Problème d'ordonnancement cyclique de base robuste

Idir Hamaz[1,2], Laurent Houssin[1,2], Sonia Cafieri[3]

[1] CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France
idir.hamaz@laas.fr
[2] Univ de Toulouse, UPS, LAAS, F-31400 Toulouse, France
[3] ENAC, Université de Toulouse, F-31055 Toulouse, France

Mots-clés : *Ordonnancement cyclique, optimisation robuste.*

1 Description du problème

Nous considérons dans ce papier une version robuste du problème d'*ordonnancement cyclique de base*. La version non cyclique du problème a été traitée par Minoux [4] mais la robustesse reste inexplorée dans les problèmes d'ordonnancement cyclique. Le problème d'ordonnancement cyclique de base occupe une place importante dans le domaine d'ordonnancement cyclique. Ceci est dû au fait qu'il est à la base de la résolution de plusieurs extensions du problème (voir [3]).

Le problème d'ordonnancement cyclique de base est définit par un ensemble $T = \{1, ..., n\}$ de tâches génériques. Chaque tâche i a une durée p_i et doit être exécutée une infinité de fois. On note $< i, k >$ la $k^{éme}$ occurrence de la tâche i. Un ordonnancement σ est une affectation des dates de début $t(i, k)$ pour chaque occurrence $< i, k >$. Un ordonnancement est dit *périodique* avec un temps de cycle α si

$$t(i, k) = t(i, 0) + \alpha k, \quad \forall i \in T,\ \forall k \geq 1. \tag{1}$$

Afin de simplifier la notation, nous définissions $t_i = t(i, 0)$ pour chaque tâche i dans T. Un ordonnancement périodique peut être défini seulement avec les dates de début des premières occurrences $(t_i)_{i\in T}$ ainsi que le temps de cycle α.

Le différentes tâches sont liées par un ensemble de contraintes de précédence génériques dites *contraintes uniformes*. Ces contraintes peuvent être représentées par un triplet (i, j, H) et sont données par :

$$t(i, k) + p_i \leqslant t(j, k + H_{ij}), \quad \forall i, j \in T,\ \forall k \geq 0. \tag{2}$$

En réexprimant les dates de début en fonction de $(t_i)_{i\in T}$ et du temps de cycle α, cela donne :

$$t_j - t_i + H_{ij}\alpha \geqslant p_i, \quad \forall i, j \in T,\ \forall k \geq 0, \tag{3}$$

où i et j sont deux tâches génériques, et H_{ij} un entier naturel, appelé *hauteur*, qui représente le décalage événementiel entre les occurrences des tâches i et j.

L'objectif du problème est de trouver un ordonnancement σ_{opt} qui satisfait les contraintes uniformes (3) et minimise le temps de cycle α.

Un graphe orienté $G = (V, E)$ peut être associé à ce problème. Un nœud du graphe G représente une tâche et un arc (i, j) existe entre les nœuds i et j si une contrainte de précédence existe entre les deux tâches correspondantes. Dans ce cas, l'arc (i, j) est étiqueté par deux valeurs, une longueur $L_{ij} = p_i$ et une hauteur H_{ij}. Deux nœuds supplémentaires 's' et 'e' sont introduits dans le graphe G ainsi que l'arc (e, s) étiqueté par une longueur 0 et une hauteur 1.

Soit la longueur $L(c)$ (resp. la hauteur $H(c)$) du circuit c du graphe G, la somme des longueurs (resp. hauteurs) de ses arcs. Le temps de cycle minimum est donné par poids moyen maximum du graphe G. Celui-ci est donné par

$$\alpha = \max_{c\in\mathcal{C}} \frac{\sum_{(i,j)\in c} L_{ij}}{\sum_{(i,j)\in c} H_{ij}}$$

Où $\mathcal{C}$ est l'ensemble des circuits du graphe G.

Le circuit c donnant le temps de cycle moyen maximum est appelé *circuit critique*. Différents algorithmes sur le calcul de circuits critiques peuvent être trouvés dans la littérature [2].

2 Introduction de l'incertitude

Les sources d'incertitude dans les problèmes d'ordonnancement sont multiples. Des pannes de machines peuvent survenir, de nouvelles tâches prioritaires peuvent arriver et les durées des tâches qui peuvent varier, etc. Dans ce travail, nous considérons que les durées des tâches sont incertaines et peuvent dévier de leurs valeurs nominales ($p_i \in [\bar{p}_i,\, \bar{p}_i + \hat{p}_i]$). La modélisation des durées incertaines $p_i(\xi)$ est basée sur l'approche introduite par Bertsimas et Sim [1]. Nous supposons qu'au plus Γ tâches peuvent dévier de leurs valeurs nominales :

$$\begin{aligned} p_i(\xi) = &\quad \bar{p}_i + \xi_i \hat{p}_i, \forall \xi \in \Xi, i \in T \\ \text{avec } \Xi = &\quad \left\{ (\xi_i)_{1 \le i \le T} \middle| \sum_{i=1}^{T} \xi_i \le \Gamma, \xi_i \in \{0,1\} \right\} \end{aligned}$$

Les variations des durées des tâches peuvent faire augmenter la valeur du temps de cycle et ainsi, casser la cyclicité de notre ordonnancement. Notre approche consiste à déterminer le temps de cycle α avant de connaître les durées réelles des tâches. Une fois l'aléa révélé, nous devons décider des dates de début des tâches. L'objectif est de chercher un temps de cycle α d'une valeur minimum, de sorte que les contraintes uniformes puissent êtres satisfaites pour tout scénario $\xi \in \Xi$. Autrement dit, si nous définissons l'ensemble D comme suit :

$$D = \{\bar{\alpha} \,|\, \forall \xi \in \Xi,\, \exists\, (t_i)_{i \in T} : t_j - t_i \ge p_i(\xi) - H_{ij}\bar{\alpha},\, \forall\, (i,j) \in E\}$$

alors, nous cherchons à résoudre :

$$\min_{\alpha \in D} \alpha$$

Le problème décrit ci-dessus est un problème robuste à deux niveaux. L'objectif du problème dépend uniquement de la variable du premier niveau α et les variables du second niveau assure la faisabilité du problème. Nous avons développé un algorithme pour la résolution du problème robuste. Celui-ci s'appuie sur un sous problème qui est une modification d'un algorithme de détection de circuit négatif (basé sur l'algorithme de Belleman-Ford). Nous montrons aussi que la valeur α_Γ du circuit critique robuste est donnée par :

$$\alpha_\Gamma = \max_{c \in \mathcal{C}} \left\{ \frac{\sum_{(i,j) \in c} \bar{p}_i}{H(c)} + \max_{\xi \in \Xi} \frac{\sum_{(i,j) \in c} \hat{p}_i \xi_i}{H(c)} \right\}$$

Références

[1] D. Bertsimas, M. Sim. *The Price of Robustness.* Operations Research, 52 (1) : 35–53, 2004.

[2] A. Dasdan. *Experimental analysis of the fastest optimum cycle ratio and mean algorithms.* ACM Transactions on Design Automation of Electronic Systems, 9 (4) : 385–418, 2004.

[3] C.Hanen, A. Munier. *Cyclic scheduling on parallel processors : an overview.* Scheduling Theory and Its Applications, chapter cyclic scheduling on parallel processors : an overview, 1995.

[4] M. Minoux. *Duality, Robustness and 2-Stage Robust Decision Models. Application to Robust PERT Scheduling.* Annales du LAMSADE, University Paris-Dauphine, France, 2007.