Borne inférieure basée sur la programmation linéaire pour un problème d'ordonnancement avec production et consommation des ressources

Abderrahim Sahli, Jacques Carlier, Aziz Moukrim
Sorbonne universités, Université de Technologie de Compiègne,
CNRS, laboratoire Heudiasyc UMR 7253, CS 60 319,
57 avenue de Landshut 60 203 Compiègne cedex, France
{abderrahim.sahli,jacques.carlier,aziz.moukrim}@utc.fr

Mots-clés : *ordonnancement, borne inférieure, programmation linéaire.*

1 Introduction

La plupart des travaux de recherches sur les problèmes d'ordonnancement traitent le cas des ressources renouvelables, c'est-à-dire des ressources qui sont exigées en début d'exécution de chaque tâche et sont restituées en fin d'exécution. Peu d'entre eux abordent les problèmes à ressources consommables, c'est-à-dire des ressources non restituées en fin d'exécution. Le problème de gestion de projet à contraintes de ressources (RCPSP) est le problème à ressources renouvelables le plus traité dans la littérature. Plus récemment, certains auteurs ont abordé des modèles plus généraux pour intégrer à la fois les ressources consommables et les ressources renouvelables. On parle alors des ressources de stock. C'est le cas des travaux de Neumann et Schwindt [5] et de Laborie [4].

Dans ce papier, nous nous intéressons au calcul de bornes inférieures pour une généralisation du problème RCPSP qui correspond au cas où les tâches sont remplacées par des évènements. Ces derniers sont liés par des relations de précédence et peuvent produire ou consommer une quantité de ressources à leur date d'occurrence. La fonction économique reste la même à savoir la minimisation de la durée totale du projet. Nous avons nommé cette généralisation ERCPSP.

2 Définition du problème

Un problème ERCPSP [3] est un problème d'optimisation combinatoire défini par un 5-uplet (X, U, K, a, v), où X est un ensemble d'évènements, U est un ensemble de relations de précédence et K est un ensemble de ressources. Une instance de ce problème est constituée de n évènements identifiés par $A = \{1, ..., n\}$ et deux évènements fictifs 0 et $n + 1$. L'évènement 0 représente par convention le début de l'ordonnancement (début du projet), et l'évènement $n + 1$ représente symétriquement la fin de l'ordonnancement ($X = A \cup \{0, n + 1\}$).

Pour un évènement i, a_i^k représente la quantité de la ressource k consommée ou produite par cet évènement à sa date d'occurrence S_i. Si $a_i^k \geq 0$ alors l'évènement i produit a_i^k unités de ressource k, sinon il consomme $|a_i^k|$ unités de ressource k. Les relations de précédence sont données par un ensemble de paires d'indices d'évènements. Si $(i, j) \in U$ alors v_{ij} désigne le temps de latence entre l'évènement i et l'évènement j. Si $v_{ij} \geq 0$ alors l'évènement j ne peut être planifié avant $S_i + v_{ij}$, sinon l'évènement i doit être planifié avant $S_j - v_{ij}$. Un ordonnancement S est une fonction qui affecte à chaque évènement i une date d'occurrence S_i. On dit que S est admissible si toutes les contraintes de ressources et de précédence sont satisfaites. Résoudre le problème ERCPSP consiste à trouver un ordonnancement S pour lequel S_{n+1} est minimal.

3 Calcul de bornes inférieures

Dans des travaux récents [1], nous avons proposé plusieurs bornes inférieures pour le problème ERCPSP qui exploitent des travaux sur les problèmes cumulatifs. Dans cette section, nous allons présenter deux autres bornes basées sur la programmation linéaire et inspirées par des travaux sur le problème RCPSP [2].

Soient $[t_0, t_1[, [t_1, t_2[, ..., [t_{L-1}, t_L[$ L intervalles de temps successifs qui représentent une décomposition de l'horizon d'ordonnancement (t_L est une borne supérieure pour la durée totale du projet). Une relaxation du problème ERCPSP peut être définie en permettant aux évènements de s'exécuter partiellement et de manière ponctuelle dans plusieurs intervalles de temps. Soit $x_{i,l} \in [0,1]$ une variable continue représentant la proportion d'exécution de l'évènement i dans l'intervalle $[t_l, t_{l+1}[$. La quantité de ressources k produite ou consommée par i dans $[t_l, t_{l+1}[$ est déterminée par $a_i^k * x_{i,l}$. La relaxation du problème ERCPSP est formulée comme suit :

$$\sum_{l=0}^{L-1} x_{i,l} = 1 \qquad \forall i \in X \tag{1}$$

$$\sum_{l=0}^{h} \sum_{i=0}^{n+1} a_i^k \times x_{i,l} \geq 0 \qquad \forall h \in [0...L-1], \forall k \in K \tag{2}$$

$$\sum_{l=0}^{h_1} x_{i,l} - \sum_{l=0}^{h_2} x_{j,l} \geq 0 \qquad \forall (i,j) \in U, \forall (h_1,h_2) \in [0...L-1]^2,\ t_{h_2+1} - v_{ij} - 1 \in [t_{h_1}, t_{h_1+1}[\tag{3}$$

$$x_{i,l} \in [0,1] \qquad \forall l \in [0...L-1], \forall i \in X \tag{4}$$

$$t_0 = 0 < t_1 < ... < t_L \tag{5}$$

On peut utiliser cette formulation linéaire pour calculer des bornes inférieures destructives. En effet, si pour une décomposition donnée le programme linéaire n'admet pas une solution, on en déduit que t_L est une borne inférieure. La méthode de décomposition peut se faire de manière polynomiale si on considère les dates au plus tôt et au plus tard des évènements comme bornes des intervalles successifs. Elle peut aussi se faire de manière pseudo-polynomiale si on considère des intervalles de durée 1. Nous avons testé ces deux méthodes de décomposition sur les instances générées par Neumann et Schwindt [5]. Les résultats obtenus montrent que les bornes sont d'excellente qualité et meilleures que ceux proposées dans [1]. En effet, pour 93% des instances la durée optimale est atteinte et l'écart moyen *GAP* est inférieur à 0.8%. Nous avons aussi remarqué que la borne pseudo-polynomiale est meilleure que la borne polynomiale, mais elle est plus couteuse en terme de temps CPU.

Références

[1] J. Carlier, A. Moukrim, and A. Sahli. Lower bounds for the event scheduling problem with consumption and production of resources. *Discrete Applied Mathematics*, pages –, 2016.

[2] J. Carlier and E. Néron. On linear lower bounds for the resource constrained project scheduling problem. *European Journal of Operational Research*, 149(2) :314 – 324, 2003. Sequencing and Scheduling.

[3] Jacques Carlier, Aziz Moukrim, and Huang Xu. The project scheduling problem with production and consumption of resources : A list-scheduling based algorithm. *Discrete Appl. Math.*, 157(17) :3631–3642, October 2009.

[4] Philippe Laborie. Algorithms for propagating resource constraints in AI planning and scheduling : Existing approaches and new results. *Artif. Intell.*, 143(2) :151–188, February 2002.

[5] Klaus Neumann and Christoph Schwindt. Project scheduling with inventory constraints. *Mathematical Methods of Operations Research*, 56(3) :513–533, 2002.