Le problème d'ordonnancement bi-objectif à une machine avec coûts de changement

H. Aissi[1], M.A. Aloulou[1], C. Artigues[2,3], N. Jozefowiez[2,3],P. Desseaux[2,3]

[1] LAMSADE ; Université Paris Dauphine, Place du Maréchal de Lattre de Tassigny, 75775 Paris Cedex 16
[2] CNRS ; LAAS ; 7 avenue du Colonel Roche, F-31077 Toulouse
[3] Université de Toulouse ; UPS, INSA, INP, ISAE ; LAAS ; F-31077 Toulouse
{Mohamed-Ali.Aloulou,Hassan.Aissi}@dauphine.fr,
{Christian.Artigues,Nicolas.Jozefowiez}@laas.fr

Mots-Clés : *Ordonnancement à une machine, coûts de changement, complexité, programmation dynamique, procédure de séparation & évaluation*

Nous considérons un problème d'ordonnancement à une machine et coûts de changements que nous notons $1|r_i, prec, c_{ij}|TSC \circ F$ dans sa forme la plus générale, c'est-à-dire un problème d'ordonnancement de n tâches de durées $(p_i)_{1\leq i\leq n}$ avec dates de lancement $(r_i)_{1\leq i\leq n}$, contraintes de précédences, et coûts de changement dépendant de la séquence $(c_{ij})^{1\leq i\leq n}_{1\leq j\leq n}$. On cherche à obtenir l'ensemble Pareto optimal pour la minimisation de la somme des coûts de changement TSC et d'une fonction $F \in \{\sum C_i, \sum U_i, L_{\max}\}$ où C_i est la date de fin de la tâche i, $U_i \in \{0,1\}$ est l'indicateur de retard de la tâche i et $L_{\max}$ est le plus grand retard algébrique des tâches. Ainsi $U_i = 1$ si et seulement si $C_i > d_i$ et $L_{\max} = \max_i(C_i - d_i)$ où d_i est la date de livraison de la tâche i. On considère le cas particulier des coûts de changement par familles (notés c_{fg}) où chaque tâche i appartient à une famille $f_i \in F$, avec $|F| \leq n$ et où les tâches de même famille peuvent être séquencées à coût nul. Nous considérons également le cas des coûts de changement par famille indépendants de la séquence (notés c_f) où le coût de changement pour passer d'une famille à l'autre est constant. Enfin, dans le cas où $c_f = 1$, TSC représente le nombre de changement de familles (on parlera de *couleurs*). Cette notation est empruntée à [5] où à la place des coûts de changement, des temps de préparation sont considérés. Les coûts de changement ont été rarement considérés dans la littérature indépendamment des temps de préparation [1], malgré leur intérêt pratique comme par exemple pour représenter des coûts monétaires ou des pénalités sur tout changement de configuration non quantifiable en temps.

1 Résultats de complexité

Dans sa forme générale, le problème est évidemment NP-difficile car la seule minimisation des coûts de changement sans contraintes de précédence revient au problème du voyageur de commerce. Nous étudions plusieurs cas particuliers dont nous établissons la complexité.

Pour le critère du nombre de tâches en retard, le problème à durées et coûts de changement par familles unitaires $1|c_f = 1, p_i = 1|TSC \circ \sum U_i$ est NP-difficile du fait de résultats connus sur les problèmes d'ordonnancement sous technologie de groupe où le coût de changement minimal est imposé ($TSC = |F| - 1$) [4]. Nous prouvons que le problème $1|p_i = 1, c_{fg}, |F| = m|TSC \circ \sum U_i$ à nombre de familles constant ($|F| = m$) est polynomial.

Pour le critère de la somme des dates de fin, nous proposons un algorithme en $O(n \log n)$ pour le problème $1|c_f, TSC = |F| - 1|\sum C_i$ (avec contrainte de technologie de groupe). En montrant au préalable qu'il existe un ordonnancement optimal dans lequel les tâches de même famille sont séquencées selon la règle de plus petite durée (SPT), nous proposons un algorithme de programmation dynamique pseudopolynomial en $O(mn^{m+1}c_{max})$ avec $c_{max} = \max c_{fg}$ lorsque le nombre de familles est constant pour le problème $1|c_{fg}, |F| = m|TSC \circ \sum C_i$. Pour les coûts de changement indépendants de la séquence $(1|c_f, |F| = m|TSC \circ \sum C_i)$, la complexité devient $O(mn^{m+1})$, ce qui est polynomial pour un nombre de familles m fixé (par exemple $O(n^3)$ pour deux familles).

Pour le critère du plus grand retard, nous proposons également un algorithme en $O(n \log n)$ pour le problème $1|c_f, TSC = |F| - 1|\sum L_{\max}$ (avec contrainte de technologie de groupe). Nous montrons par ailleurs qu'il existe une solution optimale du problème $1|c_{fg}|TSC \circ L_{\max}$ telle que toutes les tâches de même famille sont ordonnancées dans l'ordre croissant des dates de livraison (règle EDD). Nous montrons que le problème avec changements unitaires $(1|c_f = 1, |F| = m|TSC \circ \sum L_{\max})$ est polynomial lorsque le nombre de couleurs est constant et proposons également un algorithme de programmation dynamique.

Pour les problèmes à contraintes de précédence, le problème mono-objectif de minimisation du nombre de changement de couleurs $1|r_i, c_f = 1, prec|TSC$ est polynomial et peut être résolu en $O(n + |prec|)$ par un algorithme de tri topologique [3].

2 Procédure de séparation et d'évaluation

Dans cette section, le problème de minimisation du nombre de changements de couleur et du plus grand retard avec dates de lancement et de livraison $1|r_i, c_f = 1, |TSC \circ L_{\max}$ est considéré. Ce problème est NP-difficile puisque le problème mono-objectif obtenu en ignorant les couleurs $1|r_i|L_{\max}$ l'est. Ce dernier problème est toutefois très bien résolu par la procédure de séparation et évaluation (PSE) de Carlier [2]. Nous proposons une méthode ϵ-contrainte résolvant itérativement des problèmes $1|r_i, c_f = 1, TSC \leq \epsilon|L_{\max}$ par une PSE inspirée de l'algorithme de Carlier. Nous utilisons une méthode de branchement basé sur l'analyse du bloc critique d'une solution réalisable. Des contraintes de précédence sont ajoutées à chaque branchement et l'agorithme de Darte [3] nous permet d'avoir la valeur exacte du nombre de changements de couleurs minimal encore atteignable. La borne inférieure utilisée par Carlier pour le $L_{\max}$ est reprise. Des algorithmes de propagation de contraintes et des heuristiques spécifiques sont également intégrés. Les résultats sur des instances de Carlier modifiées montrent que la méthode est capable d'obtenir l'ensemble Pareto optimal pour $n = 20$ et une approximation intéressante au-delà.

Références

[1] A. Allahverdi, C.T. Ng, T.C. E. Cheng, M. Y. Kovalyov. A survey of scheduling problems with setup times or costs *European Journal of Operational Research* 187: 985–1032, 2008.

[2] J. Carlier. The one-machine scheduling problem *European Journal of Operational Research* 11: 42–47, 1982.

[3] A. Darte. On the complexity of loop fusion *Parallel computing* 26: 1175–1193, 2000.

[4] Z. Liu and W. Yu. Minimizing the number of late jobs under the group technology assumption *Journal of Combinatorial Optimization* 3: 5–15, 1999.

[5] C. N. Potts, M. Y. Kovalyov., Scheduling with batching : A review *European Journal of Operational Research*, 120: 228–249, 2000.