes génétiques sont caractérisés par l'utilisation d'une population d'individus pour effectuer la recherche sur différentes régions de l'espace de recherche afin d'optimiser une fonction prédéfinie, appelée fonction objectif, ou fitness. L'algorithme génétique a connu beaucoup d'intérêt par la communauté d'optimisation à cause de sa performance dans la résolution des problèmes complexes, sa simplicité et sa facilité d'hybridation avec d'autres métaheuristiques et de son application avec succès sur une très large gamme de problèmes.

4.1 Codage

Chaque individu définit un système de configuration particulière qui reflète une solution. Nous adoptons la même représentation de solution effectuée dans Berrichi et al. (Berrichi et al., 2010). Chaque chromosome est composée de deux parties: production et maintenance ; chaque tâche est choisie conformément avec le chromosome correspondant qui sera évalué et modifié pour minimiser le makespan. Pour mieux illustrer le codage utilisé prenons l'exemple de six tâches qui doivent être exécutées sur deux machines.

j_1	j_2	j_3	j_4	j_5	j_6	Maintenance préventive	
1	**1**	**2**	**1**	**2**	**2**	**15**	**25**

Figure 2 : Codage de la solution

Les six premiers allèles déterminent l'affectation des tâches aux machines, c'est-à-dire, la partie relative à la production. Les deux derniers allèles (15, 25) représentent la partie de la maintenance qui indique qu'une intervention sur la première machine doit être effectuée toutes les 15 unités de temps et sur la seconde machine toutes les 25 unités de temps. Ces valeurs sont générées aléatoirement dans un intervalle [LB, UB].

- LB est égale au maximum entre (minimum des temps de traitement et la date de la première arrivée des composants).
- UB est égale à (le minimum entre les dates de sortie, des dernières tâches sur les machines) moins (le minimum des temps de traitement). Le minimum entre les dates de sortie, des dernières tâches sur les machines est obtenue en utilisant l'heuristique L-PT/RC sans tenir compte de la maintenance préventive ; ceci pour assurer au moins une action de maintenance sur chaque machine. L'heuristique L-PT/RC consiste à ordonnancer les tâches dans l'ordre décroissant du rapport de leurs temps de traitement et consommation des ressources t.

4.2 Évaluation de la solution

Après avoir généré tous les individus avec une partie de production et une autre de maintenance, la fonction objectif d'un chromosome peut être calculé. L'évaluation commence par affecter les taches aux machines. Ensuite, les tâches de maintenance sont insérées selon une stratégie rationnelle (Berrichi et al., 2009). Les dates d'interventions de la maintenance préventive peuvent être avancés ou retardés selon l'ordonnancement initial.

- Affecter les tâches aux machines (aléatoirement).
- Calculer la makespan du système.
- Insérez les tâches de maintenance préventive dans l'ordonnancement selon une stratégie rationnelle. La date d'intervention de la maintenance préventive peut être avancé ou retardé selon l'affectation des tâches.
- Actualiser les dates d'ordonnancement des tâches.
- Actualiser le makespan après l'insertion des actions de la maintenance préventive et calculez l'indisponibilité du système.

Afin de comprendre le comportement de notre système et la procédure d'évaluation, nous considérons l'exemple suivant: 6 tâches qui doivent être exécutés sur 2 machines. Chaque tâche consomme une seule unité d'un seul composant. Nous supposons aussi que la durée d'intervention de la maintenance préventive sur les deux machines est égale à 5 unités de temps.

Tâches	**J1**	**J2**	**J3**	**J4**	**J5**	**J6**
p_j	13	10	12	7	15	7
c_{j1}	1	1	0	1	0	0
c_{j2}	0	0	1	0	1	1

Tableau 1 : Temps de traitement et consommation des tâches

Instants	**t=0**	**t=10**	**t=20**
A_{t1}	1	2	0
A_{t2}	2	0	1

Tableau 2 : le nombre total de composant *k* arrivé jusqu'à l'instant *t*

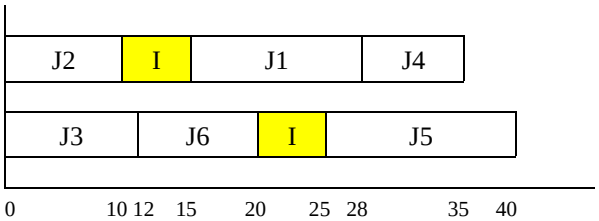

Figure 3 : Diagramme de Gantt pour l'ordonnancement de la production sans les actions de maintenance

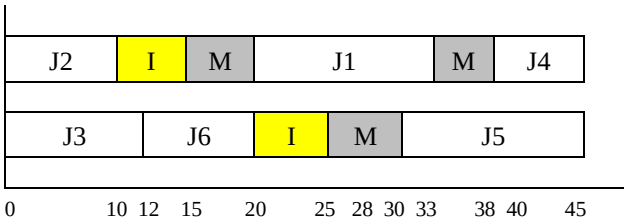

Figure 4 : Diagramme de Gantt pour l'ordonnancement de la production intégré avec les actions de maintenance

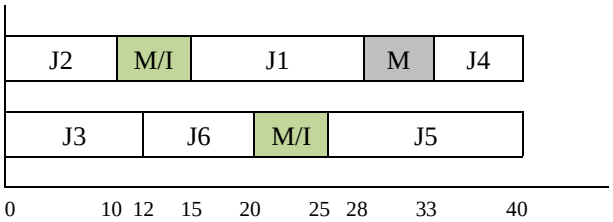

Figure 5 : Diagramme de Gantt pour l'ordonnancement de la production intégré avec les actions de maintenance et les périodes d'indisponibilités des ressources

Indisponibilité des ressources (I)

Machine indisponible pour raison de maintenance préventive (M)

Intégration de l'indisponibilité des ressources avec l'indisponibilité des machines (M/I)

Par conséquent, la structure de l'algorithme génétique hybride peut être présentée comme suit :

Algorithme : Algorithme Génétique

S'il y a un ensemble de tâches non ordonnancées **alors**
  Générer une population initiale d'une manière aléatoire
  Pour chaque chromosome
    Calculer la fonction objectif du chromosome
    Évaluer la fonction fitness du chromosome
    Si cette solution est plus petite que la meilleure solution **donc**
      Actualiser la meilleure solution
    Fin si
  Fin pour
  Tant que le critère d'arrêt n'est pas satisfait
    Appliquer l'opérateur de sélection élitiste
    Appliquer l'opérateur de croisement à deux points.
    Appliquer l'opérateur de mutation à un point
    Pour chaque chromosome
      Calculer la fonction objectif du chromosome
      Évaluer la fonction fitness du chromosome.
      Si cette solution est plus petite que la meilleure solution **donc**
        Actualiser la meilleure solution.
      Fin si
    Fin pour
    Constituer la nouvelle génération.
  Fin tant que
Fin si

5 SIMULATIONS ET RÉSULTATS

Cette section vise à valider le modèle en utilisant l'algorithme génétique (AG) et à étudier ses performances. Les expériences ont été effectuées sur le langage Java sur un Core (TM) i3 CPU - 2,13 GHz - 4,00 Go de Ram.

5.1 Analyse de sensibilité de l'algorithme génétique

Nous réalisons une analyse de sensibilité qui consiste à faire varier les paramètres de l'algorithme génétique pour trouver son meilleur paramétrage, puis pour pouvoir analyser leurs impacts sur le système. L'analyse de sensibilité est jugée par un indicateur de performance représenté par le makespan pour avoir une idée sur la qualité de la solution de cette métaheuristique. Les paramètres étudiés peuvent être résumée comme suit:

- ✓ Paramètre de mutation(P_{mut})
- ✓ Paramètre de croisement(P_{cross})
- ✓ Nombre d'itération sans amélioration ($N_{ité}$)
- ✓ Taille de population (T_{pop})

À partir différents protocoles de test que nous venons de présenter, nous pouvons constater que le réglage adéquat des paramètres de l'algorithme génétique est obtenu en retenant les valeurs qui sont résumées dans le tableau 1.

Paramètres	**Valeurs**
Mutation	0.05
Croisement	0.9
Nombre d'itérations sans amélioration	50
Taille de population	150

Tableau 3 : Paramètres de l'algorithme génétique

5.2 Environnement de test et mesures de performances

Nous réalisons une analyse de sensibilité qui consiste à faire varier

Deux paramètres peuvent être utilisés pour effectuer cette comparaison: le premier est le temps d'exécution et le second est l'écart entre la solution courante et la solution optimale. Nous générons des instances afin que le total des ressources nécessaires soit inférieur ou égal au nombre de ressources disponibles dans le système. Les contraintes suivantes doivent être respectées :

- Chaque machine doit exécuter qu'une seule tâche à la fois.
- Chaque tâche doit être traitée sans interruption.
- Tous les travaux doivent être traités pour achever le processus de fabrication.

Nous supposons que le temps de panne (resp. temps de réparation) d'une machine Mi est représenté par une distribution de probabilité exponentielle ayant comme paramètre, le taux de défaillance λi (resp. comme paramètre, le taux de réparation μi). Notons qu'au minimum, une action de maintenance préventive doit être effectuée, pour chaque machine.
Les performances du système sont évaluées en termes de mesures telles que :

- ➢ CPU_{time} est le temps d'exécution.
- ➢ Le pourcentage d'amélioration des performances du système et de diminution de l'indisponibilité des machines.
- ➢ GAP qui représente l'erreur relative entre la solution obtenue par l'AG sans maintenance préventive qui représente une borne inférieure pour notre problème et la meilleure solution obtenue pour chaque instance.

Le GAP reporté dans cette section peut être calculé par :

- $GAP = (C_{max}^{cur} - C_{max}^{best})/C_{max}^{best}$, où :
  - ✓ C_{max}^{best} est la solution obtenue par l'AG sans maintenance préventive
  - ✓ C_{max}^{cur} est la solution courante

Les tableaux (4 – 6) illustrent les résultats obtenus par l'AG pour différents problèmes. La première colonne indique l'instance étudiée. La deuxième colonne qui indique le GAP obtenu par chaque stratégie est composée elle-même de 3 autres sous-colonnes représentant respectivement :

- les résultats obtenus par l'AG Sans Maintenance préventive (notée AG-SM).
- les résultats obtenus par l'AG en intégrant les actions de Maintenance préventive avec celle des actions de Production (notée AG-MP).
- les résultats obtenus par l'AG en intégrant les actions de Maintenance préventive avec les actions de Production en intégrant les contraintes relatives aux disponibilités des Ressources (notée AG-MPR).

Instances (n, m, k)	**GAP**		
	AG-SM	**AG-MP**	**AG-MPR**
(10, 2, 2)	0.0000	0.0930	0.0465
(10, 2, 3)	0.0000	0.0952	0.0476
(20, 2, 2)	0.0000	0.0938	0.0156
(20, 2, 3)	0.0000	0.0902	0.0150
(30, 2, 2)	0.0000	0.0828	0.0138
(30, 2, 3)	0.0000	0.0787	0.0197
(40, 2, 2)	0.0000	0.0745	0.0186
(40, 2, 3)	0.0000	0.0749	0.0214
(50, 2, 2)	0.0000	0.0698	0.0203
(50, 2, 3)	0.0000	0.0701	0.0260

Tableau 4 : Comparaison entre les différents stratégies lorsque *m=2*

Instances (n, m, k)	**GAP**		
	AG-SM	**AG-MP**	**AG-MPR**
(20, 3, 2)	0.0000	0.1008	0.0336
(20, 3, 3)	0.0000	0.0984	0.0164
(30, 3, 2)	0.0000	0.0812	0.0074
(30, 3, 3)	0.0000	0.0854	0.0142
(40, 3, 2)	0.0000	0.0692	0.0122
(40, 3, 3)	0.0000	0.0709	0.0157
(50, 3, 2)	0.0000	0.0513	0.0096
(50, 3, 3)	0.0000	0.0500	0.0147

Tableau 5 : Comparaison entre les différents stratégies lorsque *m=3*

Instances (n, m, k)	**GAP**		
	AG-SM	**AG-MP**	**AG-MPR**
(20, 4, 2)	0.0000	0.1053	0.0175
(20, 4, 3)	0.0000	0.1026	0.0171
(30, 4, 2)	0.0000	0.0846	0.0077
(30, 4, 3)	0.0000	0.0821	0.0075
(40, 4, 2)	0.0000	0.0682	0.0085
(40, 4, 3)	0.0000	0.0707	0.0125
(50, 4, 2)	0.0000	0.0504	0.0101
(50, 4, 3)	0.0000	0.0503	0.0126

Tableau 6 : Comparaison entre les différents stratégies lorsque *m=4*

Les résultats obtenus par les tableaux (4 – 6) montrent que l'AG donne les meilleurs résultats en intégrant les actions de maintenance préventive avec les actions de production sous les contraintes de ressources. Ceci est du à la bonne exploitation des périodes d'indisponibilités des ressources, en essayant de combiner entre les dates d'intervention de la maintenance préventive et les périodes d'indisponibilité des ressources. Ceci que ce soit pour le nombre de machine *m=2*, *m=3* ou *m=4*.

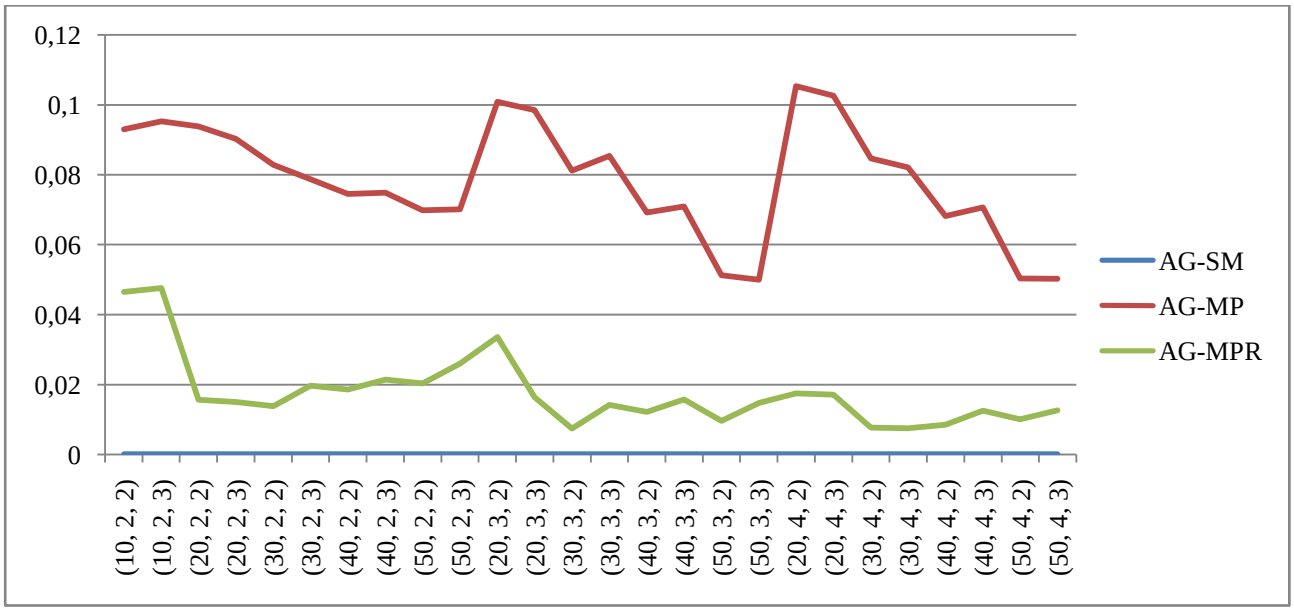

Figure 6 : Comparaison entre les différentes stratégies appliquées pour différentes tailles de problèmes

Dans la figure 6, nous remarquons que la stratégie qui intègre la maintenance avec la production donne les meilleurs résultats pour différentes tailles de problèmes car elle permet d'équilibrer l'ordonnancement en minimisant le makespan et les indisponibilités du système en exploitant efficacement les périodes d'indisponibilité des ressources et les instants d'intervention de la maintenance préventive. Donc la stratégie intégrée proposée permet une amélioration des performances du système qui peut être illustrée dans la figure 7.

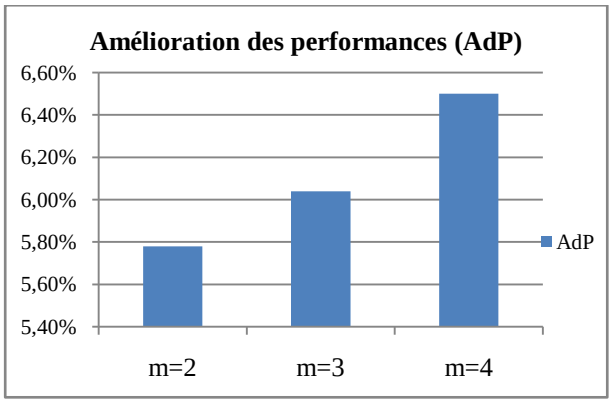

Figure 7 : Amélioration des performances du système

D'après cette figure nous remarquons que le modèle qui intègre la production, la maintenance et la gestion des ressources améliore considérablement les performances du système. En outre, nous pouvons remarquer que l'écart continu à s'améliorer lorsque la taille du problème augmente.

Pour ce qui concerne l'indisponibilité du système, nous pouvons remarquer dans la figure 8 qu'il y a une amélioration significative dans les résultats de l'approche intégrée, lorsque les actions de maintenance préventive sont insérées avec les périodes d'indisponibilités des ressources consommables. Ceci permet d'optimiser les périodes d'indisponibilités des machines.

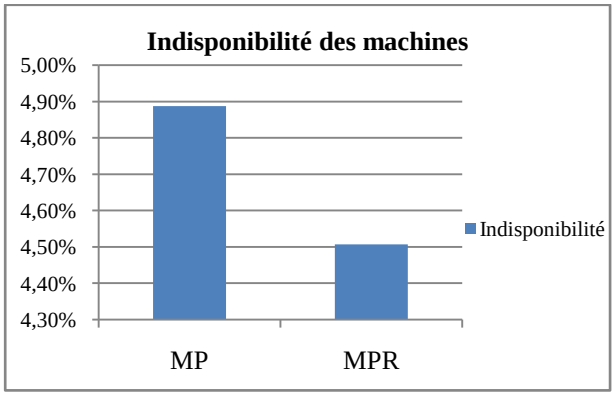

Figure 8 : Impact des deux stratégies sur l'indisponibilité des machines

De plus, nous pouvons noter que pour ce qui concerne, le temps d'exécution, il reste relativement petit pour toutes les instances considérées, n'atteignant pas les 1.5 secondes. Des résultats prometteurs ont été obtenus et plusieurs pistes de recherches sont à développer.

6 CONCLUSION ET PERSPECTIVES

Dans ce travail, nous avons proposé une stratégie visant à intégrer la gestion des actions de maintenance préventive avec celle des actions de production tout en tenant compte des contraintes de ressources non-renouvelables.
Le modèle que nous avons suggéré prend en compte la fiabilité des machines pour la réalisation de la maintenance préventive. Pour résoudre notre problème, nous avons adapté une métaheuristique basée sur les algorithmes génétiques qui intègre les deux aspects avec des ressources consommables, afin de minimiser le makespan et les indisponibilités des machines. Cette métaheuristique consiste à choisir simultanément, la machine la plus appropriée pour chaque opération, à déterminer le séquencement des tâches affectées à la même machine et à définir les instants de réalisation de la maintenance préventive. Les résultats de simulations ont démontré que l'approche intégrée proposée, reste la plus efficace pour la résolution du problème d'ordonnancement de la production et de la maintenance dans un environnement de machines parallèles sous des contraintes de ressources consommables, car ceci permet d'exploiter efficacement les périodes d'indisponibilité des ressources et les instants d'intervention de la maintenance préventive.

L'étude réalisée dans ce papier représente un potentiel considérable vue la relation d'interdépendance entre la stratégie de production et de maintenance ; et son impact sur le processus décisionnel afin d'améliorer la productivité des entreprises et d'assurer la pérennité de leurs systèmes de production. Ce domaine reste très vaste et dont l'importance ne cesse de croître. Cependant, les systèmes de production sont très complexes et les critères liés à la production sont généralement antagoniste avec ceux de la maintenance, ce qui peut mener à de nouvelles circonstances. Par conséquent, notre travail doit être enrichi en proposant d'autres solutions pratiques dans les domaines suivants :

- Dans cette étude, nous avons supposé que les machines sont identiques, cependant, les machines peuvent avoir une vitesse de traitement différente et un taux de dégradation spécifique. Pour cela, nous suggérons de considérer le système avec d'autres variantes, tout en essayant d'appliquer d'autres politiques de maintenance sur ce problème
- Les recherches menées se concentrent sur la réduction d'un seul objectif. Cependant, de nombreux objectifs peuvent être bien justifiés dans la pratique qui peuvent être souvent contradictoires pour joindre et optimiser les services de production et de maintenance. Dans ce sens, nous proposons d'utiliser des approches multi-objectif.
- Finalement, la planification de la production est supposée déjà faite, mais, ce même problème peut être identifié dans plusieurs structures, donc, il serait judicieux d'intégrer la maintenance des systèmes de production, la planification et l'ordonnancement de la production dans un seul modèle tout en essayant d'adopter de nouvelles approches génériques.

REMERCIEMENTS

The author would like to thank the Erasmus Mundus EU-METALIC project team Coordinated by Cardiff Metropolitan University. "This project has been funded with support of the European Commission. This communication reflects the view only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein."

REFERENCES

Belkaid, F., Z. Sari, and M. Souier, 2013a. A Genetic Algorithm for the Parallel Machine Scheduling Problem with Consumable Resources. *International Journal of Applied Metaheuristic Computing (IJAMC) 4 (2),* p.*17-30.*

Belkaid, F., F. Yalaoui, and Z. Sari, 2013b. A hybrid genetic algorithm for parallel machine scheduling problem with consumable resources. *IEEE International Conference on Control Decision and Information Technologies CoDIT'13.*

Berrichi, A., L. Amodeo, F. Yalaoui, E. Châtelet, and M. Mezghiche, 2009a. Bi-objective optimization algorithms for joint production and maintenance scheduling: application to the parallel machine problem. *Journal of Intelligent Manufacturing*, *20*(4), p. 389-400.

Berrichi, A., L. Amodeo F. Yalaoui, and M. Mezghiche, 2010b. "Bi-Objective Ant Colony Optimization approach to optimize production and maintenance scheduling", Computers & Operations Research, Vol. 37, p. 1584–1596.

Berrichi, A., and F. Yalaoui, 2013. Efficient bi-objective ant colony approach to minimize total tardiness and system unavailability for a parallel machine scheduling problem. *The International Journal of Advanced Manufacturing Technology*, *68*(9-12), p. 2295-2310.

Blazewicz, J.,J. K. Lenstra and A. H. G. Rinnooy Kan. 1983. Scheduling subject to resource constraints: classification and complexity. *Discrete Applied Mathematics*, 5(1), p. 11–24.

Carlier, J. and A. H. G. Rinnooy Kan, 1982. Scheduling subject to nonrenewable resource constraints. *Operations Research Letters*, 1:52–55.

Cassady, CR., E. Kutanoglu, 2003. Minimizing job tardiness using integrated preventive maintenance planning and production scheduling. *IIE Trans*, 35:503– 513.

Dahane, M., C. Clementz, and N. Rezg, 2010. Effects of extension of subcontracting on a production system in a joint maintenance and production context. *Computers & Industrial Engineering*, *58*(1), p. 88-96.

Daniels, R. L., B. J. Hoopesand, andJ. B. Mazzola, 1997. An analysis of heuristics for the parallel-machine flexible-resource scheduling problem. *Annals of Operations Research* 70, p. 439–472.

Daniels, R. L., S. Y.Hua, and S. Webster, 1999. Heuristics for parallel-machine flexible resource scheduling problems with unspecified job assignment. *Computers and Operations Research* 26, p. 143–155.

Ebeling, CE., 1997. An introduction to reliability and maintainability engineering. McGraw-Hill, USA.

Edis, E. B., C. Oguz, and I. Ozkarahan, 2013. Parallel Machine Scheduling with Additional Resources: Notation, Classification, Models and Solution Methods. *European Journal of Operational Research*.

Kaabi J., C. Varnier, N. Zerhouni, 2002a. Heuristics for schedulingmaintenance and production on a single machine. IEEEConference on Systems, Man and Cybernetics. October 6–9Hammamet, Tunisia.

Kaabi J., C. Varnier, N. Zerhouni, 2003b. Genetic algorithm for scheduling production and maintenance in a Flow Shop. Laboratory of automatic of Besançon, France.

Kai Li, Ye Shi., Y. Shan-lin, andC. Ba-yi, 2011. Parallel machine scheduling problem to minimize the makespan with resource dependent processing times, *Applied Soft Computing*, p. 5551-5557.

Lee, C.-Y., 1996. Machine scheduling with an availability constraint. *Journal of Global Optimization*, *9*, p. 395–416.

Lee, K., L. Lei, M. Pinedo and S. Wang, 2013. Operations scheduling with multiple resources and transportation considerations. *International Journal of Production Research*, *51*(23-24), p.7071-7090.

Lenstra, J. K., A. H. G. Rinnooy Kan, and P. Brucker, 1977. Complexity of Machine Scheduling Problems, *Annals of Discrete Mathematics*, Vol. 1, p.342-362.

Sethi, R, 1977. On the complexity of mean flow time scheduling, *Mathematics of Operations Research*, 2(4), p. 320-330.

Liao, C. J., C. M. Chen, and C. H. Lin, 2007. Minimizing makespan for two parallel machines with job limit on each availability interval. *Journal of the Operational Research Society*, *58*(7), p. 938–947.

Moradi, E., S. M. T. Fatemi Ghomi, and M. Zandieh, 2011. Bi-objective optimization research on integrated fixed time interval preventive maintenance and production for scheduling flexible job-shop problem. *Expert systems with applications*, *38*(6), p. 7169-7178.

Nahmias, S., 2004. *Production and Operations Analysis*. McGraw Hill Higher Education.

Ruiz, R., J. C. García-Diaz, and C. Maroto, 2007. Considering scheduling and preventive maintenance in the flowshop-sequencing problem. *Computers & Operations Research*, 34(11), p. 3314–3330.

Schmidt, G., 2000. Scheduling with limited machine availability. *European Journal of Operational Research*, 121, p. 1–15.

Villemeur, A., 1991. Reliability, availability, maintainability and safety assessment. Wiley, USA.

Xu, D., K. Sun, and H. Li, 2008. Parallel machine scheduling with almost periodic non-preemptive maintenance and jobs to minimize makespan. Computers and Operations Research, 35, p. 1344-1349.