Planification d'agents et Ordonnancement de production
Règles d'élimination et heuristique

ROADEF'08, 25 - 27 février 2008, Clermont-Ferrand (France)

O. Guyon[1,2], P. Lemaire[2], É. Pinson[1] et D. Rivreau[1]

[1] Institut de Mathématiques Appliquées (UCO)
[2] École des Mines de Nantes

26 février 2008

Sommaire

1. Introduction
   - Problème
   - Formalisation intuitive
   - Formalisation compacte
2. Décomposition et génération de coupes
   - Processus général
   - Coupes initiales énergétiques
3. Conclusion
   - Expérimentations numériques
   - Bilan

Plan

1. Introduction
   - Problème
   - Formalisation intuitive
   - Formalisation compacte
2. Décomposition et génération de coupes
   - Processus général
   - Coupes initiales énergétiques
3. Conclusion
   - Expérimentations numériques
   - Bilan

Présentation du problème

Données

- *J (n jobs)*
  - durée p_j, domaine d'exécution $D_j = [r_j, d_j] \subseteq H$
  - préemptif
  - requiert un opérateur maîtrisant la compétence $c_j \in C$ à chaque instant d'exécution
- *O (m opérateurs)*
  - ensemble de roulements affectables $\Omega_o \subseteq \Omega$ (de coût η_ω^o)
  - ensemble de compétences maîtrisées $C_o \subseteq C$

Objectif

Ordonnancer au moindre coût les n jobs en affectant à chaque opérateur un roulement, tout en satisfaisant les besoins en main d'oeuvre (effectif et compétences)

Références bibliographiques sur le problème

- *Artigues C., Gendreau M. et Rousseau L.-M. (2006)*
  - Etat de l'art sur le problème
  - Méthode hybride (RO/PPC) de résolution
- *Danniels R.L. et Mazzola J.B. (1994)*
  - Durée des jobs variant selon le mode de fabrication
- *Alfares H. et Bailey J. (1997)*
  - Trouver un compromis entre le coût des opérateurs et la durée globale du projet
- ...

Plan

1. Introduction
   - Problème
   - Formalisation intuitive
   - Formalisation compacte
2. Décomposition et génération de coupes
   - Processus général
   - Coupes initiales énergétiques
3. Conclusion
   - Expérimentations numériques
   - Bilan

Variables de décision

Affectation des roulements aux opérateurs

$$\forall o \in O, \forall \omega \in \Omega_o, y_\omega^o = \begin{cases} 1 & \text{si roulement } \omega \text{ affecté à opérateur } o \\ 0 & \text{sinon} \end{cases}$$

Affectation des jobs aux unités de temps

$$\forall j \in J, \forall t \in H, x_{jt} = \begin{cases} 1 & \text{si une unité du job } j \text{ est exécutée à } t \\ 0 & \text{sinon} \end{cases}$$

Affectation des compétences aux unités de temps

$$\forall o \in O, \forall c \in C, \forall t \in H, z_{oct} = \begin{cases} 1 & \text{si l'opérateur } o \text{ utilise } c \text{ à } t \\ 0 & \text{sinon} \end{cases}$$

Programme linéaire

$$
\begin{array}{rll}
[Q] : \min & \displaystyle\sum_{o \in O} \sum_{\omega \in \Omega_o} \eta_\omega^o y_\omega^o & \\
\forall o \in O & \displaystyle\sum_{\omega \in \Omega_o} y_\omega^o & = 1 \\
\forall j \in J & \displaystyle\sum_{t \in D_j} x_{jt} & = p_j \\
\forall t \in H, \forall c \in C & \displaystyle\sum_{j \in J / c_j = c} x_{jt} & = \displaystyle\sum_{o \in O / c \in C_o} z_{oct} \\
\forall t \in H, \forall o \in O & \displaystyle\sum_{c \in C_o} z_{oct} & \leq \displaystyle\sum_{\omega \in \Omega_o} \sigma_\omega^t \cdot y_\omega^o \\
\forall o \in O, \forall \omega \in \Omega_o & y_\omega^o \in \{0,1\} & \\
\forall j \in J, \forall t \in H & x_{jt} \in \{0,1\} & \\
\forall o \in O, \forall c \in C_o, \forall t \in H & z_{oct} \in \{0,1\} &
\end{array}
$$

avec : $\forall \omega \in \Omega, \forall t \in H, \sigma_\omega^t = 1$ ssi ω couvre t

Plan

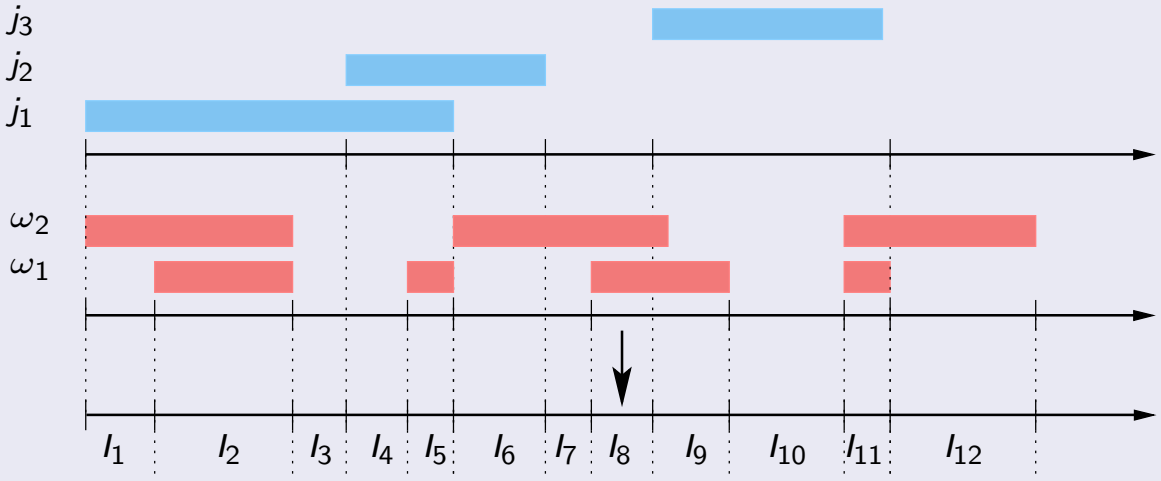
1. Introduction
   - Problème
   - Formalisation intuitive
   - Formalisation compacte
2. Décomposition et génération de coupes
   - Processus général
   - Coupes initiales énergétiques
3. Conclusion
   - Expérimentations numériques
   - Bilan

Formalisation compacte - Intervalles de temps

Constat

Tous les instants de temps n'influent pas sur la solution

Illustration



Nouvelles variables de décision

$\forall j \in J, \forall k \in K \quad x_{jk} \in [0, \min(p_j, l_k)]$ où $l_k =$ longueur de $\mathcal{I}_k$

Formalisation compacte - Profils de compétence

Constat

$\forall o \in O, \forall c \in C, \forall t \in H \quad z_{oct} \notin$ fonction objectif de [Q]

Conséquence

Inutile de différencier opérateurs ayant les mêmes compétences

Nouvelles variables de décision

- Création de profils de compétence $\Theta \subseteq \mathcal{P}(C)$
- $\forall o \in O \rightarrow$ un unique profil $\theta_o \in \Theta$
- $\theta_{o_1} = \theta_{o_2} \Leftrightarrow C_{o_1} = C_{o_2}$
- $\forall \theta \in \Theta, \forall c \in \theta, \forall k \in K \quad z_{\theta ck} \in [0, l_k \cdot |\{o \in O/_{\theta_o = \theta}\}|]$

Programme linéaire

$$
\begin{array}{lll}
[P] : \min & \displaystyle\sum_{o \in O} \sum_{\omega \in \Omega_o} \eta_\omega^o y_\omega^o & \\
\forall o \in O & \displaystyle\sum_{\omega \in \Omega_o} y_\omega^o & = 1 \\
\forall j \in J & \displaystyle\sum_{k \in K / \mathcal{I}_k \subseteq D_j} x_{jk} & = p_j \\
\forall k \in K, \forall c \in C & \displaystyle\sum_{j \in J / c_j = c} x_{jk} & = \displaystyle\sum_{\theta \in \Theta / c \in \theta} z_{\theta ck} \\
\forall k \in K, \forall \theta \in \Theta & \displaystyle\sum_{c \in \theta} z_{\theta ck} & \leq \displaystyle\sum_{o \in O / \theta_o = \theta} \sum_{\omega \in \Omega_o / \mathcal{I}_k \subseteq \omega} l_k \cdot y_\omega^o \\
\forall o \in O, \forall \omega \in \Omega_o & & y_\omega^o \in \{0, 1\} \\
\forall j \in J, \forall k \in K & & x_{jk} \in [0, \min(p_j, l_k)] \\
\forall \theta \in \Theta, \forall c \in \theta, \forall k \in K & & z_{\theta ck} \in [0, l_k \cdot |\{o \in O / \theta_o = \theta\}|]
\end{array}
$$

Techniques de résolution

Obtenir une borne supérieure réalisable le plus rapidement possible

- MIP
- Décomposition et génération de coupes (2 versions)

Plan

1. Introduction
   - Problème
   - Formalisation intuitive
   - Formalisation compacte
2. Décomposition et génération de coupes
   - Processus général
   - Coupes initiales énergétiques
3. Conclusion
   - Expérimentations numériques
   - Bilan

Etapes

- Décomposition de $[P] \rightarrow [MP]$ et $[SP]$
- Résolution de $[MP] \hookrightarrow \bar{y}$: roulement $\rightarrow$ opérateur
- Résolution de $[SP(\bar{y})]$: $\bar{y}$ est réalisable ?
  - Jobs exécutables sur leur fenêtre temporelle d'exécution ?
  - Affectation réalisable de compétences aux opérateurs ?
- Si $[SP(\bar{y})]$ parvient à planifier tous les jobs :
  - $\bar{y} \Rightarrow$ solution réalisable
  - Génération d'une coupe de borne invalidant, dans $[MP]$, les solutions de coût supérieur à celui de $\bar{y}$
- Sinon : Génération d'une coupe invalidant $\bar{y}$ dans $[MP]$
- Itérer jusqu'à ce que :
  - $[MP]$ n'ait plus de solution (optimum)
  - limite de temps dépassée (borne supérieure réalisable)

Problème maître [MP]

$$[MP] : \min \sum_{o \in O} \sum_{\omega \in \Omega_o} \eta_\omega^o y_\omega^o$$

$$\forall o \in O \quad \sum_{\omega \in \Omega_o} y_\omega^o = 1$$

$$Cut$$

$$\forall o \in O, \forall \omega \in \Omega_o \quad y_\omega^o \in \{0, 1\}$$

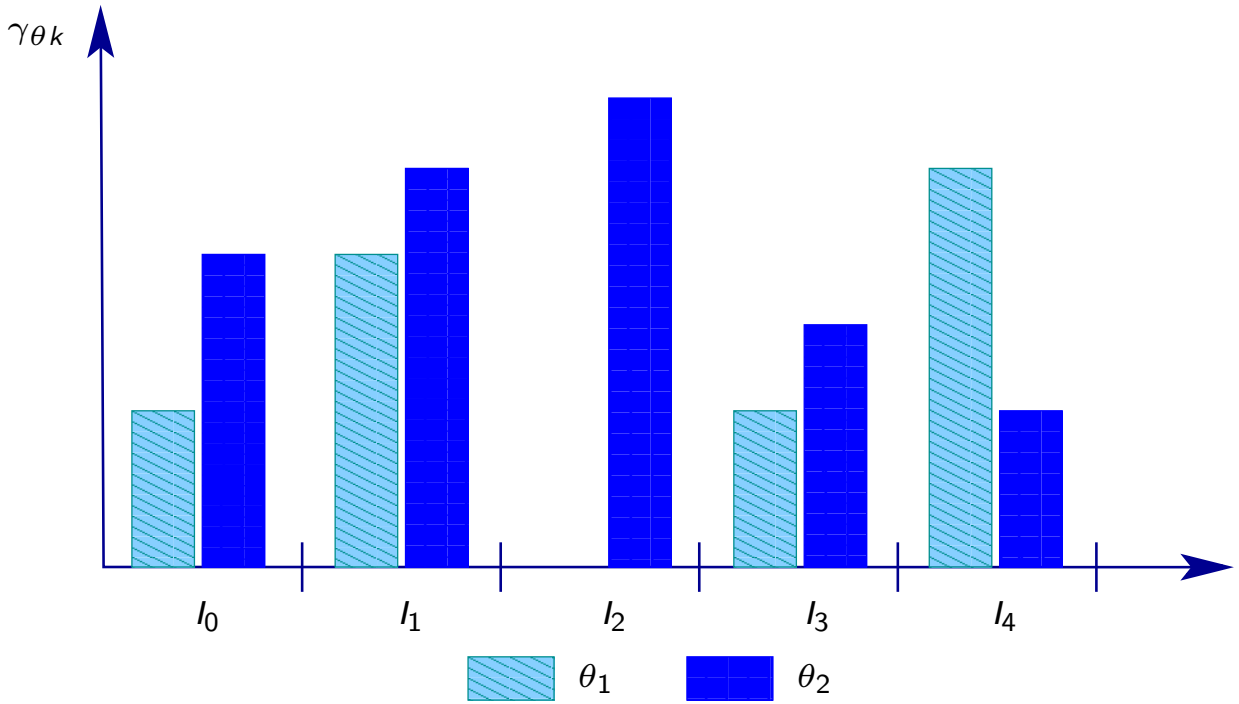
Résolution de [MP]

Affectation des roulements aux opérateurs $\rightsquigarrow \bar{y}$

Courbe de disponibilité des opérateurs - compétences

Courbe de disponibilité des opérateurs - compétences

$$\forall \theta \in \Theta, \forall k \in K \quad \gamma_{\theta k} = \sum_{b \in \Theta K^+} \sum_{\substack{o \in O / \\ \theta_o = \theta_b}} \sum_{\substack{\omega \in \Omega_o / \\ \mathcal{I}_{k_b} \subseteq \omega}} l_{k_b} \cdot \bar{y}_\omega^o$$



Sous-problème [SP($\bar{y}$)]

$$[SP(\bar{y})] : \quad \max \quad f_{\bar{y}} = \sum_{j \in J} \sum_{k \in K / \mathcal{I}_k \subseteq D_j} x_{jk}$$

$$\forall j \in J \quad \sum_{k \in K / \mathcal{I}_k \subseteq D_j} x_{jk} \leq p_j$$

$$\forall k \in K, \forall c \in C \quad \sum_{j \in J / c_j = c} x_{jk} = \sum_{\theta \in \Theta / c \in \theta} z_{\theta ck}$$

$$\forall k \in K, \forall \theta \in \Theta \quad \sum_{c \in \theta} z_{\theta ck} \leq \sum_{o \in O / \theta_o = \theta} \sum_{\omega \in \Omega_o / \mathcal{I}_k \subseteq \omega} l_k \cdot \bar{y}^o_\omega$$

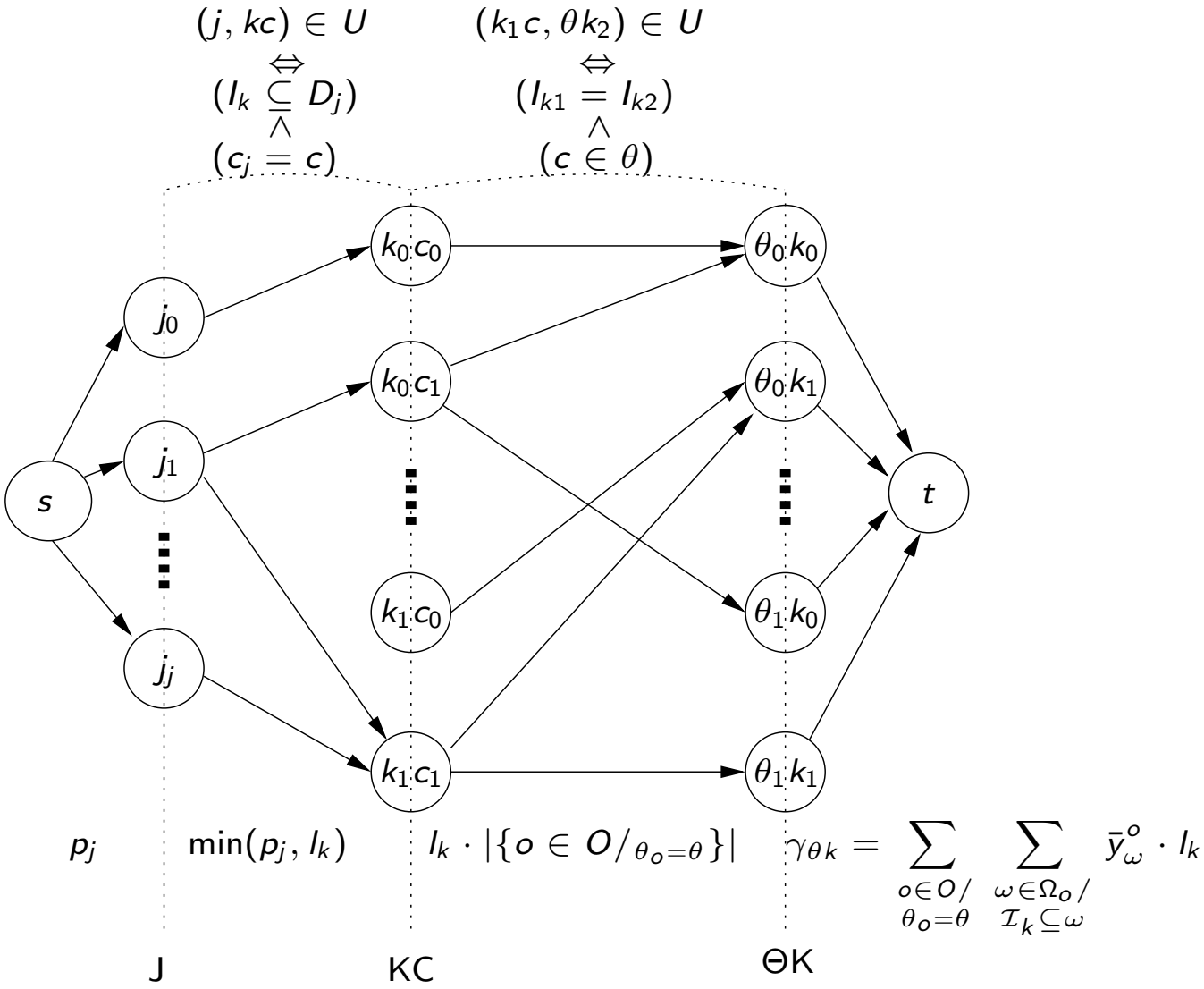
$$\forall j \in J, \forall k \in K \quad x_{jk} \in [0, \min(p_j, l_k)]$$

$$\forall \theta \in \Theta, \forall c \in \theta, \forall k \in K \quad z_{\theta ck} \in [0, l_k \cdot |\{o \in O / \theta_o = \theta\}|]$$

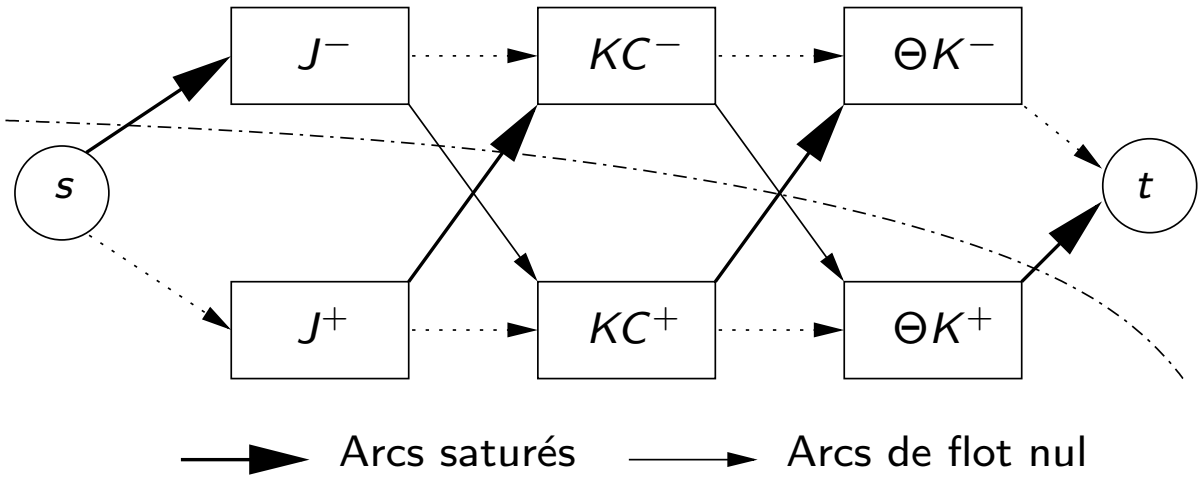
Résolution de [SP($\bar{y}$)]

Flot maximal sur un graphe $G_{\bar{y}} = (X, U)$ décomposable en niveaux

Définition du graphe de flot $G_{\bar{y}}$



Coupe minimale sur $G_{\bar{y}}$



$$F^- = \sum_{b \in \Theta K^-} flot(u_{bt}) = \sum_{j \in J^-} p_j + \sum_{j \in J^+} \sum_{a \in KC^-} \min(p_j, l_{k_a}) + \sum_{a \in KC^+} \sum_{b \in \Theta K^-} l_{k_b} \cdot |\{o \in O / \theta_o = \theta_b\}|$$

Nécessairement, $F^+ = \sum_{b \in \Theta K^+} flot(u_{bt}) \geq \sum_{j \in J} p_j - F^-$

Génération de coupes

Expression de la coupe

$$F^+ \geq \sum_{j \in J} p_j - F^-$$

$$\Leftrightarrow \sum_{b \in \Theta K^+} \sum_{\substack{o \in O/ \\ \theta_o = \theta_b}} \sum_{\substack{\omega \in \Omega_o/ \\ \mathcal{I}_{k_b} \subseteq \omega}} y_\omega^o \cdot l_{k_b} \geq \nu$$

avec

$$\nu = \sum_{j \in J^+} p_j - \sum_{j \in J^+} \sum_{a \in KC^-} \min(p_j, l_{k_a}) - \sum_{a \in KC^+} \sum_{b \in \Theta K^-} l_{k_b} \cdot |\{o \in O/_{\theta_o = \theta_b}\}|$$

$[MP^r] \rightarrow$ [MP] avec r coupes

$$
\begin{array}{lll}
[MP^r] : \min & \displaystyle\sum_{o \in O} \sum_{\omega \in \Omega_o} \eta_\omega^o y_\omega^o & \\
\forall o \in O & \displaystyle\sum_{\omega \in \Omega_o} y_\omega^o & = 1 \\
\forall i \in Q_r & \displaystyle\sum_{b \in \Theta K_i^+} \sum_{\substack{o \in O / \\ \theta_o = \theta_b}} \sum_{\substack{\omega \in \Omega_o / \\ \mathcal{I}_{k_b} \subseteq \omega}} y_\omega^o \cdot l_{k_b} & \geq \nu_i \\
\forall o \in O, \forall \omega \in \Omega_o & y_\omega^o \in \{0, 1\} &
\end{array}
$$

Caractérisation de $[MP^r]$

Sac à Dos Multichoix Multidimensionnel

Quelle stratégie pour résoudre $[MP^r]$?

Résolution optimale

- À chaque itération : affectation $\bar{y}$ de moindre coût
- Optimum dès la 1ère affectation *réalisable*
- + : moins d'itérations / - : tps de résolution par itération
- Comment ? MIP

Résolution réalisable

- À chaque itération : affectation $\bar{y}$ réalisable (pour $[MP]$)
- + : tps par itération, solution réalisable rapide
- - : plus de contraintes (contraintes de borne), plus d'itérations
- Comment ? MIP avec arrêt 1ère solution ou *Feasibility Pump*

Feasibility Pump

Caractérisation

- S'applique à tout problème *MIP* :

$$\min\{c^T x : Ax \geq b, \quad x_j \in \mathbb{N} \quad \forall j \in \mathcal{I}\}$$

- But : Trouver rapidement une solution entière réalisable
- Principe : Générer deux trajectoires (avec espoir *convergentes*) x^* et $\tilde{x}$ partiellement réalisables
  - x^* respecte les contraintes linéaires
  - $\tilde{x}$ respecte les contraintes d'intégralité

Références bibliographiques

- *Fischetti M., Glover F. et Lodi A. (2005)*
- *Bertacco L., Fischetti M. et Lodi A. (2005)*
- *Achterberg T. et Berthold T. (2006)*

Feasibility Pump

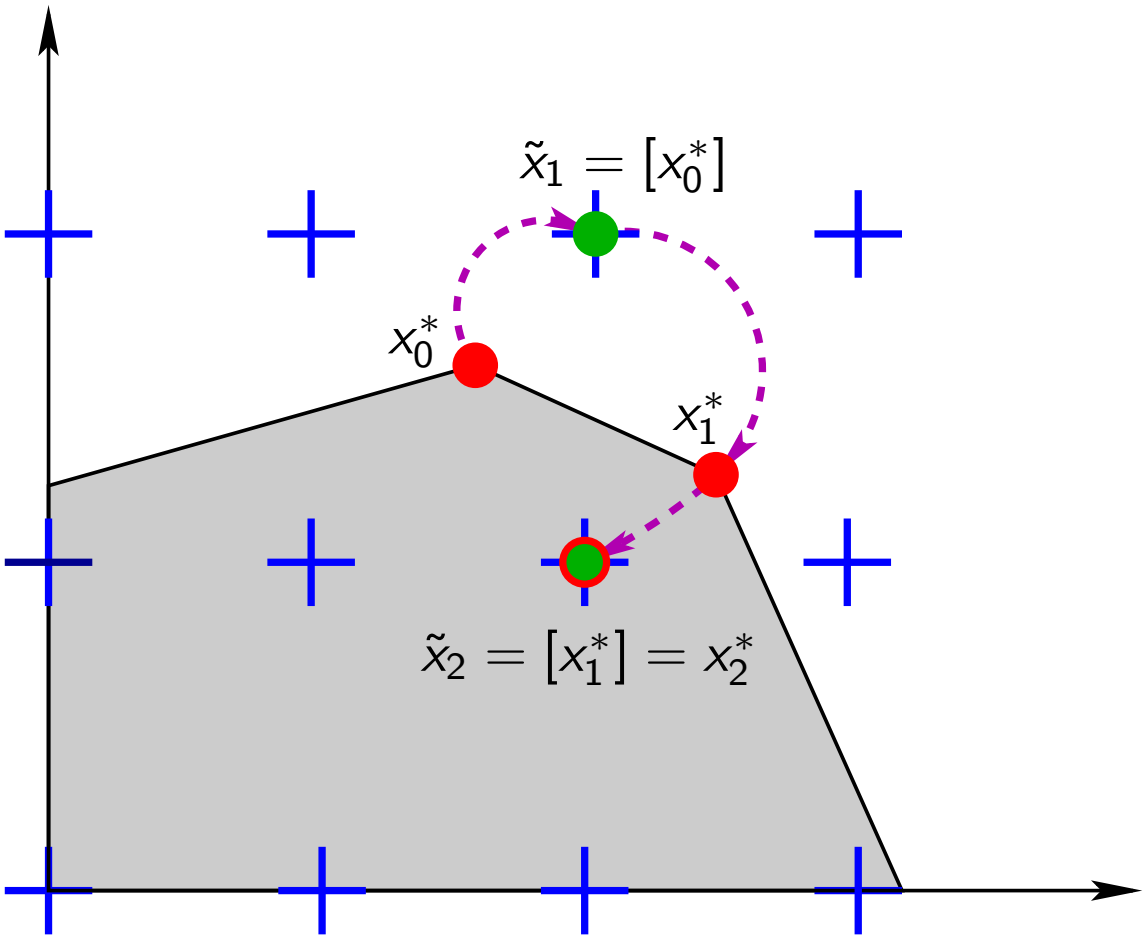
Caractérisation

- S'applique à tout problème *MIP* :

$$\min\{c^T x : Ax \geq b, \quad x_j \in \mathbb{N} \quad \forall j \in \mathcal{I}\}$$

- But : Trouver une solution entière réalisable le plus rapidement possible
- Principe : Générer deux trajectoires (avec espoir *convergentes*) x^* et $\tilde{x}$ partiellement réalisables
  - x^* respecte les contraintes linéaires
  - $\tilde{x}$ respecte les contraintes d'intégralité

Illustration



Précisions

- Première solution ?
  Solution continue réalisable (relaxation continue)
- Passage de x_i^* à $\tilde{x}_{i+1}$?

  Arrondi : $\tilde{x}_{i+1} = [x_i^*] \Leftrightarrow \forall j \in \mathcal{I} \quad (\tilde{x}_{i+1})_j = \lfloor (x_i^*)_j + 0.5 \rfloor$
- Passage de $\tilde{x}_i$ à x_i^* ?
  Minimisation de la *distance d'infaisabilité*

  $\tilde{x}_i = \mathrm{argmin}\{\Delta(x, \tilde{x}_i)/Ax \geq b\}$ avec : $\Delta(x, \tilde{x}_i) = \sum_{j \in \mathcal{I}} |x_j - \tilde{x}_j|$
- Cycle possible ! $\Rightarrow$ Processus de perturbation (*flip* et *restart*)
- En cas d'échec ? Résolution du MIP original avec :
  - contraintes d'intégralité
  - fonction objectif : $\Delta(x, \tilde{x}_n)$ avec $\tilde{x}_n$ le point "*le plus réalisable*"
  - limitation du solveur à stopper dès la $1^{\text{ère}}$ solution réalisable

Plan

1. Introduction
   - Problème
   - Formalisation intuitive
   - Formalisation compacte
2. Décomposition et génération de coupes
   - Processus général
   - Coupes initiales énergétiques
3. Conclusion
   - Expérimentations numériques
   - Bilan

Coupes initiales énergétiques

Accélérer la vitesse de convergence du processus ?

- Idée : Initialiser l'ensemble *Cut* de [*MP*]
- Action : Utiliser le raisonnement énergétique

Références bibliographiques sur le raisonnement énergétique

- *Lopez P., Erschler J. et Esquirol P. (1992)*
  Ordonnancement de tâches sous contraintes
- *Baptiste P., Pape C.L. et Nujiten W. (1999)*
  Problème d'ordonnancement cumulatif
  (ressources peuvent exécuter plusieurs tâches en même temps)

Raisonnement énergétique

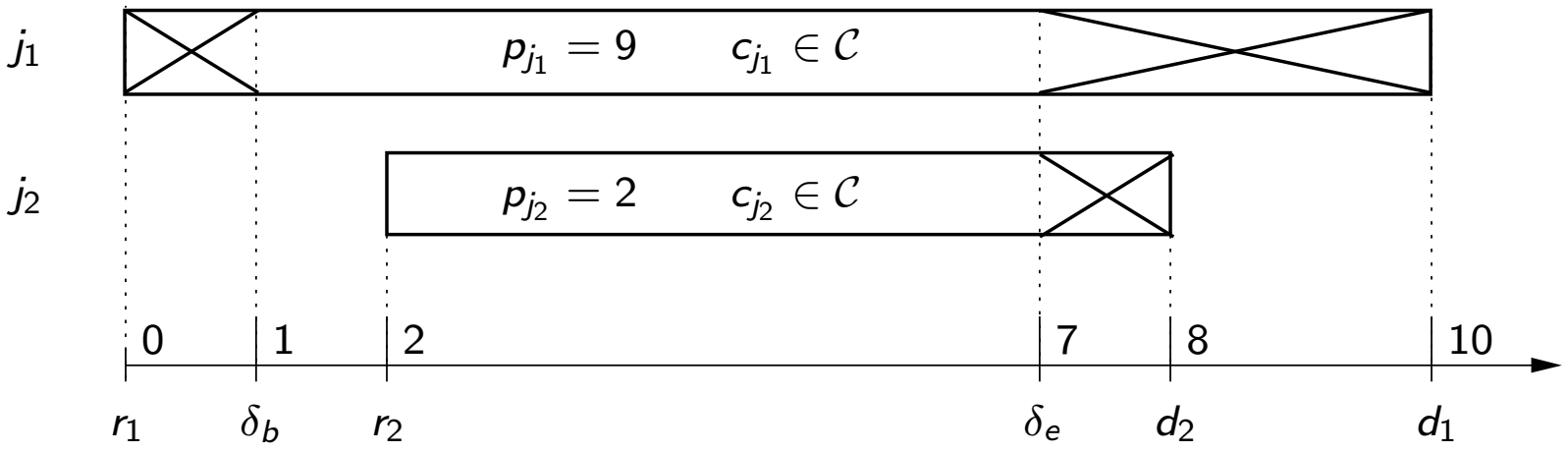
Vision énergétique du problème

- compétences $\longrightarrow$ *ressources consommables*
- opérateurs $\longrightarrow$ *fournisseurs de ressources*
- jobs $\longrightarrow$ *consommateurs de ressources*

Principe

Définir des périodes de temps $[\delta_b, \delta_e]$ où il existe une consommation minimale obligatoire pour un sous-ensemble de compétences $\mathcal{C} \subseteq C$

Illustration : coupe initiale sur $[1, 7]$ pour $\mathcal{C}$?



- Job j_1 : 5 unités effectuables en dehors de $[1, 7]$
  $\Rightarrow 9 - 5 = 4$ unités à exécuter sur $[1, 7]$
- Job j_2 : 1 unité effectuable en dehors de $[1, 7]$
  $\Rightarrow 2 - 1 = 1$ unité à exécuter sur $[1, 7]$
- $\hookrightarrow \exists$ consommation obligatoire en compétences de $\mathcal{C}$ sur $[1, 7]$

Caractérisation d'une coupe initiale énergétique sur une période $[\delta_b, \delta_e]$ et un sous-ensemble de compétences $\mathcal{C}$

Coupe initiale énergétique pour $\mathcal{C}$ sur $[\delta_b, \delta_e]$

$$\sum_{t \in \delta} \sum_{\substack{o \in O / \\ \{C_o \cap \mathcal{C} \neq \emptyset\}}} \sum_{\omega \in \Omega_o} \sigma_\omega^t y_\omega^o \geq \sum_{c \in \mathcal{C}} \sum_{\substack{j \in J / \\ c_j = c}} \left(p_j - (\delta_b - r_j)^+ - (d_j - \delta_e)^+\right)^+$$

$$\text{avec } (a)^+ = \max(a, 0) \quad \forall a \in \Re$$

Sélection expérimentale des coupes initiales

Précautions d'usage ⇒ Sélection des coupes initiales obligatoire

- $\exists$ grand nb de périodes de temps (non forcément connexes)
- $2^{|C|}$ sous-ensembles de compétences

Ensembles de compétences considérés

- Ensemble Θ des profils de compétences
- Singletons (une seule compétence)

Filtrage par dominance

- Ajout de toutes les coupes générées à l'ensemble *Cut* de $[MP]$
- Résolution de la relaxation continue de $[MP]$
- Tri des coupes selon leur *slack* à l'optimum
- Sélection des 15 premiers %

Plan

1. Introduction
   - Problème
   - Formalisation intuitive
   - Formalisation compacte
2. Décomposition et génération de coupes
   - Processus général
   - Coupes initiales énergétiques
3. Conclusion
   - Expérimentations numériques
   - Bilan

Générateur de données

- Moduler le taux de travail sur l'horizon
- Dispersion des fenêtres d'exécution des jobs
- Moduler le nombre de roulements et de compétences affectables à chaque opérateur
- Moduler le coût d'affectation d'un roulement à un opérateur selon les heures travaillées, l'expérience ...

Conditions

270 instances - (3 par jeu de paramètres) -

Paramètre	min	max	pas
$\|O\|$	15	25	10
$\|J\|$	$4 \cdot \|O\|$	$6 \cdot \|O\|$	$\|O\|$
$marge_{max}$	30	90	30
$\|C\|$	1	5	1
p_{max}	30		
$\|H\|$	480		

Outils

- Langage de développement : Java
- Solveur PLNE : Ilog Cplex 9.1
- Processeur Intel Pentium D 930 ; 3 GHz ; 2 GB RAM

Résultats : de l'intérêt de la formalisation ...

Instances résolues en moins de 5 minutes

	MIP $\hookrightarrow [Q]$	MIP $\hookrightarrow [P]$
% d'instances résolues	26.9%	59.4%
Temps moyen de résolution	125.6s	13.2s

Résultats : de l'intérêt des coupes initiales ...

Instances résolues en moins de 5 minutes

	MIP	MIP avec coupes initiales
% d'instances résolues	59.4%	72.04%
Temps moyen de résolution	52.7s	21.4s

Résultats : de l'intérêt de la Feasibility Pump ...

Déviation moyenne à la meilleure borne supérieure connue (optimum pour 80% des instances)

		Génération de coupes	
Tps limite	MIP	$[MP] \hookrightarrow$ MIP	$[MP] \hookrightarrow$ Feasibility Pump
1s	71.68%	85.50%	94.51%
5s	87.51%	94.22%	98.69%
10s	92.95%	96.45%	99.28%
30s	98.06%	98.74%	99.72%
60s	99.12%	99.33%	99.82%
300s	99.92%	99.79%	99.95%

Plan

1. Introduction
   - Problème
   - Formalisation intuitive
   - Formalisation compacte
2. Décomposition et génération de coupes
   - Processus général
   - Coupes initiales énergétiques
3. Conclusion
   - Expérimentations numériques
   - Bilan

Bilan

- Formalisation compacte performante
- Coupes initiales énergétiques intéressantes
- Approche par décomposition et génération de coupes très avantageuse
- Résolution de $[MP]$ par Feasibility Pump compétitive