

Coopération homme-machine pour la mise en œuvre d'un ordonnancement de groupes

Guillaume PINOT, Nasser MEBARKI, Jean-Michel HOC

IRCCyN — UMR CNRS 6597
1 rue de la Noé, BP 92101
44321 Nantes Cedex 3
France

`prenom.nom@irccyn.ec-nantes.fr`
`http://cifa2008.ec-lille.fr`

Résumé—L'ordonnancement de groupes permet d'introduire une flexibilité séquentielle importante tout en garantissant une certaine qualité. Ceci fournit une grande liberté lors de l'exécution de l'ordonnancement, ce qui devrait permettre de s'adapter à l'état réel de l'atelier. Nous proposons dans cet article une nouvelle méthode pour favoriser l'utilisation de la flexibilité durant l'exécution de l'ordonnancement, et ce en favorisant la coopération avec l'humain. Cette nouvelle méthode de coopération homme-machine doit néanmoins être validée. Dans cet article, nous décrivons le protocole expérimental qui nous permettra l'évaluation de cette nouvelle méthode.

Mots-clés— Ordonnancement, coopération homme-machine, ordonnancement de groupes, incertitudes, aide à la décision.

I. INTRODUCTION

Dans les ateliers manufacturiers de type *job shop*, l'ordonnancement consiste à affecter à chaque opération de chaque ordre (ou travail) du plan de fabrication, les ressources nécessaires à sa réalisation en respectant les dates de début au plus tôt et les dates d'échéances définies par la planification. Le problème du *job-shop* est modélisé comme un problème d'optimisation combinatoire composé de ressources, d'opérations et de contraintes. Les travaux (J_i) sont composés de plusieurs opérations ($O_{i,j}$). Les opérations sont exécutées sur les ressources (M_i , aussi appelées machines) pendant un temps d'exécution $p_{i,j}$. La date de début de l'opération $O_{i,j}$ est notée $t_{i,j}$, et sa date de fin est notée $C_{i,j}$. C_i correspond à la date de fin du travail J_i (cette date est en fait la date de fin de la dernière opération du travail). Dans un même travail, les opérations doivent être exécutées dans l'ordre lexicographique, c'est-à-dire que $t_{i,j+1} \geq C_{i,j}$. Un travail possède une date de livraison noté d_i . Si un travail finit après sa date de livraison ($C_i \geq d_i$) alors ce travail est en retard. Une ressource ne peut exécuter qu'une opération à la fois.

Généralement, une fonction objectif régulière, qui est une fonction monotone des $C_{i,j}$, est utilisé pour l'évaluation d'une solution d'un problème de *job shop*. Le but est de trouver une solution minimisant cette fonction objectif. Le *makespan*, noté C_{\max} , calculé $\max C_{i,j}$, qui correspond au temps total de l'exécution de l'ordonnancement, est un objectif régulier classique. Un autre objectif régulier est le retard algébrique maximum, noté L_{\max} , calculé $\max(C_i - d_i)$.

Dans cette approche, le problème du *job shop* est donc résolu comme un problème déterministe où les données sont considérées comme précises et fiables. En pratique, les problèmes d'ordonnancement sont souvent soumis à une incertitude sur les données : incertitude sur les durées et les dates de disponibilité des opérations, opérations urgentes ou imprévues à réaliser, disponibilité incertaine des ressources [1]. Pour pallier ces incertitudes, une méthode d'ordonnancement reposant sur l'introduction de flexibilité séquentielle dans les opérations a été proposée [2]. Ceci est réalisé en déterminant, plutôt qu'un ordonnancement unique, un ordonnancement de groupes, c'est-à-dire en définissant une séquence de groupes d'opérations permutable sur chaque machine. Il reste donc un certain nombre de décisions qui devront être prises dans l'atelier durant l'exécution de l'ordonnancement, cette phase étant appelée la phase réactive de l'ordonnancement.

Mais ces décisions ne sont pas triviales à réaliser. En effet, la machine ne possède que des informations incomplètes de l'atelier : modélisation simplifiée, incertitudes non prévisibles. L'humain peut apporter une vision plus précise, plus exacte de l'atelier. Il peut aussi anticiper des problèmes non encore visibles, c'est pourquoi il est important de favoriser une coopération entre l'humain et la machine pour combiner les avantages de chacun pour réaliser des décisions pertinentes.

Après une description de l'ordonnancement de groupes, nous proposons une nouvelle phase réactive permettant de faciliter la coopération homme-machine. Nous décrivons ensuite le protocole expérimental permettant d'évaluer cette nouvelle méthode.

II. LES GROUPES D'OPÉRATIONS PERMUTABLES

On trouve dans [2] une première présentation des groupes d'opérations permutable. Le but de cette méthode est de fournir de la flexibilité séquentielle pendant l'exécution de l'ordonnancement tout en garantissant une qualité minimale correspondant au pire des cas. Cette méthode est largement étudiée depuis vingt ans, notamment dans [2], [3], [4], [5]. Ce dernier article fournit une description théorique de la méthode.

Cette méthode utilise une approche prédictive réactive

[6], c'est-à-dire que cette méthode se décompose en deux parties :

- La phase prédictive, dans laquelle une solution au problème est calculé avant son exécution. Généralement, la solution proposée est un ensemble d'ordonnements.
- La phase réactive, dans laquelle un ordonnancement est exécutée dans l'atelier. Cette phase se base sur la solution proposée durant la phase prédictive, et prend en compte l'état réel de l'atelier. Ainsi, l'ordonnement est choisi en prenant en compte les incertitudes.

L'avantage d'une telle décomposition est qu'elle combine un processus d'optimisation durant la phase prédictive, processus permettant d'obtenir un ordonnancement de bonne qualité, et une méthode permettant de s'adapter au problème réel malgré les incertitudes durant la phase réactive.

A. Définition de l'ordonnement de groupes

Un groupe d'opérations permutable est un ensemble d'opérations à exécuter sur une certaine ressource dans un ordre arbitraire. Il est noté G_k . Le groupe contenant l'opération $O_{i,j}$ est noté $g(i,j)$.

Un ordonnancement de groupes est une liste ordonnée de groupes (d'opérations permutable) sur chaque machine, à exécuter dans cet ordre. Sur une machine, le groupe après (resp. avant) G_k est noté G_k^+ (resp. G_k^-) et est appelé le groupe successeur (resp. prédécesseur) de G_k . En utilisant les mêmes conventions, le groupe après (resp. avant) $g(i,j)$ est noté $g^+(i,j)$ (resp. $g^-(i,j)$).

Un ordonnancement de groupes est réalisable si chaque permutation sur les opérations d'un même groupe donne un ordonnancement réalisable (c'est-à-dire un ordonnancement qui ne viole pas de contraintes). Ainsi, un ordonnancement de groupes décrit un ensemble d'ordonnement valide, sans les énumérer.

Les descripteurs de qualité d'un ordonnancement de groupes sont les mêmes que pour l'ordonnement classique. Ainsi, [5] définit que la qualité d'un ordonnancement de groupes correspond à la pire qualité présente dans l'ensemble des ordonnancements semi-actifs.

B. Exemple

Pour illustrer ces définitions, étudions un exemple. La figure 2 représente un ordonnancement de groupes réalisable résolvant le problème de la figure 1. Il est constitué de sept groupes : deux groupes de deux opérations et cinq groupes d'une opération. Cet ordonnancement de groupes définit quatre ordonnancements semi-actifs¹ correspondants (figure 3). On remarquera que ces ordonnancements ont des qualités différentes : par exemple, pour le *makespan*, la qualité dans le meilleur des cas est $C_{\max} = 10$ et la qualité dans le pire des cas est $C_{\max} = 17$.

La phase réactive consiste à choisir un ordonnancement décrit par l'ordonnement de groupes à exécuter. Il peut être vu comme un ensemble de décisions : chaque décision consiste à choisir une opération dans le groupe à exécuter lorsque ce groupe comporte plus d'une opération. Ainsi,

¹Un ordonnancement semi-actif est un ordonnancement où l'on exécute les opérations au plus tôt. Pour une séquence d'opérations, il existe un unique ordonnancement semi-actif.

i	j	$M_{i,j}$	$p_{i,j}$
1	1	1	3
1	2	2	3
1	3	3	3
2	1	2	4
2	2	3	3
2	3	1	1
3	1	3	2
3	2	1	2
3	3	2	2

Fig. 1. Un problème de *job shop*

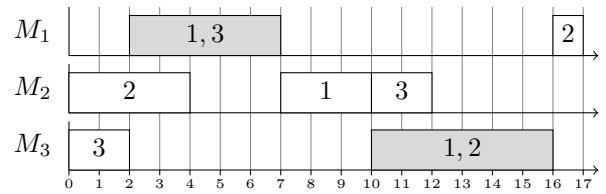


Fig. 2. Un ordonnancement de groupes résolvant le problème décrit dans la figure 1

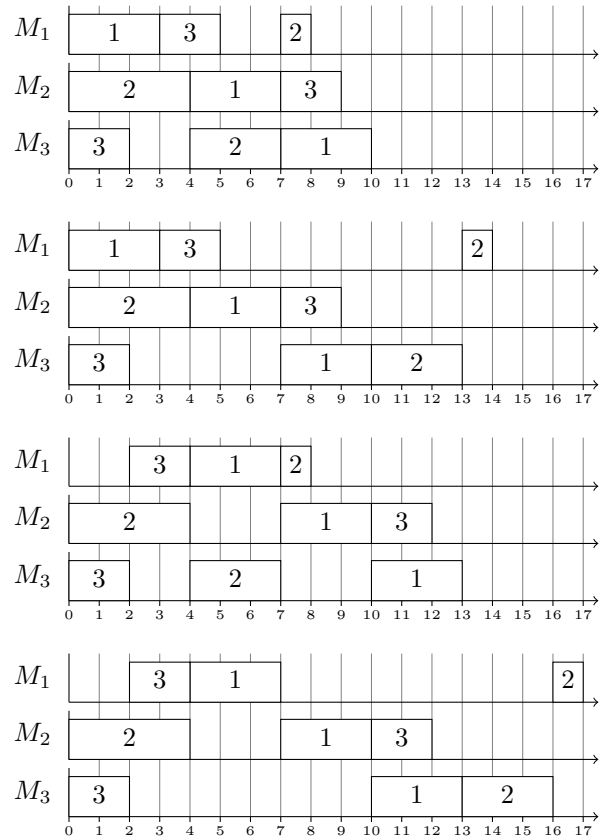


Fig. 3. Ordonnements semi-actifs décrits par la figure 2

pour l'ordonnancement de groupes de la figure 2, deux décisions seront à prendre : Sur M_1 , au début de l'ordonnancement, il faudra choisir si l'on exécute d'abord l'opération $O_{1,1}$ ou $O_{3,2}$. Supposons que l'on choisisse d'exécuter $O_{1,1}$ en premier. Sur M_3 , après avoir exécuté $O_{3,1}$, nous avons une décision à effectuer : exécuter en premier $O_{1,3}$ ou $O_{2,2}$. Si nous décidons d'exécuter $O_{2,2}$ en premier, nous avons alors réalisé le premier ordonnancement de la figure 3.

C. Évaluation d'un ordonnancement de groupes

L'ordonnancement de groupes possède une propriété intéressante : le calcul de la qualité d'un ordonnancement de groupes dans le pire des cas se fait en temps polynomial (voir [5] pour la description de l'algorithme). Ainsi, il est possible de calculer sans problème la qualité dans le pire des cas, même pour les ordonnancements de groupes de très grande taille. Par conséquent, cette méthode peut être utilisée pour calculer la qualité dans le pire des cas en temps réel pendant l'exécution de l'ordonnancement. Grâce à cette propriété, il est possible de surveiller la qualité d'un ordonnancement de groupes dans un système d'aide à la décision de manière dynamique.

Basé sur des méthodes de calcul similaires, [7] propose une adaptation de la marge libre à l'ordonnancement de groupes : la marge libre séquentielle. La marge libre séquentielle d'une opération représente le retard maximum de l'opération par rapport à son exécution au plus tôt assurant que tous les ordonnancements décrits par l'ordonnancement de groupes soient sans retard. Exécuter l'opération possédant la plus grande marge libre séquentielle augmente les marges des autres opérations du groupe, ce qui permet de préserver au maximum la flexibilité temporelle. Lors de son utilisation en ligne, plusieurs cas peuvent se produire :

- Si toutes les marges libres séquentielles du groupe en cours d'exécution sont positives, quelle que soit l'opération choisie, les ordonnancements décrits seront sans retard.
- Si une marge libre séquentielle est négative, tous les ordonnancements décrits ne sont pas sans retard, notamment certains commençant par l'opération possédant la marge négative. Dans ce cas, il est conseillé d'exécuter des opérations possédant de grandes marges pour augmenter la marge négative en espérant la rendre positive par la suite. Si quelques marges (mais pas toutes) sont négatives, la situation est similaire.
- Si toutes les marges sont négatives, quelles que soient les décisions de l'opérateur sur ce groupe, il existera des ordonnancements décrits avec des retards. Il est par contre tout à fait possible qu'il existe des ordonnancements décrits sans retard.

Cette marge est un outil précieux lors de l'exécution de l'ordonnancement de groupes. Par contre, elle ne permet pas de savoir s'il existe ou non un ordonnancement sans retard lorsqu'il y a une ou plusieurs marges séquentielles libres négatives.

Un algorithme permettant de calculer la qualité dans le meilleur des cas permettrait de pallier ce problème : si le retard algébrique maximum dans le meilleur des cas est négatif ou nul, cela signifie qu'il existe au moins un or-

donnement décrit sans retard. Dans [8], [9], [10], nous réalisons une étude du meilleur des cas dans un ordonnancement de groupes. [8] propose un algorithme polynomial calculant une borne inférieure du plus petit $C_{i,j}$ pour chaque opération dans un ordonnancement de groupes. Cet outil donne une borne inférieure pour chaque objectif régulier dans un ordonnancement de groupes. Une borne inférieure améliorée pour le *makespan* est également proposée. Les performances de cette dernière borne sont très bonnes. [9] décrit différentes heuristiques pour l'ordonnancement de groupes. Une heuristique donne une borne supérieure de la qualité dans le meilleur des cas d'un ordonnancement de groupes, ce qui apporte une information complémentaire à celle donnée par les bornes inférieures. Dans cet article, le *shifting bottleneck* [11] est adaptée à l'ordonnancement de groupes. Cette adaptation, conçue pour optimiser le *makespan*, est très efficace : en moyenne, l'écart à l'optimal de l'ordonnancement trouvé est de 1,5%. [10] propose une méthode exacte pour trouver le meilleur des cas de n'importe quel objectif régulier dans un ordonnancement de groupes. Des expérimentations sont faites sur le *makespan*. Sur les 40 instances de *job shop* testées, 15 sont résolues en moins d'une seconde, 32 en moins d'une minute, 34 en moins d'une heure et 36 en moins d'une journée. 4 instances restent non résolues au bout d'une semaine de calcul.

D. Flexibilité et robustesse

L'ordonnancement de groupes permet donc de décrire un ensemble d'ordonnancement de manière implicite (c'est-à-dire sans énumérer les ordonnancements) tout en garantissant une certaine qualité. Lors de l'exécution d'un tel ordonnancement, il est possible de choisir la séquence d'opérations adapté à l'état réel de l'atelier.

De plus, la flexibilité ajoutée à l'ordonnancement devrait permettre d'absorber des incertitudes. Trois études ont essayé de vérifier cette propriété.

[4] a étudié l'impact de la perturbation des temps d'exécution sur la somme pondérée des retards par rapport à des heuristiques statiques et dynamiques. Quand les temps d'exécutions ne sont pas trop perturbés, l'ordonnancement de groupes donne de meilleures performances.

[12] a étudié l'impact de la perturbation des temps d'exécutions, des dates de livraison et des dates de disponibilité sur un problème à une machine, et compare l'ordonnancement de groupes à une heuristique statique. En moyenne, les performances sont meilleures avec l'ordonnancement de groupes.

[13] a étudié l'impact sur un système réel de la non-modélisation de temps de transport entre deux opérations dans l'ordonnancement de groupes. La méthode montre de bonnes performances, même lorsque les temps de transport du système réel sont comparables aux temps d'exécution.

La flexibilité apportée par l'ordonnancement de groupes semble donc bien donner une certaine robustesse à l'ordonnancement. Mais encore faut-il utiliser correctement cette flexibilité durant la phase réactive.

III. ANALYSE DE LA PHASE RÉACTIVE D'ORABAID

ORABAID (ORdonnancement d'Atelier Basé sur l'Aide à la Décision) est une méthode d'ordonnancement basé

sur l'ordonnancement de groupes. Cette méthode est implémentée dans le progiciel ORDO [14].

La phase réactive d'ORABAID se base sur la marge libre séquentielle. La machine donne à l'opérateur la marge libre séquentielle de chaque opération du groupe à exécuter. La machine suggère d'exécuter l'opération possédant la plus grande marge. L'opérateur peut néanmoins décider d'exécuter une autre opération, qu'elle fasse partie du groupe en cours d'exécution ou pas. La machine l'informe sur les répercussions de son choix sur les marges des opérations, et il valide ou non en dernier ressort.

Ainsi, cette phase réactive permet un contrôle total du décideur sur l'ordonnancement tout en garantissant une certaine qualité. De plus, en cas de retard possible, le problème est détecté avant qu'il ne soit trop tard, ce qui permet au décideur d'anticiper et donc de réagir plus tôt.

Le principal inconvénient de cette phase réactive est qu'elle laisse le décideur choisir l'opération avec très peu d'aide : soit le décideur choisit l'opération proposée par le système, soit il doit évaluer les conséquences des autres décisions une à une, sans avoir d'autres indicateurs que la marge libre séquentielle. Le principal risque est que le système soit sous-utilisé : devant la complexité de la tâche, le décideur suivra la proposition donnée par la machine sans se poser de question comme le remarque Cegarra dans [15]. La flexibilité offerte par l'ordonnancement de groupes serait alors sous-utilisée et l'humain n'interviendrait que très peu dans la réalisation de l'ordonnancement.

IV. PROPOSITIONS POUR LA PHASE RÉACTIVE

De nombreuses expérimentations ([16], [15]) montrent que les performances obtenues en combinant l'humain avec la machine sont meilleures que celles obtenues en utilisant uniquement l'un ou l'autre. Il est donc utile de favoriser cette coopération pour améliorer l'ordonnancement.

Pour cela, nous proposons une approche multicritères pour l'utilisation en ligne d'un ordonnancement de groupes. Un certain nombre d'indicateurs sont proposés à l'opérateur pour l'aider dans son choix. Nous pouvons remarquer que, dans ses perspectives ([12]), Esswein indique qu'utiliser une approche multicritères pour la phase réactive pourrait permettre d'améliorer la méthode ORABAID. Cette modélisation multicritères semble donc être adaptée.

Cette approche multicritères est nécessaire car les décisions à prendre dans un atelier impliquent différents critères comme la flexibilité et la performance, leur importance respective étant dépendante du contexte. En effet, on peut imaginer que lorsque l'atelier est soumis à de nombreuses incertitudes, il est important de maximiser sa flexibilité afin d'obtenir une solution plus robuste. Par contre, lorsque l'atelier subit moins d'aléas, il semble judicieux de se focaliser sur les performances de l'atelier.

Plutôt que de proposer un unique indicateur à l'opérateur comme le fait ORABAID, notre système proposera plusieurs indicateurs. Ces indicateurs devront représenter les connaissances de la machine sur l'ordonnancement de groupes en cours d'exécution, notamment ce pour quoi la machine est adaptée : réaliser des calculs. Chacun de ces indicateurs peut représenter soit un critère de décision, soit il peut donner des informations

complémentaires sur les opérations à l'opérateur.

Ainsi, l'opérateur reçoit des informations qu'il ne pourrait pas connaître seul, et peut les combiner à ses propres compétences et connaissances comme sa gestion des compromis et sa connaissance de l'état de l'atelier. L'opérateur peut ainsi faire son choix de manière éclairée, sans que la machine ne lui dicte directement une réponse. Il serait ainsi plus impliqué dans le processus de décision et la coopération homme-machine serait plus efficace.

Les indicateurs calculés propres à l'ordonnancement de groupes que nous proposons sont :

- la marge libre séquentielle d'une opération, également utilisée par ORABAID ;
- la qualité dans le pire des cas pour une opération, qui donne la pire qualité prédite de l'ordonnancement si la décision de l'opérateur est d'exécuter cette opération ;
- la qualité dans le meilleur des cas pour une opération, qui donne la meilleure qualité prédite de l'ordonnancement si la décision de l'opérateur est d'exécuter cette opération.

Pour la qualité dans le meilleur et dans le pire des cas, différents objectifs réguliers peuvent être utilisés, notamment le *makespan* et le retard algébrique maximum. Ces indicateurs sont directement reliés à des critères tels que la flexibilité pour la marge libre séquentielle, le respect des dates de livraisons pour le retard algébrique maximum dans le pire et le meilleur des cas.

Des indicateurs plus simples seront également proposés à l'opérateur :

- les durées opératoires des opérations ($p_{i,j}$) ;
- les temps d'exécution restant sur les travaux ;
- le nombre d'opérations restant dans le travail ;
- les dates de livraison des travaux.

Ces indicateurs donnent des informations complémentaires directement liées aux opérations, permettant à l'opérateur de mieux les appréhender. Par exemple, les durées opératoires des opérations donnent des informations sur le travail qu'il aura à réaliser.

Nous pensons que cette nouvelle formulation de la phase réactive de l'ordonnancement de groupes devrait favoriser la coopération homme-machine et ainsi améliorer les performances de l'ordonnancement. Nous proposons dans la suite de cet article un protocole expérimental permettant de vérifier ces hypothèses.

V. EXPÉRIMENTATION

A. But

Pour évaluer cette coopération homme-machine, il est nécessaire d'utiliser une démarche empirique. En effet, aucune donnée théorique ne peut valider ou infirmer une interaction homme-machine. Il est nécessaire de tester cette interaction par une expérience.

Le but de cette expérimentation est de comparer la méthode réactive d'ORABAID avec notre nouvelle méthode réactive. Comme la différence entre les deux méthodes peut se résumer aux indicateurs proposés à l'opérateur, l'expérience étudiera l'impact de ces indicateurs sur la coopération et sur les performances.

Nous espérons que notre nouvelle méthode rendra l'opérateur plus actif [17] dans le système homme-machine,

c'est-à-dire qu'il participera directement à la réalisation de l'ordonnancement avec l'aide de la machine et non qu'il utilisera la machine sans rien apporter au système (dans ce cas, l'opérateur sera passif). Il est bien sûr possible que l'opérateur soit plus ou moins actif, c'est pourquoi nous parlerons du niveau d'activité. Cette activité de l'opérateur devrait rendre le système homme-machine plus efficace et ainsi les performances de l'ordonnancement devraient être meilleures. L'activité de l'opérateur ajouterait également autre chose : l'opérateur serait plus attentif à ce qui se passe dans l'atelier, ce qui lui permettrait de mieux assumer ses responsabilités sur le bon fonctionnement de l'atelier.

B. Plan de l'expérience

Dans cette section, nous allons lister les facteurs de l'expérimentation, qui correspondent aux données que nous maîtrisons ; et les variables dépendantes, qui correspondent aux données donnant les résultats de l'expérimentation.

Le premier facteur de cette expérimentation sera le type d'indicateurs proposé. Suivant que nous proposons uniquement la marge libre séquentielle ou d'autres indicateurs, nous obtenons la méthode réactive d'ORABAID ou notre nouvelle méthode.

Le second facteur sera le niveau d'activité de l'opérateur. Nous allons donc contrôler le niveau d'activité durant l'expérience.

Nous utiliserons comme variable dépendante les performances de l'ordonnancement élaboré grâce au système homme-machine. Les objectifs classiques de l'ordonnancement, notamment le *makespan* et le retard algébrique maximum seront utilisés.

Une autre variable dépendante sera l'utilisation des différents indicateurs par l'opérateur. Cette variable nous permettra d'identifier les indicateurs les plus utilisés par l'opérateur, et donc les indicateurs les plus pertinents pour le bon fonctionnement du système homme-machine.

C. Méthode expérimentale

L'architecture logicielle sera composée de deux programmes : un programme simulant l'atelier de production appelé simulateur, et un programme de supervision surveillant le programme de simulation et calculant les différents indicateurs appelé superviseur. Le superviseur donnera les indicateurs à l'opérateur. Il fait partie du système homme-machine étudié. Le simulateur possède également des interactions avec l'opérateur : l'opérateur lui donnera sa décision (l'opération à exécuter). Ce programme ne fait pas partie du système homme-machine étudié : il simule l'atelier de production.

La simulation ne s'effectue pas en temps réel : le simulateur avance dans le temps le plus rapidement possible jusqu'à ce qu'il arrive à un événement nécessitant une décision de l'opérateur. La simulation se met alors en pause et attend la décision de l'opérateur. L'opérateur peut alors consulter le superviseur (et observer l'état de l'atelier sur le simulateur) pour choisir l'opération à exécuter. Une fois cette opération choisie, l'opérateur indique au simulateur sa décision. Cette procédure continue jusqu'à la fin de l'ordonnancement, moment où on peut obtenir les performances de l'ordonnancement.

Pour obtenir des résultats plus réalistes, des incertitudes seront présentes lors de l'exécution de l'ordonnancement : les données traitées par le simulateur (durées opératoires, temps de transfert des produits d'une machine à une autre) seront plus ou moins différentes des informations du superviseur. Ainsi, nous pourrions tester différents scénarios, comme par exemple :

- Il n'y a pas d'incertitudes (les données du simulateur et du superviseur sont équivalentes) : Le superviseur donnerait alors des prédictions toujours exactes.
- De petites incertitudes sont présentes tout au long de l'ordonnancement (par exemple, les durées opératoires sont légèrement différentes entre les données du simulateur et du superviseur ou des temps de transports entre les opérations sont présents dans la simulation, mais le superviseur ne les connaît pas) : Le superviseur donnerait alors des prédictions approximatives.
- Une incertitude localisée, mais de grande ampleur (par exemple, une durée opératoire pour une opération donnée sur le superviseur est 20 fois plus longue que dans les données du simulateur) : Nous serions alors dans le cas d'un retard causé par un événement isolé, Le superviseur ne prédirait pas ce retard, mais s'adapterait après l'incertitude à l'état du simulateur. Par la suite, le simulateur donnerait des prédictions exactes, mais aux performances bien moindres que prévu au départ.

Lors de ces expériences, nous allons enregistrer un certain nombre de données qui serviront au dépouillement :

- Pour chaque décision, nous enregistrerons une date de début et une date de fin. Cette information nous permettra tout d'abord de déterminer le temps passé par l'opérateur sur chaque décision. Elle nous permettra aussi de placer les différentes actions enregistrées dans le temps pour une décision donnée.
- Pour suivre l'utilisation des indicateurs par l'opérateur, notre version du superviseur ne montrera pas directement les indicateurs. Un tableau vide mettant les opérations en ordonnée et les indicateurs en abscisse sera proposé à l'opérateur. Pour accéder à ces données, il devra cliquer sur la case vide correspondant à la donnée qu'il veut voir pour que celle-ci s'affiche. Ainsi, nous pouvons enregistrer les dates des différents accès aux indicateurs. Nous pouvons ainsi savoir quels indicateurs sont utilisés pour une décision, ainsi que le moment où ils sont utilisés dans le processus de décision.
- Pour évaluer la coopération du système homme-machine, nous effectuerons un enregistrement sonore de l'opérateur durant l'expérimentation. L'opérateur aura pour consigne de verbaliser ce qu'il fait, c'est-à-dire de dire tout haut ce qui lui passe spontanément par la tête. Ainsi, nous pourrions suivre le processus de coopération avec la machine grâce à ces enregistrements.

Comme nous l'avons dit dans la section précédente, le niveau d'activité de l'opérateur sera un facteur, et nous avons donc besoin de la contrôler. Ceci permettra de voir l'impact du niveau d'activité de l'opérateur dans ce système homme-machine, et ainsi déterminer s'il est nécessaire de favoriser cette activité. Pour rendre actif un opérateur passif, après que l'opérateur ait pris sa décision, la machine suggérera

une autre solution. Cette suggestion forcera l'opérateur à justifier son choix, où à justifier le choix de la machine, et ainsi le rendra actif dans le processus de décision.

Les participants à cette expérience seront des étudiants en troisième année d'étude supérieure en gestion de la production. Bien que ce ne soient pas des professionnels de l'ordonnancement, leur formation leur donne une connaissance des ateliers de productions et de l'ordonnancement. Le premier échantillon sera composé d'une dizaine de personnes. Si les résultats ne sont pas suffisamment fiables, il sera par la suite possible d'effectuer d'autres expériences pour rendre les résultats plus précis. Pour valider les résultats, l'analyse se basera sur des méthodes statistiques permettant de connaître les intervalles de confiances correspondant aux résultats.

D. Résultats

Le résultat le plus important sera l'influence du niveau d'activité sur les performances et la coopération. En effet, si l'activité permet d'obtenir de meilleures performances, proposer une méthode favorisant cette activité est primordiale. C'est l'hypothèse que nous favorisons. Bien qu'il soit peu probable que l'activité diminue la coopération, il est possible qu'elle diminue les performances. Dans ce cas, cela révélerait que la machine seule serait plus efficace que le système homme-machine. Il serait dans ce cas plus judicieux de proposer une méthode entièrement automatisé, plutôt qu'un système homme-machine. Par rapport à la littérature existante et à nos intuitions, cette hypothèse nous semble peu probable. Il reste une dernière possibilité : l'activité n'influe pas sur les performances et/ou la coopération. Dans ce cas, il n'est pas primordial de favoriser cette activité, bien qu'elle puisse avoir d'autres effets positifs, notamment sur l'attention de l'opérateur.

Un autre résultat important est celui pour lequel cette expérimentation sera réalisé : la comparaison de la méthode ORABAID et de notre nouvelle méthode vis-à-vis des performances et de la coopération. Nous pensons que les performances et la coopération seront meilleures avec notre nouvelle méthode. Si cette hypothèse est confirmée, l'apport de notre nouvelle méthode sera alors confirmé. Si, par contre, la méthode ORABAID surpasse notre nouvelle méthode, nous devons analyser pourquoi, pour proposer par la suite une autre méthode mieux adaptée à ce problème.

Les autres résultats que nous apporterons cette expérience nous permettront de mieux comprendre le fonctionnement du système homme-machine pour la méthode ORABAID ainsi que pour notre nouvelle méthode. Grâce à ces informations, nous pourrons ensuite améliorer le système homme-machine, quels que soient les résultats précédents. Ces résultats sont :

- Les indicateurs utilisés par l'utilisateur, et leur contexte. Ceci nous permettra de sélectionner les indicateurs les plus pertinents pour l'utilisateur, et de les rendre plus accessibles. Suivant les informations utilisées dans les indicateurs (signe, ordre de grandeur, etc.), des outils graphiques pourront être utilisés pour rendre leur lecture plus simple.
- Le niveau de confiance accordé par l'opérateur dans le superviseur en fonction des incertitudes. Cette infor-

mation nous permet d'évaluer la qualité ressentie des algorithmes par l'utilisateur, et éventuellement nous permettrait de cibler les algorithmes à améliorer.

- Les remarques de l'opérateur sur le superviseur. Ces remarques nous permettraient d'améliorer le superviseur.

Ces résultats nous permettront donc d'analyser la phase réactive de l'ordonnancement de groupes, que ce soit la méthode ORABAID ou notre nouvelle méthode. Grâce à cette analyse, nous pourrons ensuite améliorer la phase réactive pour faciliter la coopération homme-machine et rendre le système homme-machine plus performant.

VI. CONCLUSION

Dans cet article, nous avons présenté l'ordonnancement de groupes, et plus particulièrement sa phase réactive. Nous avons ensuite proposé une nouvelle méthode pour cette phase réactive, avec comme but de favoriser la coopération homme-machine. Pour évaluer la phase réactive de l'ordonnancement de groupes, nous proposons une démarche empirique. Les implications des résultats de cette future étude sont ensuite discutées.

Par la suite, nous allons réaliser cette expérimentation, analyser les résultats et prendre en compte ces résultats pour une nouvelle version de la phase réactive de l'ordonnancement de groupes.

RÉFÉRENCES

- [1] Christian Artigues. Optimisation et robustesse en ordonnancement sous contraintes de ressources, 2004. Habilitation à diriger des recherches, Université d'Avignon.
- [2] Jacques Erschler et François Roubellat. An approach for real time scheduling for activities with time and resource constraints. R. Slowinski et J. Weglarz, editors, *Advances in project scheduling*. Elsevier, 1989.
- [3] Jean-Charles Billaut et François Roubellat. A new method for workshop real-time scheduling. *International Journal of Production Research*, 34(6) :1555–1579, 1996.
- [4] S. David Wu, Eui-Seok Byeon, et Robert H. Storer. A graph-theoretic decomposition of the job shop scheduling problem to achieve scheduling robustness. *Operations Research*, 47(1) :113–124, 1999.
- [5] Christian Artigues, Jean-Charles Billaut, et Carl Esswein. Maximization of solution flexibility for robust shop scheduling. *European Journal of Operational Research*, 165(2) :314–328, September 2005.
- [6] Jean-Charles Billaut, Aziz Moukrim, et Eric Sanlaville. Introduction. Jean-Charles Billaut, Aziz Moukrim, et Eric Sanlaville, editors, *Flexibilité et robustesse en ordonnancement*, Traité IC2, pages 15–34. Hermes Science, Paris, December 2004.
- [7] Véronique Thomas. *Aide à la décision pour l'ordonnancement d'atelier en temps réel*. Thèse de doctorat, Université Paul Sabatier, Toulouse, 1980.
- [8] Guillaume Pinot et Nasser Mebarki. Best-case lower bounds in a group sequence for the job shop problem. *Proceedings of the 17th IFAC World Congress*, 2008.
- [9] Guillaume Pinot et Nasser Mebarki. Heuristiques pour le meilleur des cas dans un ordonnancement de groupes. *Actes de la 7ème Conférence Francophone de Modélisation et Simulation (MOSIM'08)*, 2008.
- [10] Guillaume Pinot et Nasser Mebarki. Le meilleur des cas dans un ordonnancement de groupes. *ROADEF 2008, Livre des résumés*, 2008.
- [11] Joseph Adams, Egon Balas, et Daniel Zawack. The shifting bottleneck procedure for job shop scheduling. *Management Science*, 34(3) :391–401, 1988.
- [12] Carl Esswein. *Un apport de flexibilité séquentielle pour l'ordonnancement robuste*. Thèse de doctorat, Université François Rabelais Tours, 2003.
- [13] Guillaume Pinot, Olivier Cardin, et Nasser Mebarki. A study on the group sequencing method in regards with transportation

in an industrial FMS. *Proceedings of the IEEE SMC 2007 International Conference*, 2007.

- [14] François Roubellat, Jean-Charles Billaut, et Michel Villaumie. Ordonnancement d'atelier en temps réel : d'ORABAID à ORDO. *Revue d'automatique et de productique appliquées*, 8(5) :683–713, 1995.
- [15] Julien Cegarra. *La gestion de la complexité dans la planification : le cas de l'ordonnancement*. Thèse de doctorat, Université de Paris 8, 2004.
- [16] Penelope M. Sanderson. The human planning and scheduling role in advanced manufacturing systems : an emerging human factors domain. *Human Factors*, 31(6) :635–666, 1989.
- [17] Emilie M. Roth, Kevin B. Bennett, et David D. Woods. Human interaction with an “intelligent” machine. Giuseppe Mancini, David D. Woods, et Erik Hollnagel, editors, *Cognitive Engineering in Complex Dynamic Worlds*, pages 23–69. Academic Press, London, 1988.