

Propagation de contraintes généralisées pour la résolution du Job Shop avec “time lags”

M.-J. Huguet^{1,2}, C. Artigues^{1,2}, M. Dugas^{1,2}, P. Lopez^{1,2}

¹ CNRS; LAAS; 7 avenue du colonel Roche, F-31077 Toulouse, France

{dugas, huguet, artigues, lopez}@laas.fr

² Université de Toulouse; UPS, INSA, INP, ISAE; LAAS; F-31077 Toulouse, France

Mots-Clés : *Job Shop, Time Lag, Propagation de contraintes, Branch and Bound*

1 Présentation du problème

Le problème de Job Shop avec Time Lag (JSTL) est une généralisation du problème de Job Shop dans lequel il existe des contraintes temporelles visant à limiter la distance minimale et maximale entre deux opérations consécutives d’un même job. Le problème JSTL comporte un ensemble $J = \{J_i\}_{1..n}$, de n jobs devant être réalisés sur un ensemble $M = \{M_k\}_{k=1..m}$ de m machines. Chaque job J_i est constitué d’une séquence de n_i opérations; on note O_{ij} la $j^{\text{ème}}$ opération du job J_i . Chaque opération O_{ij} doit être exécutée sans interruption sur une unique ressource $m_{ij} \in M$. De plus, la distance entre deux opérations O_{ij} et O_{ij+1} avec $1 \leq j < n_i$ est limitée par une valeur minimale TL_{ijj+1}^{\min} et une valeur maximale TL_{ijj+1}^{\max} . On cherche à déterminer un ordonnancement dont la durée totale (*makespan*) est minimale. Il existe peu de travaux dans la littérature sur le JSTL. Dans [4], les auteurs considèrent un problème d’ordonnancement à une machine avec contraintes de time lags. Ils ont montré que, sauf pour certains types de contraintes de précédence entre jobs, le problème était NP-difficile. Dans [1], les auteurs ont montré que de nombreux problèmes d’ordonnancement, comme par exemple les problèmes d’ordonnancement multi-machines ou les problèmes d’ordonnancement avec machines polyvalentes, pouvaient se ramener à un problème d’ordonnancement à une machine avec time lag minimum et maximum. Ils ont proposé également une méthode exacte pour sa résolution. Plus récemment, [2] ont étudié le problème de JSTL et ont proposé un algorithme mémétique pour sa résolution. L’objectif de notre travail est de montrer l’apport de propagation de contraintes temporelles généralisées pour la résolution du problème JSTL. Pour résoudre le JSTL, nous proposons une heuristique basée sur un classement des jobs et une méthode arborescente (branch-and-bound) incluant des propagations de contraintes temporelles généralisées.

2 Méthode de résolution

En raison de la présence de contraintes de time lags maximaux, les heuristiques de liste classiquement utilisées pour la résolution des problèmes de Job Shop ne peuvent être facilement étendues. Un ordonnancement réalisable trivial pour le JSTL (de très mauvaise qualité vis-à-vis du makespan) correspond au placement des différents jobs les uns à la suite des autres. Nous proposons ici une

nouvelle heuristique basée sur un classement des jobs. Pour chaque job J_i , l'heuristique tente de positionner les différentes opérations O_{ij} qui le compose dans des espaces disponibles sur les machines compte tenu des contraintes de time lag à respecter entre chacune de ces opérations. Si le placement ne s'avère pas possible, le job est décalé à la suite des jobs précédemment placés. Dans le pire des cas, notre heuristique fournit l'ordonnancement trivial présenté ci-dessus. Les différents classements étudiés pour les jobs sont : l'ordre lexicographique, l'ordre croissant ou décroissant des durées des jobs et l'ordre croissant ou décroissant de la somme des time lags maximaux et des durées des jobs.

Le branch-and-bound (B&B) proposé permet de déterminer s'il existe ou non une solution au JSTL ayant un makespan inférieur ou égal à une valeur H donnée. Ce B&B est inclus dans une procédure de recherche dichotomique faisant varier les valeurs de l'horizon H entre une borne inférieure obtenue par propagation de contraintes et une borne supérieure fournie par l'heuristique présentée précédemment. Une des spécificités de ce B&B est d'exploiter à chaque noeud une généralisation des propagations de contraintes de ressources utilisées classiquement en ordonnancement. Ces propagations exploitent un graphe temporel disjonctif dans lequel chaque opération est représentée par deux noeuds correspondant à ses dates de début et de fin. Les propagations se basent, d'une part, sur un algorithme incrémental de recherche de plus longs chemins et, d'autre part, sur des extensions de règles classiques exploitant les résultats de l'algorithme de recherche de plus longs chemins. Sur ce principe, des règles de propagation telles que les précédences interdites, certains raisonnements de sélections immédiates, le raisonnement énergétique ont pu être généralisés. Ce n'est pas actuellement le cas pour la règle classique de *edge-finding*.

3 Expérimentations

Nous considérons les instances de JSTL proposées par [2] qui sont une généralisation de certains problèmes de job shop à des time lags maximaux entre chaque paire d'opérations consécutives d'un même job. Ces instances sont caractérisées par un paramètre α tel que plus la valeur de α est faible et plus la contrainte de time lag est forte. Les expérimentations menées montrent l'apport des propagations de contraintes généralisées y compris par rapport à la règle de *edge-finding*. Sur les 58 instances testées, le B&B trouve l'optimum pour 36 d'entre elles. Les propagations généralisées sont les seules à conduire à l'optimum pour 32 instances. Les propagations non généralisées et le *edge-finding* sont les seules à obtenir l'optimum dans 1 cas seulement alors qu'on dénombre 3 cas d'égalité. Par ailleurs, le B&B s'avère plus performant que l'algorithme mémétique de [2] uniquement sur les problèmes de petite taille (la01 à la05) avec $\alpha \in \{0, 5, 1, 2\}$ pour lesquels nous trouvons toujours l'optimum alors que l'algorithme mémétique présente une déviation de 5% à l'optimum. En revanche pour les problèmes sans attente ($\alpha = 0$) ou pour des problèmes de plus grande taille (pour lesquels [2] ont proposé des résultats) l'algorithme mémétique est plus performant.

Références

- [1] P. Brucker, T. Hilbig, J. Hurink. A branch and bound algorithm for a single machine scheduling problem with positive and negative time lags. *Discrete Applied Mathematics*, 94:77–99, 1999.
- [2] A. Caumont, Ph. Lacomme, N. Tchernev. A memetic algorithm for the job shop with time-lags. *Computers and Operations Research*, 35:2331–2356, 2008.
- [3] J. Hurink, J. Keuchel. Local search algorithm for a single-machine scheduling problem with positive and negative time-lags. *Discrete Applied Mathematics*, 112:179–197, 2001.
- [4] E.D. Wikum, D.C. Llewellyn, G.L. Nemhauser. One-machine generalized precedence constrained scheduling problems. *Operations Research Letters*, 16(2):87–99, 1994.