Méthodes exacte et approchée pour le problème de flow shop avec robot

Jacques Carlier[1], Mohamed Haouari[2,3], Mohamed Kharbeche[1,3], Aziz Moukrim[1]

[1] Université de Technologie de Compiègne, UMR CNRS 6599, Centre de Recherches de Royallieu, France
{jacques.carlier, mohamed.kharbeche, aziz.moukrim}@hds.utc.fr
[2] Department of Industrial Engineering, Faculty of Engineering, Ozyegin University, Istanbul, Turkey
mohamed.haouari@ozuegin.edu.tr
[3] Ecole Polytechnique de Tunisie, Combinatorial Optimization Research Group, 2078 La Marsa, Tunisie

Mots-Clés : *Flow-shop avec blocage, AG, Formulation mathématique, séparation et évaluation.*

1 Introduction

Notre contribution consiste à présenter une méthode exacte pour le problème de flow-shop avec un robot. Une méthode approchée pour ce problème a déjà été développée par Carlier et al.[1]. Cette méthode consiste à décomposer le problème en deux sous problèmes : Ordonnancement des tâches sur les machines et optimisation des mouvements du robot. On présentera, dans un premier lieu, une nouvelle borne inférieure basée sur l'affectation d'une tâche à la première position et une autre à la dernière position. Ensuite, une résolution approchée basée sur l'algorithme génétique utilisant des algorithmes de listes. Dans un dernier lieu et avant de présenter les résultats experimentaux, un programme linéaire et un algorithme de séparation et évaluation seront décrits.

2 Présentation du problème

Le problème du flow-shop avec un robot est une généralisation du problème du flow-shop de permutation. Ce problème peut être décrit comme suit. Etant donnés un ensemble de n tâches, m machines et un robot pour transférer ces tâches, à partir d'un lieu de chargement noté M_0, successivement sur les machines M_1, M_2,..,M_m dans le même ordre jusqu'à ce qu'elles atteignent le lieu de déchargement M_{m+1}. Une machine ne peut exécuter qu'une seule tâche à la fois, qui ne peut être traitée simultanément que sur une seule machine et la préemption n'est pas autorisée. Le robot ne peut transporter qu'une seule tâche à la fois. Dans cette ligne de production, la capacité de stockage entre les machines est nulle, ainsi qu'entre le lieu de chargement et la première machine, et le lieu de déchargement et la dernière machine. L'absence d'espace de stockage entre les machines conduit à des situations de blocage. En effet, la machine reste bloquée jusquà ce que l'opération de la tâche sur la machine suivante soit terminée et que cette dernière quitte cette machine. Une autre situation de blocage est due au robot lorsqu'une machine a déjà terminé le traitement d'une tâche et que ce dernier est déjà occupé par le transfert d'une autre tâche. L'objectif est de trouver la séquence des tâches sur les machines et d'optimiser les mouvements du robot afin de minimiser le temps de fin de la dernière tâche sur la dernière machine (makespan). Une étude extensive de ces problèmes et plus généralement des cellules robotisées se trouve dans le livre de Dawande et al.[2].

3 Borne inférieure et méthodes approchées

L'idée derrière cette nouvelle borne inférieure est de fixer une tâche à la première position et une autre à la dernière position. Ensuite, il faut calculer une borne inférieure en se basant sur la relaxation de deux machines consécutives.

Pour le problème du flow shop avec robot, nous proposons une méthode de résolution approchée basée sur l'algorithme génétique utilisant trois stratégies différentes. La première consiste à trouver la meilleure solution de la première phase de Carlier et al.[1] en utilisant l'algorithme génétique puis résoudre exactement les mouvements du robot de la séquence obtenue. La deuxième utilise un algorithme de liste dans le calcul du fitness dans l'AG. Cet algorithme consiste à traiter les opérations suivant leurs dates de début en commençant par celle qui a la date de début la plus petite. La troisième métaheuristique utilise un seul cycle du robot dans le calcul du fitness. Dans ce dernier cas, nous supposons que le robot fait un mouvement répététif quelle que soit la séquence des tâches. Ces métaheuristiques donnent un gap par rapport à la solution optimale inférieur à 1.05%.

4 Méthode exacte

Nous proposons une nouvelle formulation mathématique pour ce problème qui se base essentielement sur trois variables. Des variables binaires x_{ij}=1 si la tâche j est affectée à la position i et y_{lh}=1 si l'opération h du robot est exécutée immédiatement après l'opération l. Des variables continues $t_i, \sigma(j)$ la date de début du transfert entre la machine M_i et la machine $M_i + 1$ de la tâche à position j. Cette formulation mathématique nous a permis de résoudre des instances de petite taille de 16 tâches et 3 machines, 14 tâches et 4 machines et 12 tâches et 5 machines. La raison pour laquelle, nous avons développé une méthode exacte basée sur l'algorithme de séparation et évaluation. Dans cet algorithme, nous calculons une borne supérieure en utilisant une métaheuristique. Ensuite, nous développons l'arbre suivant la séquence obtenue en appliquant la stratégie de branchement profondeur d'abord. Lorsque toutes les tâches sont placées, on détermine les mouvements exacts du robot et on met à jour la borne supérieure. Nous avons pu conclure que cette méthode peut résoudre des instances de taille plus grande telque 18 tâches et 3 machines, 16 tâches et 4 machines et 14 tâches et 5 machines. De plus, le temps de calcul de l'algorithme de séparation et évaluation est beaucoup moins inférieur que celui de la fromulation mathématique.

Références

[1] J. Carlier, M. Haouari, M. Kharbeche and A. Moukrim. An optimization-based heuristic for the robotic cell problem. *European Journal of Operational Research*, 202 : 636–645, 2010.

[2] M.W. Dawande, H.N. Geismar, S.P. Sethi, C. Sriskandarajah, *Throughput Optimization in Robotic Cells*, Springer, 2007.