

Stockage intermédiaire dans l'industrie du shampoing

Rabah BELAID
rabah.belaid@etu.univ-tours.fr

Vincent T'KINDT
tkindt@univ-tours.fr

*
Carl ESSWEIN
carl.esswein@univ-tours.fr

Introduction

Nous abordons dans cette communication un problème de stockage intermédiaire dans l'industrie du shampoing. Des lots de produits cosmétiques liquides sont fabriqués puis stockés temporairement dans des cuves avant d'être conditionnés. Les différences des caractéristiques physico-chimiques entre les produits peuvent imposer le nettoyage d'une cuve avant sa réutilisation. L'objectif de ce travail est de planifier l'exploitation des cuves en vue de minimiser le nombre des nettoyages tout en respectant les contraintes industrielles. Une brève définition de ce problème est proposée dans la première section. La deuxième section portera sur la modélisation et la complexité de ce problème. La dernière section décrira de manière succincte les méthodes de résolution utilisées ainsi que les résultats préliminaires des évaluations numériques.

1 Présentation du problème

Nous disposons d'unités de fabrication qui produisent des produits cosmétiques par lots de 12 tonnes, appelés *batch*, et les transfèrent dans des cuves de stockage. Ces lots sont ensuite vidés des cuves lors du conditionnement. Un batch est une quantité fixe et indivisible d'un même produit cosmétique (*famille*). Tout batch i est fabriqué par une unité de fabrication, doit être stocké entièrement dans une seule cuve et doit aussi être entièrement conditionné par une seule ligne de conditionnement. Toutes les lignes de conditionnement suivent un planning fixé au préalable, on connaît donc pour chaque batch i la date de début au plus tôt de son conditionnement r_i . Nous disposons de L cuves de stockage qui sont des conteneurs dont la capacité varie entre 12, 20 et 24 tonnes. Si deux batches i et j de familles différentes, doivent se succéder dans la même cuve ℓ , un nettoyage d'une durée $S_{i,j}$ devra être effectué. Nous devons ordonnancer n batches, sur les L cuves de stockage en vue de minimiser le nombre des opérations de nettoyage.

*Université François-Rabelais Tours, Laboratoire d'Informatique, Polytech'Tours ; 64 avenue Jean Portalis, 37200 Tours, FRANCE

2 Modélisation et complexité

Dans la version généralisée du problème, Nous modélisons une cuve de stockage comme un flowshop à deux machines avec un buffer de capacité limitée. La première machine traite les opérations de transfert d'un batch depuis une unité de fabrication vers la cuve de stockage. La deuxième machine traite les opérations de vidage de la cuve qui sont contraintes par des dates de début au plus tôt. La capacité du buffer représente celle de la cuve, ainsi on ne pourra pas transférer dans une cuve plus de batches qu'elle ne peut en contenir. Les temps de nettoyage correspondent aux setups. Ces derniers sont bloquants et sont appliqués simultanément sur les deux machines ainsi que sur le buffer qui doit être vide. Cette version du problème est montré NP-difficile. Dans le cas particulier abordé dans ce travail, le nombre de cuves est supposé être suffisant et la date de début au plus tôt sur la deuxième machine est fixée comme date de début d'exécution impérative. Cette version est proche des problèmes d'ordonnancement à intervalles fixés (*Fixed interval scheduling* [1]) dont certaines propriétés vont servir à la résoudre. Si les capacités des cuves sont strictement inférieures à 2 *batch*, le problème est montré équivalent à un problème d'affectation et ainsi polynomial. Par contre, si la capacité des cuves est supérieure ou égale à 2 batches, une modélisation en problème d'affectation multi-index (NP-difficile [2]) est possible, mais la complexité n'est pas démontrée.

3 Méthodes de résolution et résultats

Nous proposons deux méthodes de résolution pour le problème des cuves dont la capacité est de deux batches. La première est un algorithme de liste muni d'un outil d'anticipation sur les L prochains batches de la liste. Une descente locale sur un voisinage 2-*opt* est ensuite appliquée sur la solution retournée. La deuxième méthode est un algorithme de colonies de fourmis artificielles [3] hybridé avec le recuit simulé [4]. Le paramétrage de l'algorithme de fourmis n'est toujours pas fixé, mais des tests préliminaires effectués sur des instances de petites tailles, ou les résultats des deux méthodes sont comparés aux solutions optimales retournées par CPLEX, montrent que la deuxième méthode retourne des résultats bien meilleurs que ceux de la première avec des temps d'exécution plus importants certes, mais acceptables. La tendance s'est confirmée lors des tests sur les instances de grandes tailles, ou les solutions des deux méthodes sont comparées entre elles. Ainsi toutes les solutions de l'algorithme de fourmis artificielles dominant celles de l'algorithme de liste couplé avec la descente locale. Ce résultat était attendu vu que l'algorithme de fourmis artificielles fait une exploration plus étendue l'espace des solutions.

References

- [1] M.Y. Kovalyov, C.T. Ng and T.C. Edwin Cheng. Fixed interval scheduling: Models, applications, computational complexity and algorithms. *EJOR*, 178 :331–342, 2007.
- [2] R. Burkard, M. Dell'Amico and S. Martello. Assignment problems. *SIAM*, 2009.
- [3] M. Dorigo and T. Stutzle. Ant Colony Optimization. *MIT press*, 2004.
- [4] V. T'kindt, N. Monmarché, F. Tercinet and D. Laugt. An Ant Colony Optimization algorithm to solve a 2-machine bicriteria flowshop scheduling problem. *EJOR*, 142 :250–257, 2002.