Ordonnancement sur machines parallèles de tâches sécables avec temps de réglage uniformes

Aurélien Mombelli, Laurent Deroussi, Nathalie Grangeon, Sylvie Norre, Matthieu Py

Université Clermont Auvergne, CNRS, Clermont Auvergne INP, Mines Saint-Etienne, LIMOS, 63000 Clermont-Ferrand, France

prenom.nom@uca.fr

Mots-clés : *machines parallèles, Temps de réglage, Tâches Sécables*

1 Introduction

Nous étudions un problème d'ordonnancement où les tâches peuvent être fractionnées pour être traitées par plusieurs machines. Un temps de réglage est nécessaire sur une machine pour passer d'une tâche à une autre. Pendant ce réglage, une machine ne peut ni traiter ni préparer une autre tâche. Les temps de réglage sont indépendants de la tâche, de la machine et de la séquence. Les tâches ont une durée minimum de traitement et les machines doivent être utilisées à tout instant d'un horizon fixé. L'objectif est de minimiser le nombre de réglages nécessaires. La figure 1 montre un exemple d'ordonnancement sur quatre machines, un horizon de cinq heures et les durées minimales suivantes : $T_1 \geq 6$, $T_2 \geq 2$, $T_3 \geq 2$, $T_4 \geq 1$, $T_5 \geq 1$, $T_6 \geq 1$, $T_7 \geq 1$.

<table>
<tr><th></th><th>1h</th><th>2h</th><th>3h</th><th>4h</th><th>5h</th></tr>
<tr><td>machine 1</td><td colspan="2">tâche 1</td><td>réglage</td><td colspan="2">tâche 2</td></tr>
<tr><td>machine 2</td><td colspan="3">tâche 3</td><td>réglage</td><td>tâche 4</td></tr>
<tr><td>machine 3</td><td colspan="5">tâche 1</td></tr>
<tr><td>machine 4</td><td>tâche 5</td><td>réglage</td><td>tâche 6</td><td>réglage</td><td>tâche 7</td></tr>
</table>

FIG. 1 – Exemple d'ordonnancement de sept tâches sur quatre machines sur un horizon de temps de cinq heures. les tâches 1 et 3 durent une heure de plus que leur durée minimale.

Ce problème est très proche de celui étudié par Schalekamp et al [1] dont l'objectif est la minimisation du makespan. Leur étude décrit des propriétés des solutions optimales, démontre la complexité NP-Hard du problème et propose un schéma d'approximation en temps polynomial.

Notre problème diffère légèrement en deux points : nos tâches ont une durée minimale non fixe et notre objectif est de minimiser le nombre de réglages.

2 Modélisation du problème

Nous avons choisi de discrétiser l'horizon de temps (24h) en créneaux d'une heure. Pendant un créneau, soit la machine réalise une tâche, soit elle est en réglage. Tout réglage dure une heure.

En utilisant les notations suivantes :

- M : l'ensemble des machines
  $i \in M$ une machine
- N : l'ensemble des tâches
  $j \in N$ une tâche

- $N' = N \bigcup \{0\}$: l'ensemble des tâches et réglage
  $j \in N'$ une tâche ou un réglages
- $(T_j)_{j \in N}$: l'ensemble des temps minimum d'affectation par tâches (T_j est le temps minimum en heures pour la tâche j)
- C : l'ensemble des 24 créneaux d'1h.
- $(X_{i,j,t} \in \{0,1\})_{\substack{i \in M \\ j \in N' \\ t \in C}}$: l'ensemble des variables de décisions affectant la tâche j sur la machine i à la période t.

Le problème peut donc s'écrire ainsi :

$$\begin{array}{rlll}
\textbf{minimiser} & \sum\limits_{\substack{i \in M \\ t \in C}} X_{i,0,t} & & (P.*) \\
\textbf{tel que} & \sum\limits_{j \in N'} X_{i,j,t} = 1 & \forall \substack{i \in M \\ t \in C} & (P.1) \\
& X_{i,j,t} = X_{i,j,t+1} + X_{i,0,t+1} & \forall \substack{i \in M \\ j \in N \\ t \in C \setminus \{24\}} & (P.2) \\
& X_{i,0,1} = 0 & \forall i \in M & (P.3) \\
& \sum\limits_{\substack{i \in M \\ t \in C}} X_{i,j,t} \geq T_j & \forall j \in N & (P.4)
\end{array}$$

L'objectif (P.*) est de minimiser le nombre de réglages sur l'ensemble des machines et créneaux. (P.1) impose qu'une et une seule tâche ou réglage soit sélectionné sur chaque machine et chaque créneau. (P.2) exprime le fait qu'une tâche (autre qu'un réglage) est suivie par elle-même ou un réglage. (P.3) empêche qu'un réglage soit sélectionné sur le premier créneau. (P.4) impose que chaque tâche soit affectée suffisamment pour satisfaire sa durée minimale.

Afin de faciliter la résolution, certaines propriétés des solutions sont recherchées :

Proposition 1 : *Soit une tâche j tel que $T_j \geq 24$, il existe une solution optimale dans laquelle 24 créneaux sont affectées sur une machine à cette tâche.*

Proposition 2 : *Dans une solution optimale, si une tâche est affectée à une machine, alors l'ensemble des créneaux de cette tâche sur cette machine est contigu.*

Proposition 3 : *S'il existe un ensemble de G tâches $(j_1, \cdots, j_G)$ tel que*

$$23 \leq \sum_{j \in \{j_1, \cdots, j_G\}} T_j + (G-1) \leq 24$$

Alors il existe une solution optimale dans laquelle cet ensemble de tâches est affecté à une même machine.

Ces propriétés seront utilisées, si elles sont vraies, en prétraitement d'une heuristique.

3 Conclusions et perspectives

La première étape est de démontrer ces propositions puis de les traduire en contraintes. Le modèle n'a, cependant, pas beaucoup de chances de traîter des instances de grande taille en des temps raisonables puisque nos instances peuvent monter jusqu'à 20 machines et 60 tâches. Nous envisageons la création de schémas d'approximations qui devraient être plus simples que ceux proposés par Schalekamp et al [1] car nos bornes sont plus facilement exprimables.

Références

[1] Frans Schalekamp, René Sitters, Suzanne van Der Ster, Leen Stougie, Víctor Verdugo, Anke van Zuylen *Split scheduling with uniform setup times.* Journal of scheduling, 18(2), 119-129, 2015.