

Problème d'Ordonnancement d'Atelier dans un Environnement Incertain et Polyvalent

MOHAMED FRIHAT¹, CHÉRIF SADFI¹, ATIDEL B.HADJ ALOUANE¹

¹OASIS, ENIT

1002 Tunis, Tunisie

frihatmohamed@hotmail.fr, cherif.sadfi@gmail.com, atidel.hadj@enit.rnu.tn

Résumé - Ce papier traite un problème d'ordonnancement, très souvent rencontré dans l'industrie, en général, et dans les tanneries, en particulier. Nous présentons un problème d'ordonnancement de production et d'affectation des employés sous différentes contraintes. Le problème d'ordonnancement de production est un job shop hybride soumis à des contraintes temporelles d'attente « Time Lags » et de changement de série « Setup Times ». Dans plusieurs cas, ce problème est couplé au problème de planification des employés. Ce dernier consiste à affecter les opérateurs aux différentes tâches en considérant leurs contraintes de disponibilité et les contraintes de qualification imposées. De plus, l'environnement n'est pas toujours stable, plusieurs incertitudes existent et différentes perturbations peuvent intervenir. En effet, les durées opératoires sont incertaines, les pannes des machines, l'absentéisme et la non-conformité des articles sont des aléas possibles et fréquents. Pour bien estimer ces différentes incertitudes, nous employons la logique floue, basée sur l'expérience et l'expertise humaine. L'objectif visé est d'avoir un ordonnancement réalisable, stable et robuste, permettant de minimiser le makespan et le retard maximal. Dans ce cadre, nous proposons des mesures de robustesse bien adaptées à notre problème et une heuristique constructive à base de règles de priorité assurant un compromis entre les différents objectifs considérés. Une étude comparative avec d'autres règles de priorité, basée sur des instances réelles d'une tannerie tunisienne, permet d'évaluer les performances de notre heuristique.

Abstract - This paper studies a scheduling problem, often encountered in the manufacturing industry and particularly in the tanneries. We study an integrated production scheduling and employee timetabling problem under various constraints. We consider a hybrid job shop scheduling problem with time lags and sequence-dependent setup times. The employee timetabling problem consists of assigning employees to tasks under availability and qualification constraints. In addition, the environment is not always stable, many uncertainties exist and various disturbances may occur. Indeed, uncertain processing times, machine failure, operator absenteeism and product nonconformity are frequent disturbances. In order to estimate these uncertainties, we use fuzzy logic reasoning, based on human expertise and experience. The objective is the determination of a feasible, stable and robust solution, minimizing the makespan and the maximum lateness. Furthermore, we propose robustness measures well adapted to our problem. A constructive heuristic algorithm, based on priority rules, is proposed, which ensures a compromise between the different objectives. A comparative study with other priority rules, using real instances of a Tunisian leather tannery, is used to evaluate our heuristic performances.

Mots clés - ordonnancement, makespan, retard maximal, robustesse, heuristique.

Key words - scheduling, makespan, maximum lateness, robustness, heuristic.

1 INTRODUCTION

L'évolution que connaissent le monde industriel et la pression concurrentielle poussent les entreprises à améliorer leurs méthodes de production. Quoique plusieurs logiciels de gestion de production mettent à disposition des gestionnaires d'atelier un outil d'ordonnancement, ces derniers demandent toujours un outil plus spécifique, mieux adapté et plus performant. Dans le secteur manufacturier, nous nous intéressons au problème réel et plus complexe intégrant l'ordonnancement de la production et la planification des employés. En effet, l'ordonnancement doit prendre en compte les ressources particulières de l'entreprise, les

contraintes temporelles telles que les délais d'attente minimaux et/ou maximaux (Time Lags) entre les tâches et le temps de changement de série (Setup Time). Il doit aussi prendre en compte toutes les contraintes de ressources (opérateurs et machines) et les différentes incertitudes et perturbations qui peuvent intervenir. C'est en particulier la problématique de tanneries où la variété et le nombre des articles, la complexité de l'organisation des ateliers « Job-Shop », la flexibilité des machines et la polyvalence des employés, rendent la gestion de la production assez compliquée. Afin d'avoir un planning d'ordonnancement faisable, robuste et stable, nous devons

intégrer les différentes contraintes du problème et les différentes incertitudes possibles.

Ce papier est scindé en quatre parties : D'abord, nous présentons un état de l'art sur le problème couplé d'ordonnancement et de planification des employés, ainsi que l'ordonnancement dans un contexte incertain. Ensuite, nous présentons notre problème et nous détaillons la méthode de logique floue permettant d'estimer certaines incertitudes. Par la suite, nous définissons des mesures de robustesse et de stabilité et nous proposons un modèle mathématique pour notre problème. Enfin, nous proposons une heuristique constructive à base de règles de priorité et nous évaluons ses performances sur des instances réelles de notre partenaire industriel.

2 ETAT DE L'ART

Etant donné la complexité d'intégrer la planification des employés et l'ordonnancement de la production, peu de travaux sont menés dans ce domaine. [Daniels et Mazzola, 1994] considèrent le problème d'ordonnancement de flow shop où chaque opération doit être traitée par sa machine dédiée et chaque employé ne peut être affecté qu'à un ensemble d'opérations selon ses compétences acquises. Les auteurs proposent la méthode de Branch and Bound pour la résolution d'un tel problème. Dans le même contexte, [Alfares et Bailey, 1997] proposent un programme linéaire en nombre entiers et une méthode de programmation dynamique basée sur des heuristiques. En outre, [Artigues et al, 2007] proposent un état de l'art sur une telle problématique dans les domaines de la production et du transport et une technique de résolution hybride combinant la programmation linéaire et la programmation par contraintes. [Hooker, 2005] s'intéresse aussi dans ses travaux à cette problématique et propose une méthode de résolution hybride, basée sur la décomposition de Benders. [Guyon et al, 2010] proposent pour le problème job shop et la planification des employés un modèle indexé sur le temps et une méthode de résolution de type Branch and Cut.

Plus récemment, [Frihat et al, 2014] proposent un modèle intégrant la planification des employés et l'ordonnancement de job shop hybride avec des contraintes d'attente « time lags ». Les auteurs proposent des méthodes exactes basées sur les techniques de coupes et les approches hybrides (programmation linéaire et programmation par contraintes).

Ces travaux traitent un problème d'ordonnancement statique et déterministe sans tenir compte les différentes perturbations et incertitudes possibles. [MacCarthy et Liu, 1993] étudient l'écart entre l'ordonnancement théorique et pratique et comment les modèles d'ordonnancement classiques ne permettent pas de répondre aux besoins de l'environnement réel. Plusieurs perturbations possibles dans différentes organisations d'atelier sont étudiées dans la littérature avec trois différentes approches : ordonnancement réactif, ordonnancement prédictif réactif et ordonnancement proactif robuste.

L'ordonnancement réactif permet de prendre les décisions en temps réel. Aucun planning n'est établi en avance, cette approche repose souvent sur des règles de priorité.

L'approche prédictive réactive est très utilisée dans l'ordonnancement dynamique. Cette approche, basée sur l'ordonnancement et le ré-ordonnancement, consiste à réparer et réviser le planning d'ordonnancement prévu pour faire face aux événements imprévus. Différentes stratégies de ré-ordonnancement

sont proposées et divisées en deux classes : le ré-ordonnancement partiel et le ré-ordonnancement complet. [Sabuncuoglu et Bayiz, 2000] montrent l'efficacité du ré-ordonnancement partiel par rapport au ré-ordonnancement complet en termes de stabilité et coût du calcul. Pour conclure sur les performances des stratégies de ré-ordonnancement, différentes mesures de robustesse et de stabilité sont introduites.

L'ordonnancement proactif robuste consiste à établir un ordonnancement prédictif capable d'absorber les différentes perturbations. [Mehta et Uzsoy, 1999] proposent un ordonnancement prédictif basé sur l'insertion de tampons de temps, pour leur problème à machine unique avec des pannes, en considérant la minimisation du retard maximal comme objectif.

Pour la résolution du problème d'ordonnancement incertain, différentes techniques sont proposées dans la littérature : heuristiques, méthaheuristiques, logique floue, réseaux de neurones, techniques hybrides, systèmes multi-agents.

Différentes heuristiques de ré-ordonnancement sont proposées comme le décalage à droite « right-shift schedule repair », ré-ordonnancement avec correspondance « match-up schedule repair » et le ré-ordonnancement partiel « partial schedule repair » [Ouelhadj et Petrovic, 2008].

[Yamamoto et Nof, 1985] étudient les performances de la technique « right shift ». Les résultats expérimentaux montrent que cette technique est plus performante que le ré-ordonnancement complet et les règles de priorité.

[Bean et al, 1991] proposent la technique « match-up schedule repair » pour un atelier avec différentes ressources, en considérant les pannes machines. Les résultats montrent les performances de cette heuristique par rapport au ré-ordonnancement complet, en fournissant des solutions quasi-optimales. En outre, différentes règles de priorité sont testées et comparées dans un environnement dynamique et stochastique, en utilisant la simulation. [Rajendran et Holthaus, 1999] présentent un état de l'art sur les différentes règles de priorité dans un job shop et flow shop. De plus, certains travaux proposent des méthaheuristiques pour le ré-ordonnancement. [Dorn et al, 1995] utilisent la recherche tabou pour réparer le planning d'ordonnancement, en considérant des durées opératoires incertaines. [Zweben et al, 1994] utilisent le recuit simulé pour le ré-ordonnancement. La méthode utilisée consiste à choisir une technique de réparation parmi cinq heuristiques en utilisant une fonction de choix, et à utiliser le recuit simulé afin de générer plusieurs itérations de réparation. [Chrysolouris et Subramaniam, 2001] proposent un algorithme génétique pour le problème de job shop, en considérant des pannes machines. Leur algorithme génère un ordonnancement alternatif suite à chaque événement imprévu. [Rossi et Dini, 2000] utilisent aussi l'algorithme génétique pour améliorer les solutions générées par les règles de priorité dans un atelier flexible en considérant différentes sources de perturbation.

Pour notre problème, nous proposons une approche proactive robuste permettant de prendre en compte les différentes incertitudes dans l'ordonnancement établi.

3 PRESENTATION DU PROBLEME

Dans notre étude, nous nous intéressons à l'allocation des différentes ressources (machines, employés) dans un atelier de type job-shop hybride à haute technicité. Plus précisément, nous

considérons le cas d'une tannerie dans laquelle la production à réaliser requiert divers types de machines flexibles dans des séquences variées intégrant des contraintes d'attente « time lags » et de changement de série « setup times ». Chaque machine nécessite, pour son pilotage, la présence des agents qualifiés. De plus, les ressources humaines sont assujetties à des contraintes d'indisponibilité.

3.1 Problème d'ordonnement de production

Nous considérons un ensemble d'étages $\{R_k\}_{k=1,\dots,r}$ et un ensemble de jobs $\{I_i\}_{i=1,\dots,n}$. Chaque job i est une séquence d'opérations non préemptives $\{O_{ij}\}_{j=1,\dots,n_i}$, et caractérisé par sa date de livraison L_i et son taux de non-conformité $(NC)_i$. Chaque étage R_k regroupe un ensemble de machines identiques $\{M_{k1}\}_{1=1,\dots,m_k}$. Chaque machine M_{k1} est caractérisée par son taux de panne $(PM)_{k1}$ et sa distribution de temps de réparation. Chaque opération O_{ij} est définie par son étage dédié R_{ij} et sa durée opératoire $p_{ij} > 0$. L'ensemble d'opérations associé à l'étage R_k est noté O_k . Les contraintes de précédences sont remplacées par des contraintes de « time lags » ($TL_{\min,ij} \geq 0$ et $TL_{\max,ij} \geq 0$ sont respectivement le temps d'attente minimal et maximal entre les opérations O_{ij} et O_{ij+1}). En outre, nous définissons, pour chaque étage, un temps de changement de série $ST_{k,i,i'} \geq 0$ dépendant de la séquence des opérations.

De plus, nous définissons pour chaque job, une opération supplémentaire de retraitement, en cas de non-conformité. Nous désignons par F_{ik} la probabilité de retraitement du job i sur l'étage k et α_{ik} la durée opératoire associée. Nous ajoutons qu'un job peut passer plusieurs fois par le même étage (flux réentrant).

Nous considérons un horizon d'ordonnement H (borne supérieure du makespan) divisé en σ périodes ou tranches horaires. L'ensemble de tranches horaires est noté $\{S_s\}_{s=1,\dots,\sigma}$. Toutes les tranches horaires ont la même durée π et toutes les durées considérées sont des entiers et sont multiples de la durée minimale π .

Le problème d'ordonnement de production étudié ici est alors la variante décisionnelle d'un problème de job-shop hybride avec des contraintes de « Time Lags » et « Setup Times ».

Les variables de décisions définies pour ce sous problème sont :

S_{ij} : date de début de l'opération O_{ij} , $\forall i \in I, j \in \llbracket 1, \dots, n_i \rrbracket$;
 $U_{ij,i'j'} = 1$: si l'opération O_{ij} précède l'opération $O_{i'j'}$; 0 sinon ($\forall i, i', j, j' \setminus R_{ij} = R_{i'j'}$) ;

$Y_{ijm} = 1$: si l'opération O_{ij} est exécutée sur la machine m de l'étage R_{ij} ; 0 sinon, $\forall i \in I, j \in \llbracket 1, \dots, n_i \rrbracket, m \in R_{ij}$.

Nous ajoutons les variables auxiliaires suivantes, indispensables pour la formulation de notre problème.

C_{ij} : date de fin de l'opération O_{ij} , $\forall i \in I, j \in \llbracket 1, \dots, n_i \rrbracket$;

C_{\max} : le makespan ;

L_{\max} : le retard maximal ;

$\theta_{ijms} = 1$ si l'opération O_{ij} est en cours d'exécution sur la machine m durant la tranche horaires s ; 0 sinon, $\forall i \in I, j \in \llbracket 1, \dots, n_i \rrbracket, s \in S, m \in R_{ij}$;

$\phi_{m,s} = 1$ si la machine m est en cours de fonctionnement durant la tranche horaire s ; 0 sinon, $\forall k \in \llbracket 1, \dots, r \rrbracket, m \in R_k, s \in S$.

3.2 Problème de planification des employés

Chaque instance fait intervenir un ensemble des employés $\{E_e\}_{e=1,\dots,\mu}$ et un ensemble des compétences $\{C_c\}_{c=1,\dots,a}$. A chaque employé e , nous associons un ensemble C_e des compétences possibles et un ensemble S_e de tranches horaires pour lesquelles il est disponible.

Chaque étage k nécessite pour le pilotage d'une machine, b_{ck} employés affectés avec la compétence c ($b_{ck} \geq 0$).

Nous notons qu'un employé ne peut pas changer la compétence avec laquelle il est affecté durant la même tranche horaire. De plus, chaque employé e est caractérisé par son taux d'absentéisme $(AE)_e$.

Les variables de décisions définies pour ce sous problème sont :

$x_{ecs} = 1$ si l'employé e est affecté avec la compétence c durant la tranche horaire s ; 0 sinon, $\forall e \in E, c \in \llbracket 1, \dots, a \rrbracket, s \in S$.

Nous ajoutons les variables auxiliaires suivantes, nécessaires pour la formulation de notre problème.

$z_{eij s} = 1$ si l'employé e est affecté à l'opération O_{ij} durant la tranche horaire s ; 0 sinon, $\forall e \in E, i \in I, j \in \llbracket 1, \dots, n_i \rrbracket, s \in S$;

$\theta_{e s} = 1$ si l'employé e est affecté durant la tranche horaire s ; 0 sinon, $\forall e \in E, s \in S$.

3.3 Objectifs

L'objectif de notre problème est de déterminer une solution réalisable permettant de minimiser le makespan et le retard maximal, garantir la stabilité et maximiser la robustesse de l'ordonnement proposé. Il s'agit d'un problème multi-objectif à trois niveaux de décision : l'opération à placer, la machine à choisir et les employés à affecter.

A notre connaissance, le problème présenté n'a pas été traité auparavant dans la littérature, même avec son aspect déterministe. En effet, nous considérons, simultanément, dans notre étude, les perturbations et les incertitudes les plus fréquentes dans l'industrie manufacturière afin de bien représenter l'environnement réel.

4 MODELISATION DES INCERTITUDES

Dans l'environnement de tanneries, plusieurs types d'incertitudes et perturbations existent. Il est très important de quantifier ces dernières et les prendre en compte dans l'élaboration du planning d'ordonnement. Pour estimer ces incertitudes, nous procédons par la logique floue.

4.1 Raisonnement Flou

Le raisonnement flou consiste à manipuler des inputs réels dans l'univers du flou et les repasser dans l'univers des réels pour répondre à un problème. Le mécanisme de la logique floue se décompose en trois grandes étapes :

- ❖ **Fuzzification** : est l'étape qui permet de passer du monde réel au monde flou, c'est-à-dire d'affecter à des données réelles une qualification dans le « langage naturel ». Elle consiste à déterminer les variables floues d'entrée et de sortie du modèle, leurs ensembles flous associés, leurs univers de discours, ainsi que les leurs fonctions d'appartenance, en se basent sur des jugements d'experts.
- ❖ **Inférence floue** : Cette étape consiste à construire les règles de décision, déterminer le degré d'activation de chaque règle déclenchée ainsi que la fonction d'appartenance de sa

conclusion. Après, il s'agit d'agrèger les fonctions d'appartenance des conclusions pour obtenir la fonction d'appartenance de la conclusion finale. Nous précisons que nous utilisons la méthode Mamdani pour calculer le degré d'activation d'une règle de décision, l'opérateur MIN pour déduire la fonction d'appartenance associée et l'opérateur MAX pour déterminer celle de la conclusion finale.

❖ **Défuzzification** : permet une prise de décision réelle à partir de la fonction d'appartenance de la conclusion finale. Dans notre travail, nous utilisons la méthode du centre de gravité.

Dans la suite, nous utilisons la logique floue pour estimer le temps opératoire d'une opération, le taux d'absentéisme d'un employé et le taux de non-conformité d'une commande, en considérant le cas de notre partenaire industriel.

4.2 Taux de non-conformité

Nous nous intéressons dans ce paragraphe à l'application de la logique floue pour estimer le taux de non-conformité d'une commande donnée (la non-conformité par rapport les spécifications du client), afin de prendre les mesures nécessaires, bien estimer la charge de la production et prévoir des opérations supplémentaires de retraitement (des opérations pour la correction de toucher, couleur, souplesse, brillance, etc.).

Nous adaptons alors les différentes étapes citées précédemment, en se basant sur l'expérience et l'expertise humaine.

Pour calculer le taux de non-conformité, nous devons déterminer les facteurs qui influent sur sa valeur.

Après interrogation des responsables de production, nous définissons trois variables floues : deux variables d'entrée (la fréquence de fabrication de l'article et le degré de complexité de l'article qui dépend de sa couleur, son aspect, son toucher et sa gamme opératoire), et une variable de sortie (le taux de non-conformité).

Les figures 1, 2 et 3 représentent les fonctions d'appartenance pour les entrées et la sortie

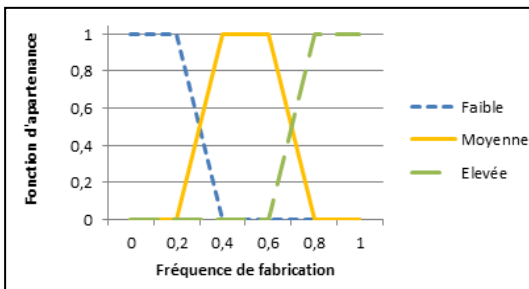


Figure 1. Fonction d'appartenance de la fréquence de fabrication

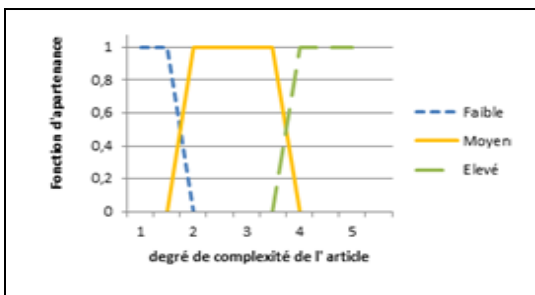


Figure 2. Fonction d'appartenance du degré de complexité de l'article

La matrice de décisions (tableau 1) a été construite de manière que la fréquence de fabrication et le degré de complexité de l'article influent conjointement sur le taux de non-conformité, c'est-à-dire que toutes les liaisons entre les prémisses sont des « ET » logiques.

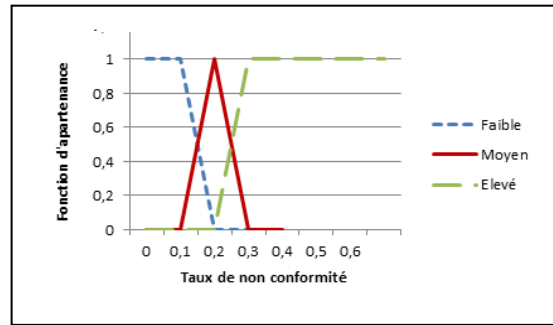


Figure 3. Fonction d'appartenance du taux de non-conformité

Tableau 1. Matrice de décisions pour le taux de non-conformité

Fréquence de fabrication	Degré de complexité		
	Faible	Moyen	Élevé
Faible	Moyen	Élevé	Élevé
Moyenne	Faible	Moyen	moyen
Élevée	Faible	Faible	moyen

4.3 Taux d'absentéisme

Nous nous intéressons maintenant à l'estimation du taux d'absentéisme de chaque employé par la logique floue.

Comme dans le cas précédent, il faut trouver les facteurs qui influent sur le taux d'absentéisme. Nous définissons deux variables d'entrée : le taux d'absentéisme calculé à partir de l'historique et l'état du climat.

L'ensemble des règles de décision est présenté dans le tableau 2.

Tableau 2. Matrice de décisions pour le taux d'absentéisme

État de climat	Taux d'absentéisme selon l'historique		
	Faible	Moyen	Élevé
Pluvieux	Faible	Élevé	Élevé
Non pluvieux	Faible	Moyen	Moyen

4.4 Durée opératoire

Le calcul du temps opératoire d'une tâche, sur certaines machines (machines de finition), dépend principalement du degré de complexité de l'article et de la qualité des peaux. Le raisonnement par la logique floue nous conduit à déterminer un coefficient correctif permettant d'estimer le temps d'exécution d'une opération

Temps opératoire = temps théorique × (1 + Coefficient correctif)
 Les variables retenues après interrogation des responsables sont deux variables d'entrée : le degré de complexité de l'article et la qualité des peaux (plus la peau présente des défauts, plus le temps de traitement est long afin de corriger ces derniers), et une variable de sortie : le coefficient correctif.

Les figures 7 et 8 représentent respectivement les fonctions d'appartenance de la qualité des peaux et le coefficient correctif.

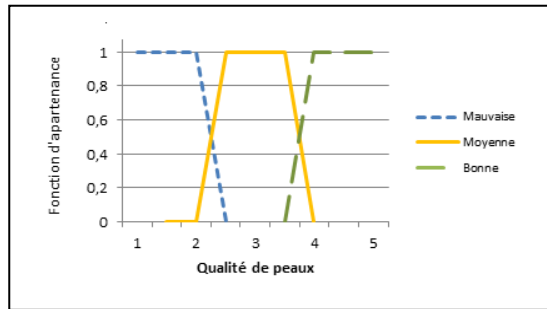


Figure 7. Fonction d'appartenance de la qualité de peaux

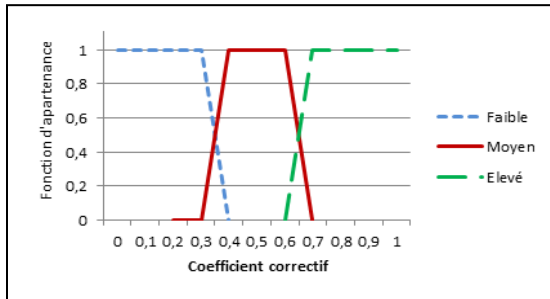


Figure 8. Fonction d'appartenance du coefficient correctif

L'ensemble des règles de décision est présenté dans le tableau 3.

Tableau 3. Matrice de décisions pour le coefficient correctif

	Degré de complexité de l'article		
Qualité de peaux	Faible	Moyen	Élevé
Mauvaise	Moyen	Élevé	Élevé
Moyenne	Faible	Moyen	Élevé
Bonne	Faible	Faible	Moyen

Dans la suite, nous désignons par \tilde{x} la valeur floue de x .

5 ROBUSTESSE ET STABILITE

5.1 Définitions

Outre les mesures de performance classiques comme le makespan et le retard maximal, des nouvelles mesures sont introduites dans la littérature récente, et qui sont principalement la robustesse et la stabilité. Ces dernières, sont particulièrement utilisées dans le cas d'un environnement incertain.

La robustesse et la stabilité sont liées à la déviation entre l'ordonnancement initial prédictif « Off_Line » et l'ordonnancement réalisé « On_Line ». En effet, la robustesse est basée sur la différence en termes de performance (valeur de la fonction objectif). Un ordonnancement est dit alors robuste, si les perturbations ne détériorent pas trop ses performances. Quant à la stabilité, elle est basée sur la différence entre le planning prévisionnel et le planning réalisé (en termes de dates ou/et séquences) [Wu et al, 1993].

5.2 Mesures proposées dans la littérature

En considérant une représentation de perturbations basée sur les scénarii, plusieurs mesures de robustesse et de stabilité sont proposées dans la littérature. Pour la robustesse, nous distinguons entre des mesures basées sur les performances réalisées comme la minimisation de la valeur moyenne de

performance réalisée. D'autres mesures sont basées sur le regret (se définit comme la différence entre la performance optimale et la performance réalisée) comme la minimisation de la valeur moyenne du regret. Différentes mesures sont aussi définies pour la stabilité. Ces mesures sont basées sur la différence entre les dates de début/fin des opérations, ou le nombre de changements dans le planning d'ordonnancement prévisionnel ou aussi le nombre des opérations perturbées.

Ces mesures de robustesse et stabilité basées sur les scénarii sont très coûteuses en termes de temps de calcul. Pour cette raison, certaines études proposent différentes approches permettant d'avoir une bonne estimation de la robustesse et/ou la stabilité de la solution, qui est facile à calculer et bien corrélée avec la valeur de la mesure réelle.

Parmi ces approches, [Leon et al, 1994] étudient la robustesse dans un environnement de type job-shop. La mesure de performance considérée est le makespan et la mesure de robustesse introduite est une combinaison convexe entre la moyenne et la variance du makespan. D'autres mesures de robustesse sont proposées aussi pour le makespan, et qui sont basées sur la marge libre (définie pour deux opérations consécutives comme la différence entre la date de fin de la première et la date de début de la deuxième). De plus, [Jensen, 2003] propose deux mesures de robustesse basées sur le voisinage. Il considère le problème de job shop avec des pannes machines et la minimisation du makespan comme critère. La première mesure est la moyenne du makespan de solutions voisines. La deuxième mesure est une estimation de la première (borne supérieure).

Pour la stabilité, nous citons l'étude de [Wu et al, 1993] portant sur le problème de ré-ordonnancement à une seule machine avec des arrêts. Les auteurs définissent deux mesures de stabilité : la première est basée sur les temps de début de jobs et la deuxième est basée sur la séquence de jobs. De plus, [Mehta et Uzsoy, 1999] considèrent le problème de job shop avec de pannes machines aléatoires et la minimisation du retard maximal comme critère. Ils proposent de mesurer la stabilité par la moyenne de la déviation de temps de fin des opérations.

5.3 Nouvelles mesures proposées

5.3.1 Stabilité

Dans le cas des tanneries, il est plus raisonnable d'adopter une mesure de stabilité basée sur la séquence des opérations. En effet, certaines opérations de finissage nécessitent un prétraitement comme la recherche de la couleur, l'élaboration des formules chimiques et la préparation des mélanges. Par conséquent, nous devons procéder la stratégie « right shift » pour le ré-ordonnancement permettant d'assurer les mêmes séquences de jobs sur les différents étages avec un simple décalage à droite en temps induit par un événement imprévu. Cette stratégie permet d'obtenir un ordonnancement stable selon la mesure proposée (la séquence des opérations).

5.3.2 Robustesse

Dans ce paragraphe, nous proposons deux mesures de robustesse associées aux deux critères considérés et bien adaptées à notre problème, en considérant les perturbations suivantes :

- L'absentéisme : l'absence imprévu d'un employé ;
- La Panne machine : l'arrêt imprévu d'une machine ;
- La Non-conformité : la commande (job) est jugée non-conforme et nécessitant une opération de retraitement.

De plus, nous ajoutons à la règle de ré-ordonnement « right shift », permettant de garantir la stabilité de la solution, les règles de ré-ordonnement suivantes :

- Traiter l'opération perturbée sur une autre machine parallèle dans le cas d'une panne-machine ;
- Affecter l'opération perturbée à un autre employé compétent dans le cas d'une absence ;
- Traiter immédiatement l'opération de retraitement, si les ressources dédiées sont disponibles durant la durée de retraitement, dans le cas d'une non-conformité.

❖ Mesure de robustesse relative au makespan

La mesure proposée est une combinaison pondérée de trois mesures relatives aux différentes perturbations considérées.

La mesure relative à la panne machine est la suivante :

$$\delta 1_{C_{\max}} = \frac{\left(\sum_{O_{ij}} \sum_{m \in R_{ij}} \left[(1 + \omega_m) \gamma_{ijm} (S_{ij+1} - C_{ij}) + (1 - \gamma_{ijm}) \sum_{s=S_{ij}/\pi}^{S_{ij+1}/\pi} (1 - \varphi_{ms}) \pi \right] \right)}{N_o}$$

Avec N_o le nombre d'opérations.

Cette mesure possède deux composantes: la première est basée sur la marge libre des opérations et la deuxième est basée sur la disponibilité des machines alternatives. En effet, plus la marge entre deux opérations consécutives du même job est grande, plus le système peut absorber les pannes machines. Puisque dans chaque étage, nous disposons de machines parallèles, la robustesse s'accroît avec la disponibilité de ces ressources alternatives durant les fenêtres d'exécution des opérations

Nous notons que $0 \leq \omega_m \leq 1$ est un coefficient inversement proportionnel aux taux de pannes et temps de réparation de la machine m .

La mesure relative à l'absentéisme est la suivante :

$$\delta 2_{C_{\max}} =$$

$$\left(\sum_{O_{ij}} \left[\sum_e \left[\sum_{s=S_{ij}/\pi}^{S_{ij+1}/\pi} (1 + \omega_e) \left(\frac{z_{eij} \pi}{n_{ij} p_{ij}} \right) (S_{ij+1} - C_{ij}) + \sum_{s=S_{ij}/\pi}^{S_{ij+1}/\pi} \sum_{c \in C_e \cap C_{e'}} \frac{x_{ecs} z_{eij} \pi}{n_{ij}} (1 - \vartheta_{e's}) \right] \right] \right) / N_o$$

Avec n_{ij} le nombre total d'employés nécessaire pour l'opération O_{ij} (ce nombre dépend plutôt de l'étage dédié).

Cette mesure comporte deux composantes: une basée sur la marge libre des opérations et une autre basée sur la disponibilité des ressources humaines alternatives.

Nous notons que $0 \leq \omega_e \leq 1$ est un coefficient inversement proportionnel au taux d'absentéisme de l'employé e $(\overline{AB})_e$.

La mesure relative à la non-conformité est la suivante :

$$\delta 3_{C_{\max}} = \frac{\left(\sum_i (1 + \omega_i) \left[\sum_k F_{ik} \sum_{m \in R_k} \sum_{s=\frac{c_i n_i}{\pi}}^{C_{\max}} \tau_{ms} \pi \right] \right)}{n}$$

Nous rappelons que n désigne le nombre de jobs.

Cette mesure est basée sur la marge libre entre la dernière opération et le makespan et la disponibilité moyenne des ressources de retraitement possibles ($\tau_{ms} = 1$ si la machine m

peut fonctionner durant s ; 0 sinon, cette variable est fonction de la variable φ_{ms} et les variables $\vartheta_{e's}$).

Nous notons que $0 \leq \omega_i \leq 1$ est un coefficient inversement proportionnel au taux de non-conformité du job i $(\overline{NC})_i$.

En calculant le poids de chaque type de perturbation, nous

déduisons la mesure de robustesse totale $\delta_{C_{\max}}$:

$$\delta_{C_{\max}} = P_1 \delta 1_{C_{\max}} + P_2 \delta 2_{C_{\max}} + P_3 \delta 3_{C_{\max}}$$

Avec :

$$P_1 = \frac{\overline{(PN)}_k}{\overline{(PN)}_k + \overline{(AB)}_e + \overline{(NC)}_l}$$

$$P_2 = \frac{\overline{(AB)}_e}{\overline{(PN)}_k + \overline{(AB)}_e + \overline{(NC)}_l}$$

$$P_3 = \frac{\overline{(NC)}_l}{\overline{(PN)}_k + \overline{(AB)}_e + \overline{(NC)}_l}$$

❖ Mesure de robustesse relative au retard maximal

En suivant la même démarche adoptée pour le makespan, nous définissons trois composantes associées aux différents types de perturbations considérées :

La mesure relative à la panne machine est la suivante :

$$\delta 1_{L_{\max}} = \frac{\left(\sum_{O_{ij}} \sum_{m \in R_{ij}} \left[(1 + \omega_m) \gamma_{ijm} \max((d_{ij} - C_{ij}), 0) + (1 - \gamma_{ijm}) \sum_{s=\frac{S_{ij}}{\pi}}^{\frac{\max(C_{ij}, d_{ij})}{\pi}} (1 - \varphi_{ms}) \pi \right] \right)}{N_o}$$

Avec d_{ij} est la date de fin au plus tard de l'opération O_{ij} .

Cette mesure possède deux termes : le premier terme est basé sur la différence entre la date de fin au plus tard de l'opération et sa date de fin C_{ij} et le deuxième terme est basé sur la disponibilité des ressources machines alternatives.

La mesure relative à l'absentéisme est la suivante :

$$\delta 2_{L_{\max}} =$$

$$\left(\sum_{O_{ij}} \left[\sum_e \left[\sum_{s=S_{ij}/\pi}^{S_{ij+1}/\pi} (1 + \omega_e) \left(\frac{z_{eij} \pi}{n_{ij} p_{ij}} \right) \max(d_{ij} - C_{ij}, 0) + \sum_{s=S_{ij}/\pi}^{\max(d_{ij}, C_{ij})/\pi} \sum_{c \in C_e \cap C_{e'}} \frac{x_{ecs} z_{eij} \pi}{n_{ij}} (1 - \vartheta_{e's}) \right] \right] \right) / N_o$$

La mesure relative à la non-conformité est la suivante :

$$\delta 3_{L_{\max}} = \frac{\left(\sum_i (1 + \omega_i) \left[\sum_k F_{ik} \sum_{m \in R_k} \sum_{s=\frac{L_i + L_{\max}}{\pi}}^{C_{\max}} \tau_{ms} \pi \right] \right)}{n}$$

Nous déduisons alors la mesure de robustesse totale $\delta_{L_{\max}}$:

$$\delta_{L_{\max}} = P_1 \delta 1_{L_{\max}} + P_2 \delta 2_{L_{\max}} + P_3 \delta 3_{L_{\max}}$$

Afin de conclure sur les performances des mesures de robustesse proposées $\delta_{C_{\max}}$ et $\delta_{L_{\max}}$, nous comparons ces mesures à la mesure classique R définie seulement par la marge libre et bien utilisée dans la littérature pour des problèmes plus simples :

$$R = \frac{\sum_{O_{ij}} (S_{ij+1} - C_{ij})}{N_o}$$

Dans ce cadre, nous présentons la méthode suivante :

Pour chaque instance de notre problème, nous considérons les étapes suivantes :

Pour $i=1, N_r$

Etape 1

- Générer aléatoirement une solution S_r pour le problème ;
- Calculer le makespan $C_{max}(S_r)$ et le retard maximal $L_{max}(S_r)$.

Etape 2

- Déterminer $\delta_{C_{max}}(S_r)$, $\delta_{L_{max}}(S_r)$ et R.

Etape 3 Pour $j=1, N_p$

- Générer des perturbations aléatoires (pannes machines, articles non conformes, absences) ;
- Calculer le délai induit par les perturbations pour le C_{max} $\sigma_{C_{max}}^j(S_r)$ et le L_{max} $\sigma_{L_{max}}^j(S_r)$.

Fin

Etape 4

- Calculer la moyenne simple de $\sigma_{C_{max}}^{1 \leq j \leq p}(S_r)$ et $\sigma_{L_{max}}^{1 \leq j \leq p}(S_r)$.

Fin

Nous considérons 30 instances réelles de notre partenaire industriel ($N_r=40$, $N_p=100$), et nous appliquons pour chaque problème et chaque mesure de robustesse une régression linéaire entre la valeur réelle de la mesure et sa valeur estimée. Nous présentons, dans la figure 9, le coefficient de corrélation associé à chaque problème et chaque mesure.

Nous notons que le niveau de signification du test Anova (analyse de la variance) pour la régression linéaire est inférieur à 0.05 pour la plupart de problèmes considérés. Ce qui montre que la régression linéaire est adéquate pour toutes les mesures considérées.

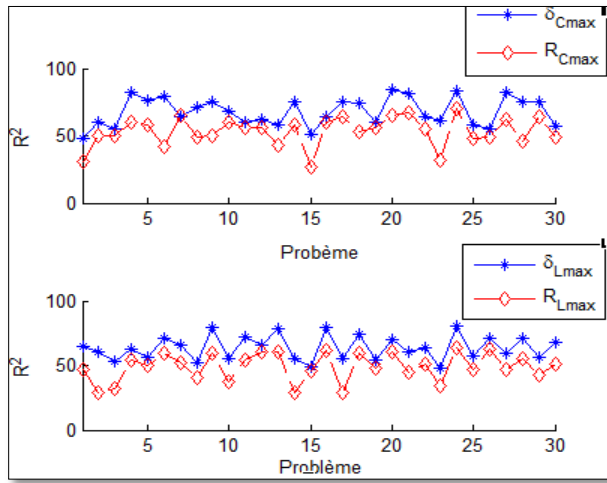


Figure 9. Résultats du coefficient R^2 pour chaque problème et chaque mesure de robustesse

Les résultats de la figure 9 montrent que la mesure $\delta_{C_{max}}$ est plus performante pour le makespan et la mesure $\delta_{L_{max}}$ est plus performante pour le retard maximal que la mesure classique R. (Plus la valeur R^2 est grande, plus la mesure est capable de prédire la valeur réelle de la robustesse).

6 MODELISATION DU PROBLEME

Dans ce paragraphe, nous proposons un modèle mathématique pour notre problème basé sur la programmation linéaire et la formulation logique.

Fonction objectif :

$$\min (C_{max}, L_{max}, -\delta_{C_{max}}, -\delta_{L_{max}})$$

Les contraintes du problème job shop sont :

$$C_{max} \geq C_{i n_i} \quad \forall i \in I, j \in [1, \dots, n_i] \quad (1)$$

$$L_{max} \geq \text{MAX}(C_{i n_i} - L_i, 0) \quad \forall i \in I, j \in [1, \dots, n_i] \quad (2)$$

$$C_{ij} = S_{ij} + \widetilde{p}_j \quad \forall i \in I, j \in [1, \dots, n_i] \quad (3)$$

$$S_{ij} \geq C_{i j-1} + TL_{min_{ij-1}} \quad \forall i \in I, j \in [1, \dots, n_i] \quad (4)$$

$$S_{ij} \leq C_{i j-1} + TL_{max_{ij-1}} \quad \forall i \in I, j \in [1, \dots, n_i] \quad (5)$$

$$S_{ij} \geq (C_{i'j'}) + ST_{k i' i} - N \left[\begin{array}{c} 2 - (\gamma_{ijm} + \gamma_{i'j'm}) \\ + U_{ij, i'j'} \end{array} \right] \quad (6)$$

$$S_{i'j'} \geq (C_{ij}) + ST_{k i i'} - N \left[\begin{array}{c} 2 - (\gamma_{ijm} + \gamma_{i'j'm}) \\ + (1 - U_{ij, i'j'}) \end{array} \right] \quad (6)$$

$\forall O_{ij}, O_{i'j'} \in O_k, m \in R_k$

$$\sum_{m \in R_{ij}} \gamma_{ijm} = 1 \quad \forall i \in I, j \in [1, \dots, n_i] \quad (7)$$

$$\varphi_{m,s} = \text{MIN} \left(1, \sum_i \sum_j \theta_{ijm s} \right) \quad (8)$$

$\forall k \in [1, \dots, r], m \in R_k, s \in S$

Les contraintes 4 et 5 illustrent les relations de précédence entre les opérations du même job, en intégrant les différentes contraintes de Time Lags.

Les contraintes 6 sont des contraintes disjonctives assurant que chaque machine traite au plus une seule opération à la fois.

Les contraintes 7 précisent que chaque opération doit être exécutée sur une seule machine.

Les contraintes 8 définissent la disponibilité de chaque machine.

Les contraintes du problème d'affectation des employés sont :

$$\sum_{c \in C_e} \sum_s x_{ecs} = 0, \quad \forall e \in E \quad (9)$$

$$\sum_{s \in T_e} \sum_{c \in C_e} x_{ecs} = 0, \quad \forall e \in E \quad (10)$$

$$z_{eij s} \leq \sum_{c \in C_e} b_{ck}, \quad \forall k \in [1, \dots, r], O_{ij} \in O_k, e \in E, s \in S \quad (11)$$

$$z_{eij s} \leq \sum_{c \in C_e} x_{ecs}, \quad \forall e \in E, s \in S, i \in I, j \in [1, \dots, n_i] \quad (12)$$

$$\sum_{c \in C_e} x_{ecs} \leq 1, \quad \forall e \in E, s \in S \quad (13)$$

$$\sum_e x_{ecs} = \beta_{cs}, \quad \forall s \in S, c \in [1, \dots, a] \quad (14)$$

$$\vartheta_{e s} = \text{MIN} \left(1, \sum_{c \in C_e} x_{e c s} \right) \quad \forall e \in E, s \in S \quad (15)$$

Les contraintes 9 précisent qu'un employé ne peut être affecté qu'avec une compétence pour laquelle, il est jugé qualifié ;

Les contraintes 10 assurent que chaque employé ne peut travailler que durant les tranches horaires pour lesquelles il est disponible ;

Les contraintes 11 traduisent qu'un employé ne peut être affecté à une tâche qu'avec une compétence requise ;

Chaque employé est de plus contraint, par les contraintes 13 à ne pas changer de poste durant la même tranche horaire.

Les contraintes 14 assurent l'équilibre entre la charge et la capacité en termes de compétences.

Les contraintes 15 définissent la disponibilité de chaque employé.

Les contraintes couplantes sont :

$$S_{ij} < (s + 1)\pi \cap C_{ij} > s\pi \cap \gamma_{ijm} = 1 \rightarrow \theta_{ijsm} = 1 \quad (16)$$

$$\forall i \in I, j \in \llbracket 1, \dots, n_i \rrbracket, m \in R_{ij}$$

$$S_{ij} \geq (s + 1)\pi \cup C_{ij} \leq s\pi \cup \gamma_{ijm} = 0 \rightarrow \theta_{ijsm} = 0 \quad (17)$$

$$\forall i \in I, j \in \llbracket 1, \dots, n_i \rrbracket, m \in R_{ij}$$

$$\beta_{cs} = \sum_k \sum_{m \in R_k} \sum_i \sum_j \theta_{ijsm} b_{ck} \quad \forall s \in S, c \in \llbracket 1, \dots, a \rrbracket \quad (18)$$

Ces contraintes permettent de quantifier la charge en compétences.

7 RESOLUTION DU PROBLEME

Pour la résolution de notre problème, nous proposons une heuristique à base de règles de priorité permettant d'avoir une solution réalisable respectant les différentes contraintes du problème et assurant un compromis entre les différents critères de performance considérés. Cette heuristique fait appel à d'autres heuristiques associées aux différentes décisions qui doivent être prises : choix de l'opération à placer, choix de la machine et choix des opérateurs.

7.1 Choix de l'opération

Afin de choisir l'opération à placer dans le planning parmi un ensemble d'opérations, nous combinons deux règles de priorité :

- **Règle de priorité FCFS**: cette règle consiste à prioriser les opérations qui peuvent être ordonnancées le plus tôt, afin d'avoir un ordonnancement décalé à gauche et minimiser alors le makespan. La date d'ordonnancement au plus tôt de chaque opération est calculée en considérant les différentes contraintes (disponibilité, précedence, « Time Lags » et « Setup Times »), avec une technique permettant de tester et remplir les différents trous générés par l'ordonnancement ;
- **Règle de priorité LST**: cette règle consiste à prioriser les opérations ayant des dates de début au plus tard les plus petites, afin de minimiser les retards. La date de début au plus tard est estimée par la méthode Monte Carlo permettant de considérer plusieurs scénarii aléatoires, en se basant sur les différentes incertitudes du problème (pannes machines, non-conformités et absentéisme).

Pour bénéficier de ces deux règles, nous adoptons la démarche suivante : Soit L1 la liste des opérations possédant une date de début au plus tôt inférieure à un seuil S1 (à fixer auparavant). Nous calculons les dates de début au plus tard des opérations de la liste L1 et nous retenons celles ayant une date inférieure à un seuil S2 (à fixer auparavant). Enfin, nous choisissons au hasard une opération parmi ces opérations.

La définition de deux seuils S1 et S2 a pour but de pouvoir paramétrer l'heuristique en fonction de l'objectif le plus visé (C_{\max}/L_{\max}). La sélection aléatoire de l'opération, a pour but de bien explorer l'espace de recherche.

En plus des contraintes de précedence classiques, certaines gammes opératoires imposent une durée maximale ou minimale entre certaines opérations consécutives du même job. Les techniques suivantes permettent la satisfaction de ces contraintes pour chaque opération O_{ij} à placer :

- Si l'opération O_{ij+1} doit assurer un temps d'attente minimal, nous ajoutons cette durée à sa date de disponibilité;
- Si l'opération O_{ij+1} doit assurer un temps d'attente maximal avec son prédécesseur O_{ij} , nous plaçons l'opération O_{ij+1} juste après O_{ij} , et nous décalons cette dernière à droite, si nécessaire jusqu'à que la contrainte soit respectée.

Cette idée est inspirée par les travaux de [Deppner, 2004].

7.2 Choix de la machine

Si l'étage concerné comporte plusieurs machines, dans ce cas une décision relative à la machine doit être prise.

Nous testons les heuristiques suivantes :

- ✓ **H1** : Choisir la machine permettant de placer l'opération au plus tôt (heuristique par défaut) ;
- ✓ **H7** : Choisir la machine au hasard ;
- ✓ **H8** : Choisir la machine ayant ω_m le plus élevée (taux de pannes et temps de réparation les plus faibles) et permettant de placer l'opération à $t \leq r_{ij} + S_3$ (S_3 seuil à fixer auparavant et r_{ij} la date de début au plus tôt de l'opération courante O_{ij}). Une faible valeur de S_3 permet de ne pas trop retarder l'opération.

7.3 Choix des opérateurs

Pour chaque activité ou encore compétence, nous testons les heuristiques suivantes :

- ✓ **H1** : Déterminer les opérateurs disponibles pour l'exécution de l'opération et choisir l'opérateur le moins polyvalent pour ne pas retreindre l'exécution des autres opérations, (Heuristique par défaut) ;
- ✓ **H9** : Déterminer les opérateurs disponibles pour l'exécution de l'opération et choisir l'opérateur au hasard ;
- ✓ **H10** : Déterminer les opérateurs disponibles pour l'exécution de l'opération et choisir l'opérateur ayant ω_e le plus élevé (taux d'absentéisme le plus faible).

Nous précisons que dans le cas de plusieurs choix possibles fournis par une heuristique, nous procédons par tirage aléatoire.

8 RESULTATS ET INTERPRETATIONS

Afin de conclure sur les performances de notre heuristique et étudier l'influence et l'apport de certaines techniques et règles de priorité, nous considérons 40 instances réelles d'une tannerie tunisienne (25*8 jobs et 20 employés), et précisément de son département finissage. En outre, nous choisissons un environnement fortement incertain.

Nous précisons que les durées opératoires, le taux d'absentéisme de chaque employé, ainsi que le taux de non-conformité de chaque commande sont estimés par la logique floue.

Nous ajoutons que chaque simulation est répétée 50 fois et nous considérons la moyenne pour chaque performance considérée.

Pour les interprétations de nos résultats, nous considérons pour le makespan et le retard maximal l'indicateur classique A défini pour chaque heuristique H_i et chaque instance j par :

$$A_{ij} = \frac{V_j(H_i)}{\forall_i (V_j(H_i))}$$

Avec $V = Cmax$ ou $Lmax$ ou δ_{Cmax} ou δ_{Lmax} et \forall l'opérateur MIN ou MAX (Selon le critère de performance considéré).

Et nous considérons les indicateurs suivants pour la robustesse :

$$Nb_{max_i} = \frac{\sum_{j=1}^{40} \partial_{ij}}{40}, \text{ avec } \partial_{ij} = 1, \text{ si } A_{ij} = 1; 0 \text{ sinon}$$

$$B_i = \text{MIN}_{1 \leq j \leq 40} (A_{ij})$$

$$C_i = (\text{MIN}_{1 \leq j \leq 40} (A_{ij}))$$

Il s'agit dans un premier temps de comparer notre technique basée sur la simulation Monte-Carlo H1 et la méthode déterministe H0, permettant de déterminer la date de début au plus tard de chaque opération.

Les résultats prouvent que la technique utilisée détériore légèrement le retard maximal (figure 10) et améliore la robustesse δ_{Lmax} (tableau 3). Nous retenons alors cette technique.

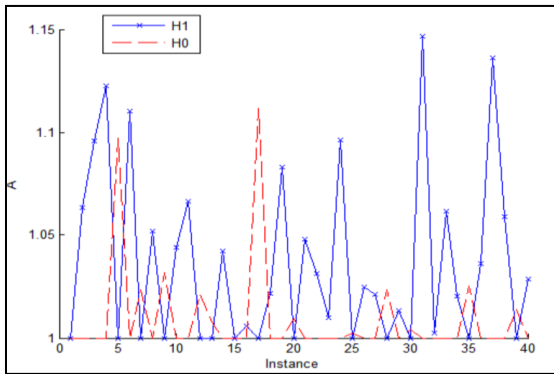


Figure 10. Résultats du retard maximal pour H1 et H0

Tableau 3. Résultats de la robustesse δ_{Lmax} pour H1 et H0

Heuristique	H1	H0
Nb_{max}	0.75	0.3
B	0.61	0.33
C	0.97	0.81

Dans un deuxième temps, il s'agit de comparer l'heuristique utilisée pour la sélection de l'opération avec d'autres règles de priorité. Dans ce cadre, nous testons les heuristiques suivantes :

H1 : l'heuristique proposée ;

H2 (SPT) : Choisit l'opération de plus faible temps de fabrication ;

H3 (FIFO) : Choisit la première opération à être disponible ;

H4 : Choisit l'opération dont le job a le taux de non-conformité le plus grand ;

H5 (EDD) : Choisit l'opération dont le job aura la plus faible durée de fabrication quand l'opération sera terminée ;

H6 (FOPNR) : Choisit l'opération pour laquelle il reste le plus d'opérations à suivre dans la gamme.

D'après les résultats, nous constatons que les heuristiques H1, H3 et H6 sont les plus performantes en termes de makespan et retard maximal (figures 11 et 12), et les heuristiques H1 et H2 sont les plus robustes. Nous affirmons alors que notre heuristique

permet d'avoir un bon compromis entre les différents objectifs considérés.

En outre, nous comparons les différentes heuristiques proposées pour le choix de la machine (H1, H7 et H8).

Les résultats montrent les performances de l'heuristique H1 en termes de makespan (figure 13) et retard maximal et les performances de l'heuristique H8 en termes de robustesse.

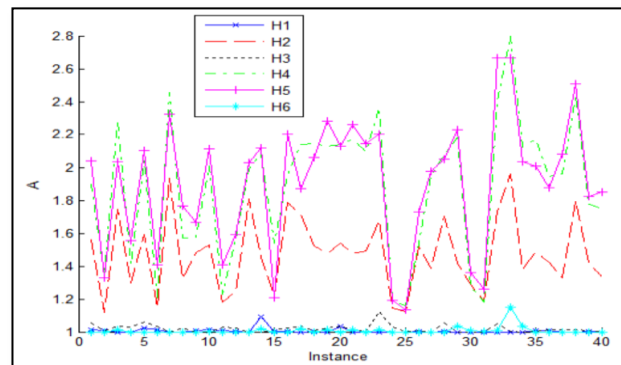


Figure 11. Résultats du makespan pour les différentes heuristiques.

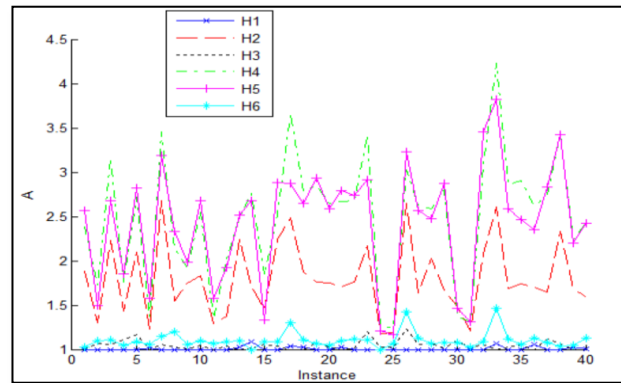


Figure 12. Résultats du retard maximal pour les différentes heuristiques

Tableau 4. Résultats de la robustesse δ_{Cmax} pour les différentes heuristiques

Heuristique	H1	H2	H3	H4	H5	H6
Nb_{max}	0.15	0.3	0.05	0.3	0.175	0.025
B	0.49	0.57	0.45	0.40	0.45	0.50
C	0.76	0.86	0.72	0.83	0.76	0.73

Tableau 5. Résultats de la robustesse δ_{Lmax} pour les différentes heuristiques

Heuristique	H1	H2	H3	H4	H5	H6
Nb_{max}	0.27	0.37	0.1	0.15	0.1	0.05
B	0.62	0.83	0.66	0.62	0.56	0.69
C	0.92	0.94	0.90	0.86	0.82	0.86

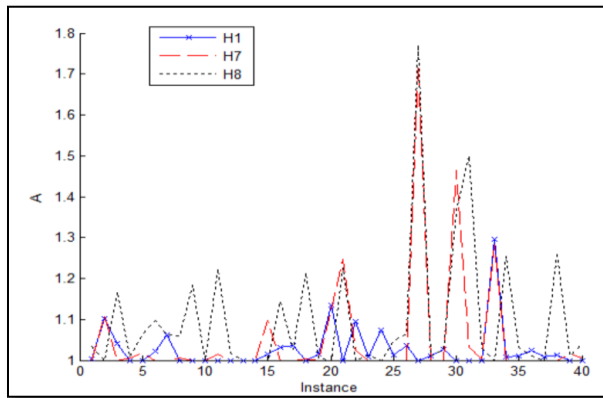


Figure 13. Résultats du makespan pour les différentes heuristiques

Une étude comparative entre les différentes heuristiques proposées pour la sélection des opérateurs (H1, H9 et H10) montre que H1 est plus performante pour le makespan et le retard maximal et H10 est plus robuste pour les critères makespan (Tableau 6) et retard maximal.

Tableau 6. Résultats de la robustesse δ_{cmax} pour les différentes heuristiques

Heuristique	H1	H9	H10
Nb _{max}	0.25	0.1	0.67
B	0.11	0.18	0.85
C	0.84	0.82	0.98

9 CONCLUSION

Confrontée aux différentes perturbations et incertitudes, les entreprises du secteur manufacturier cherchent de plus en plus à améliorer la fonction ordonnancement, pour bien gérer leur système de production. Dans ce contexte, ce papier traite un problème d'ordonnancement d'atelier sous différentes contraintes. Nous avons utilisé la logique floue pour l'estimation des incertitudes et nous avons défini deux mesures de robustesse associées aux critères makespan et retard maximal pour un tel problème. De plus, nous avons proposé une heuristique constructive à base de règles de priorité permettant d'avoir un compromis entre les performances et la robustesse de l'ordonnancement. Les résultats ont été testés sur des instances réelles d'une tannerie tunisienne.

Nous proposons dans nos futurs travaux de tester des algorithmes génétiques multi-objectifs et de proposer d'autres stratégies de ré-ordonnancement bien adaptées à notre problème afin de définir une approche proactive et réactive plus efficace et plus robuste.

10 REMERCIEMENTS

Les auteurs souhaitent remercier l'Agence Nationale de Promotion de la Recherche Scientifique pour le suivi et le financement du projet dans le cadre du dispositif PASRI/MOBIDOC.

11 REFERENCES

- Artigues, C., Gendreau, M., Rousseau, L.M., (2007) A flexible model and a hybrid exact method for integrated employee timetabling and production scheduling, *CORS/Optimization Days 2006 Joint Conference*, 3867, pp. 67-84.
- Alfares, K.H., Bailey, J.E., (1997) Integrated project task and manpower scheduling, *IIIE Transactions*, 29, pp. 711-717.
- Bean, J.C., Birge, J. R., Mittenthal, J., Noon, C.E., (1991). Match up scheduling with multiple resources release dates and disruptions. *Journal of Operations Research*, 39(3), 471-483.
- Chryssolouris, G., Subramaniam, V., (2001) Dynamic scheduling of manufacturing job shops using genetic algorithms. *Journal of Intelligent Manufacturing*, 12(3), 281-293.
- Daniels, R.L., Mazzola, J.B., (1994) Flow shop scheduling with resource flexibility, *Operations Research*, 42, pp. 504-522
- Deppner, F., (2004) Ordonnancement d'atelier avec contraintes temporelles entre opérations. Thèse de doctorat .LORIA Nancy.
- Dorn, J., Kerr, R. M., Thalhammer, G., (1995) Reactive scheduling: improving the robustness of schedules and restricting the effects of shop floor disturbances by fuzzy reasoning. *International Journal of Human Computer Studies*, 42, 687-704.
- Frihat, M., Sadfi, C., HadjAlouane, A., (2014) Optimization of Integrated Employee Timetabling and Hybrid Job Shop Scheduling under Time Lag Constraints, *2nd International Conference on Control, Decision and Information Technologies*, Metz, France.
- Guyon, O., Lemaire, P., Pinson, E., Rivreau, D., (2010) Solving an integrated Job-Shop problem with human resource constraints, *12th International Workshop on Project Management and Scheduling*, Tours France.
- Hooker, J.N., (2005) A hybrid method for planning and scheduling, *Constraints*, 10, pp. 385-401.
- Jensen, M.T., (2003) Generating Robust and Flexible Job Shop Schedules Using Genetic Algorithms, *IEEE Transactions on Evolutionary Computation*, 275-288.
- Leon, V.J., Wu, S.D., Store, R.H., (1994) Robustness measures and robust scheduling for job shops. *IIE Transactions*, 26(5): 32-43.
- MacCarthy, B. L., Liu, J., (1993) Addressing the gap in scheduling research: a review of optimization and heuristic methods in production scheduling. *International Journal of Production Research*, 31(1), 59-79.
- Mehta, S.V., Uzsoy, R., (1999) Predictable Scheduling of a Single Machine Subject to Breakdowns, *Int. J. Computer Integrated Manufacturing*, 15-38.
- Ouelhadj, D., Petrovic, S., (2008) A survey of dynamic scheduling in manufacturing systems. *Journal of Scheduling*, 417-431.
- Rajendran, C., Holthaus, O., (1999) A comparative study of dispatching rules in dynamic flow shops and job shops. *European Journal of Operational Research*, 116(1), 156-170.
- Rossi, A., Dini, G., (2000) Dynamic scheduling of FMS using a real-time genetic algorithm. *International Journal of Production Research*, 38(1), 1-20.
- Sabuncuoglu, I., Bayiz, M., (2000) Analysis of reactive scheduling problems in a job shop environment. *European Journal of Operational Research*, 126(3), 567-586.
- Wu, S.D., Storer, R.N., Chang, P., (1993) "One-Machine Rescheduling Heuristics with Efficiency and Stability as Criteria," *Computers Ops Res.*, 20.1.
- Yamamoto, M., Nof, S.Y., (1985) Scheduling/rescheduling in the manufacturing operating system environment. *International Journal of Production Research*, 23(4), 705-722.
- Zweben, M., Daun, B., Deale, M., (1994) Scheduling and rescheduling with iterative repair. In M. Zweben & M. S. Fox (Eds.), *Intelligent scheduling* (pp. 241-254). San Mateo: Morgan Kaufmann.