Branch and Bound hybride pour un problème de job-shop soumis à des contraintes de ressources humaines

Olivier Guyon[1], Pierre Lemaire[2], Eric Pinson[3], David Rivreau[3]

[1] Ecole des Mines de Saint-Etienne, Centre Microélectronique de Provence
880 avenue de Mimet 13541 Gardanne (France)

[2] Grenoble-INP / UJF-Grenoble 1 / CNRS, G-SCOP UMR5272
46 avenue Félix Viallet, 38031 Grenoble cedex 1 (France)

[3] PRES LUNAM Université / LISA EA4094 CNRS / Université Catholique de l'Ouest
3 Place André Leroy 49008 Angers (France)

`contact : eric.pinson@uco.fr`

Mots-clés : *branch and bound, job-shop, planification d'agents, probing.*

1 Description du problème

Nous considérons un atelier où la production à réaliser requiert divers types de machines dans des séquences variées de type job-shop. Chaque machine nécessite pour son utilisation la présence d'un employé qualifié à son pilotage. Les ressources humaines sont assujetties à des contraintes légales restreignant leur disponibilité. La production doit être entièrement ordonnancée et le critère d'optimisation retenu est la minimisation des coûts salariaux. Ce problème coïncide avec le cas particulier des instances *correspondance activité-ressource* de [AGRV09].

2 Formalisation en Programme Linéaire en variables 0-1

On cherche à produire un ensemble J de n jobs sur m machines. Chaque job i est caractérisé par une séquence propre d'opérations $\{O_{ij}\}_{j=1,\ldots,m}$ non préemptives. O_{ij} s'effectue sur la machine m_{ij} et sa durée est p_{ij}. Par la suite, les notations classiques ρ_{ik}, r_{ik} et d_{ik} désignent respectivement la durée, la date de démarrage au plus tôt et la date échue de l'opération du job i s'exécutant sur la machine k ; r_{ik} et d_{ik} étant définies par récurrence selon (1).

$$\begin{array}{lll|lll}
r_{ik} = 0 & i = 1,\ldots,n & k = m_{i1} & d_{ik} = C_{max} & i = 1,\ldots,n & k = m_{im} \\
r_{il} = r_{ik} + \rho_{ik} & i = 1,\ldots,n & j = 1,\ldots,m-1 & d_{ik} = d_{il} - \rho_{il} & i = 1,\ldots,n & j = 1,\ldots,m-1 \\
 & k = m_{ij} & l = m_{i(j+1)} & & k = m_{ij} & l = m_{i(j+1)}
\end{array} \tag{1}$$

Pour son exécution, O_{ij} requiert la présence d'un employé qualifié sur m_{ij}. Nous notons E l'ensemble des μ employés et K_e l'ensemble des machines sur lequel l'employé e peut intervenir. L'atelier fonctionne en 3×8, l'horizon de planification s'étend ainsi sur un ensemble $\{s\}_{s=0,\ldots,\sigma-1}$ de σ tranches horaires consécutives. La durée effective d'une tranche horaire est fixe et notée π pour un horizon $H = \sigma \times \pi$. Le coût d'affectation de l'employé e sur la machine k durant la tranche s est désigné par c_{eks}, étant entendu qu'un employé ne change pas de machine en cours de poste. e est disponible sur un sous-ensemble de tranches horaires $\mathcal{S}_e$ et ne peut travailler qu'au maximum une fois parmi trois tranches horaires consécutives (contrainte légale).

L'objectif du problème est de déterminer un ordonnancement réalisable pour le problème de job-shop en affectant une séquence de couples (machine utilisée - tranche horaire travaillée) à chaque employé, de telle sorte que chaque job soit exécuté avant une date connue $C_{max} \leq H$ et que les besoins en main d'œuvre (effectif et compétence) soient remplis au moindre coût.

Nous proposons ci-dessous une formalisation sous forme de PL-01 de ce problème. Les variables binaires x_{eks} et y_{ikt} valent respectivement 1 si l'opérateur e est affecté à la machine k sur la tranche s et si le job i démarre à l'instant t sur la machine k.

$$[P]: \quad \min \Theta = \sum_{e\in E}\sum_{k\in K_e}\sum_{s\in S_e} c_{eks}\cdot x_{eks} \tag{2}$$

$$\sum_{k\notin K_e}\sum_{s=0}^{\sigma-1} x_{eks} + \sum_{k\in K_e}\sum_{s\notin S_e} x_{eks} = 0 \qquad e=1,\ldots,\mu \tag{3}$$

$$\sum_{k\in K_e}(x_{eks}+x_{ek(s+1)}+x_{ek(s+2)}) \leq 1 \qquad e=1,\ldots,\mu \quad s=0,\ldots,\sigma-3 \tag{4}$$

$$\sum_{t=0}^{d_{ik}-\rho_{ik}} t\cdot y_{ikt} + \rho_{ik} \leq C_{max} \qquad i=1,\ldots,n \quad k=m_{im} \tag{5}$$

$$\sum_{t=r_{ik}}^{d_{ik}-\rho_{ik}} y_{ikt} = 1 \qquad i=1,\ldots,n \quad k=1,\ldots,m \tag{6}$$

$$\sum_{t=0}^{r_{ik}} y_{ikt} + \sum_{t=d_{ik}-\rho_{ik}+1}^{C_{max}} y_{ikt} = 0 \qquad i=1,\ldots,n \quad k=1,\ldots,m \tag{7}$$

$$\sum_{u=r_{ik}+\rho_{ik}}^{t} y_{ilu} - \sum_{u=r_{ik}}^{t-\rho_{ik}} y_{iku} \leq 0 \qquad i=1,\ldots,n \quad j=1,\ldots,m-1 \quad k=m_{ij}$$
$$l=m_{i(j+1)} \quad t=r_{ik}+\rho_{ik},\ldots,d_{il}-\rho_{il} \tag{8}$$

$$\sum_{i=1}^{n}\sum_{u=\max(r_{ik},t-\rho_{ik}+1)}^{\min(d_{ik}-\rho_{ik},t)} y_{iku} \leq 1 \qquad k=1,\ldots,m \quad t=0,\ldots,C_{max} \tag{9}$$

$$\sum_{e\in E} x_{eks} - \sum_{i=1}^{n}\sum_{u=\max(r_{ik},t-\rho_{ik}+1)}^{\min(d_{ik}-\rho_{ik},t)} y_{iku} \geq 0 \quad k=1,\ldots,m \quad t=0,\ldots,C_{max} \quad s=\lfloor t/\pi \rfloor \tag{10}$$

Un employé ne peut être affecté qu'à des machines et des tranches horaires sur lesquelles il est respectivement qualifié et disponible (3). Chaque employé ne peut travailler qu'au maximum une tranche horaire parmi trois tranches consécutives (4). (5) assure que chaque job est terminé avant C_{max}. (6) et (7) spécifient pour leur part que toute opération est exécutée dans sa fenêtre d'exécution. La séquence opératoire propre de chaque job est respectée (8). (9) exprime les contraintes de disponibilité des machines (au plus une opération en cours sur chaque machine à un instant donné). Enfin, les contraintes couplantes (10) assurent la présence d'un employé sur chaque machine et à chaque instant où une opération est en cours d'exécution.

3 Résolution

Nous avons développé une méthode exacte hybridant approche arborescente de type Procédure de Séparation et Evaluation Séquentielle et technique de génération de coupes de réalisabilité. Cette méthode exploite la décomposition naturelle du problème global en deux sous-problèmes : un problème de planification d'agents et un problème de job-shop à contraintes de disponibilité. Des méthodes de génération d'inégalités valides en pré-process (notamment du *probing*) ont en outre été étudiées. Notre approche s'avère particulièrement adaptée à la problématique ; ses résultats dominent en effet ceux obtenus avec l'un des meilleurs solveurs commerciaux actuels (Ilog Cplex 12.1) et avec la méthode décrite dans [AGRV09].

Références

[AGRV09] C. Artigues, M. Gendreau, L.-M. Rousseau, and A. Vergnaud. Solving an integrated employee timetabling and job-shop scheduling problem via hybrid branch-and-bound. *Computers & Operations Research*, 36 :2330–2340, 2009.

[GLPR10] O. Guyon, P. Lemaire, E. Pinson, and D. Rivreau. Solving an integrated job-shop problem with human resource constraints. *Working Paper EMSE CMP-SFL 2010/4*, September 2010.