

Couplage Planification et Ordonnancement: Approche hiérarchique et décomposition

Soutenance de thèse de doctorat d'informatique

Olivier Guyon

Directeurs

Éric Pinson

Pierre Lemaire & David Rivreau

Rapporteurs

Philippe Chrétienne

Stéphane Dauzère-Pères

Examineurs

Vincent Barichard

Nadia Brauner

19 mai 2010



- ① Contexte et enjeux
- ② Planification du personnel contrainte par la réalisabilité de la production
- ③ Problème de job-shop intégrant des contraintes de ressources humaines
- ④ Conclusion et prolongements
- ⑤ Production scientifique

Partie I

Contexte et enjeux

I Contexte et enjeux

Contexte

Contexte physique

Contexte physique

- Systèmes manufacturiers

Contexte physique

- Systèmes manufacturiers
- Deux piliers organisationnels (postes de décision)

Contexte physique

- Systèmes manufacturiers
- Deux piliers organisationnels (postes de décision)
 - Planification d'emplois du temps des employés

Contexte physique

- Systèmes manufacturiers
- Deux piliers organisationnels (postes de décision)
 - Planification d'emplois du temps des employés
 - Ordonnancement de la production

Contexte physique

- Systèmes manufacturiers
- Deux piliers organisationnels (postes de décision)
 - Planification d'emplois du temps des employés
 - Ordonnancement de la production
- En pratique, processus de décision réalisé en deux temps

clairement sous-optimal

Contexte physique

- Systèmes manufacturiers
- Deux piliers organisationnels (postes de décision)
 - Planification d'emplois du temps des employés
 - Ordonnancement de la production
- En pratique, processus de décision réalisé en deux temps

clairement sous-optimal

Contexte scientifique : intégration de ces deux niveaux de décision

Contexte physique

- Systèmes manufacturiers
- Deux piliers organisationnels (postes de décision)
 - Planification d'emplois du temps des employés
 - Ordonnancement de la production
- En pratique, processus de décision réalisé en deux temps

clairement sous-optimal

Contexte scientifique : intégration de ces deux niveaux de décision

- Littérature dense dans le domaine du transport
⇒ Fortes économies réalisées

Contexte physique

- Systèmes manufacturiers
- Deux piliers organisationnels (postes de décision)
 - Planification d'emplois du temps des employés
 - Ordonnancement de la production
- En pratique, processus de décision réalisé en deux temps

clairement sous-optimal

Contexte scientifique : intégration de ces deux niveaux de décision

- Littérature dense dans le domaine du transport
 - ⇒ Fortes économies réalisées
- Littérature relativement pauvre dans le domaine manufacturier

Objectif de la thèse

Proposer des méthodes pour résoudre des problématiques, dans le milieu industriel, couplant les deux niveaux de décision :

- Planification d'emplois du temps des employés
- Ordonnancement de la production

Techniques de résolution à expérimenter

Techniques de résolution à expérimenter

- **Méthodes exactes** (principalement)

Techniques de résolution à expérimenter

- **Méthodes exactes** (principalement)
- Tester l'intérêt d'**exploiter la structure décomposable** intuitive des problèmes

Techniques de résolution à expérimenter

- **Méthodes exactes** (principalement)
- Tester l'intérêt d'**exploiter la structure décomposable** intuitive des problèmes
- **Approches par décomposition**, avec interaction entre les niveaux de décision au travers de coupes de réalisabilité

Techniques de résolution à expérimenter

- **Méthodes exactes** (principalement)
- Tester l'intérêt d'**exploiter la structure décomposable** intuitive des problèmes
- **Approches par décomposition**, avec interaction entre les niveaux de décision au travers de coupes de réalisabilité
- *Pourquoi ?* Démarche similaire déjà appliquée avec succès par Detienne et al. pour un problème de planification d'agents (sans décision sur l'ordonnancement de la production)

Partie II

Planification du personnel contrainte par la réalisabilité de la production

II Planification du personnel contrainte par la réalisabilité de la production

- 1 Introduction
 - Problème
 - Formalisation MIP indexée sur le temps
 - Formalisation MIP compacte
 - Techniques de résolution
- 2 Décomposition et génération de coupes
 - Processus général
 - Résolution du problème maître
 - Coupes initiales énergétiques
- 3 Expérimentations numériques
- 4 Conclusion
 - Analyse et prolongements
 - Production scientifique

II Planification du personnel contrainte par la réalisabilité de la production

- 1 Introduction
 - Problème
 - Formalisation MIP indexée sur le temps
 - Formalisation MIP compacte
 - Techniques de résolution
- 2 Décomposition et génération de coupes
- 3 Expérimentations numériques
- 4 Conclusion

Présentation du problème

Données

Présentation du problème

Données

- J (n jobs)

Présentation du problème

Données

- J (n jobs)
 - durée p_j , domaine d'exécution $D_j = [r_j, d_j] \subseteq H$

Présentation du problème

Données

- J (n jobs)
 - durée p_j , domaine d'exécution $D_j = [r_j, d_j] \subseteq H$
 - préemption autorisée

Présentation du problème

Données

- J (n jobs)
 - durée p_j , domaine d'exécution $D_j = [r_j, d_j] \subseteq H$
 - préemption autorisée
 - requiert un opérateur maîtrisant la compétence $c_j \in C$ à chaque instant d'exécution

Présentation du problème

Données

- J (n jobs)
 - durée p_j , domaine d'exécution $D_j = [r_j, d_j] \subseteq H$
 - préemption autorisée
 - requiert un opérateur maîtrisant la compétence $c_j \in C$ à chaque instant d'exécution
- O (m opérateurs)

Présentation du problème

Données

- J (n jobs)
 - durée p_j , domaine d'exécution $D_j = [r_j, d_j] \subseteq H$
 - préemption autorisée
 - 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)
↪ Roulement : canevas d'horaires de présence/absence

Présentation du problème

Données

- J (n jobs)
 - durée p_j , domaine d'exécution $D_j = [r_j, d_j] \subseteq H$
 - préemption autorisée
 - 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)
↪ Roulement : canevas d'horaires de présence/absence
 - ensemble de compétences maîtrisées $C_o \subseteq C$

Présentation du problème

Données

- J (n jobs)
 - durée p_j , domaine d'exécution $D_j = [r_j, d_j] \subseteq H$
 - préemption autorisée
 - 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)
 \hookrightarrow Roulement : canevas d'horaires de présence/absence
 - 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'œuvre (effectif et compétences)

Références bibliographiques sur le problème

- *Artigues C., Gendreau M. et Rousseau L.-M. (2006)*

Références bibliographiques sur le problème

- *Artigues C., Gendreau M. et Rousseau L.-M. (2006)*
 - État de l'art sur le problème

Références bibliographiques sur le problème

- *Artigues C., Gendreau M. et Rousseau L.-M. (2006)*
 - État de l'art sur le problème
 - Méthode hybride (RO/PPC) de résolution

Références bibliographiques sur le problème

- *Artigues C., Gendreau M. et Rousseau L.-M. (2006)*
 - État de l'art sur le problème
 - Méthode hybride (RO/PPC) de résolution
- *Danniels R.L. et Mazzola J.B. (1994)*

Références bibliographiques sur le problème

- *Artigues C., Gendreau M. et Rousseau L.-M. (2006)*
 - État de l'art sur le problème
 - Méthode hybride (RO/PPC) de résolution
- *Daniels R.L. et Mazzola J.B. (1994)*
 - Durée des jobs variant selon le mode de fabrication

Références bibliographiques sur le problème

- *Artigues C., Gendreau M. et Rousseau L.-M. (2006)*
 - État 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)*

Références bibliographiques sur le problème

- *Artigues C., Gendreau M. et Rousseau L.-M. (2006)*
 - État 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

II Planification du personnel contrainte par la réalisabilité de la production

- 1 Introduction
 - Problème
 - Formalisation MIP indexée sur le temps
 - Formalisation MIP compacte
 - Techniques de résolution
- 2 Décomposition et génération de coupes
- 3 Expérimentations numériques
- 4 Conclusion

Variables de décision

Affectation des roulements aux opérateurs

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

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

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

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

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

Programme linéaire

$$[Q] : \min \sum_{o \in O} \sum_{\omega \in \Omega_o} \eta_{\omega}^o \cdot y_{\omega}^o$$
$$\forall o \in O \quad \sum_{\omega \in \Omega_o} y_{\omega}^o = 1$$

Programme linéaire

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

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

$$\forall j \in J \quad \sum_{t \in D_j} x_{jt} = p_j$$

Programme linéaire

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

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

$$\forall j \in J \quad \sum_{t \in D_j} x_{jt} = p_j$$

$$\forall t \in H, \forall c \in C \quad \sum_{j \in J / c_j = c} x_{jt} = \sum_{o \in O / c \in C_o} z_{oct}$$

Programme linéaire

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

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

$$\forall j \in J \quad \sum_{t \in D_j} x_{jt} = p_j$$

$$\forall t \in H, \forall c \in C \quad \sum_{j \in J / c_j = c} x_{jt} = \sum_{o \in O / c \in C_o} z_{oct}$$

$$\forall t \in H, \forall o \in O \quad \sum_{c \in C_o} z_{oct} \leq \sum_{\omega \in \Omega_o} \sigma_{\omega}^t \cdot y_{\omega}^o$$

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

Programme linéaire

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

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

$$\forall j \in J \quad \sum_{t \in D_j} x_{jt} = p_j$$

$$\forall t \in H, \forall c \in C \quad \sum_{j \in J / c_j = c} x_{jt} = \sum_{o \in O / c \in C_o} z_{oct}$$

$$\forall t \in H, \forall o \in O \quad \sum_{c \in C_o} z_{oct} \leq \sum_{\omega \in \Omega_o} \sigma_{\omega}^t \cdot y_{\omega}^o$$

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

$$\forall j \in J, \forall t \in H \quad x_{jt} \in \{0, 1\}$$

$$\forall o \in O, \forall c \in C_o, \forall t \in H \quad z_{oct} \in \{0, 1\}$$

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

II Planification du personnel contrainte par la réalisabilité de la production

- 1 Introduction
 - Problème
 - Formalisation MIP indexée sur le temps
 - **Formalisation MIP compacte**
 - Techniques de résolution
- 2 Décomposition et génération de coupes
- 3 Expérimentations numériques
- 4 Conclusion

Formalisation MIP compacte - Intervalles de temps

Constat

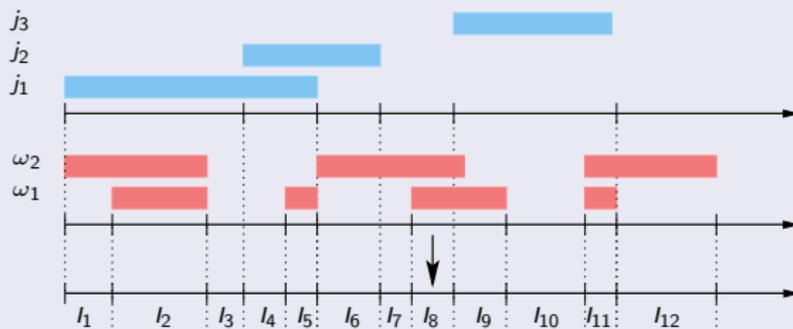
Tous les instants n'influent pas sur la solution

Formalisation MIP compacte - Intervalles de temps

Constat

Tous les instants n'influent pas sur la solution

Illustration

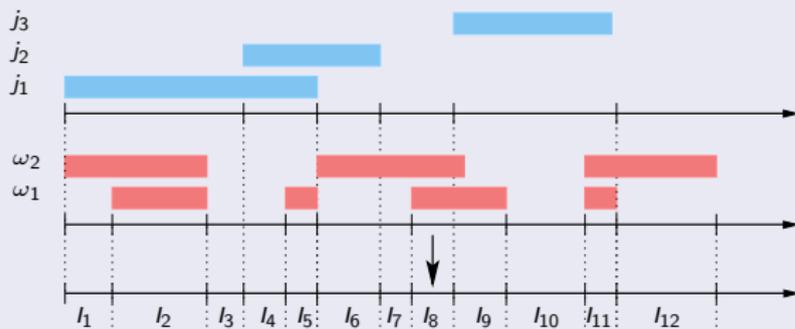


Formalisation MIP compacte - Intervalles de temps

Constat

Tous les instants 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 MIP compacte - Profils de compétence

Constat

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

Formalisation MIP compacte - Profils de compétence

Constat

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

Conséquence

Inutile de différencier opérateurs ayant les mêmes compétences

Formalisation MIP 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

Formalisation MIP compacte - Profils de compétence

Constat

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

Conséquence

Inutile de différencier opérateurs ayant les mêmes compétences

Nouvelles variables de décision

- Création de profils de compétence $\Theta \subseteq \mathcal{P}(C)$ ($\text{card } \{\Theta\} \leq m$)

Formalisation MIP 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 $\Theta \subseteq \mathcal{P}(C)$ ($\text{card} \{\Theta\} \leq m$)
- $\forall o \in O \rightarrow$ un unique profil $\theta_o \in \Theta$

Formalisation MIP compacte - Profils de compétence

Constat

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

Conséquence

Inutile de différencier opérateurs ayant les mêmes compétences

Nouvelles variables de décision

- Création de profils de compétence $\Theta \subseteq \mathcal{P}(C)$ ($\text{card } \{\Theta\} \leq m$)
- $\forall o \in O \rightarrow$ un unique profil $\theta_o \in \Theta$
- $\theta_{o_1} = \theta_{o_2} \Leftrightarrow C_{o_1} = C_{o_2}$

Formalisation MIP 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 $\Theta \subseteq \mathcal{P}(C)$ ($\text{card} \{\Theta\} \leq m$)
- $\forall o \in O \rightarrow$ 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

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

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

$$\forall j \in J \quad \sum_{k \in K / I_k \subseteq D_j} x_{jk} = p_j$$

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

$$\forall k \in K, \forall \theta \in \Theta \quad \sum_{c \in \theta} z_{\theta ck} \leq \sum_{o \in O / \theta_o = \theta} \sum_{\omega \in \Omega_o / I_k \subseteq \omega} I_k \cdot y_{\omega}^o$$

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

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

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

II Planification du personnel contrainte par la réalisabilité de la production

- 1 Introduction
 - Problème
 - Formalisation MIP indexée sur le temps
 - Formalisation MIP compacte
 - Techniques de résolution
- 2 Décomposition et génération de coupes
- 3 Expérimentations numériques
- 4 Conclusion

Techniques de résolution

Bornes inférieures

Techniques de résolution

Bornes inférieures

- Relaxation lagrangienne *totale*

Techniques de résolution

Bornes inférieures

- Relaxation lagrangienne *totale*
- Relaxation lagrangienne *partielle*

Techniques de résolution

Bornes inférieures

- Relaxation lagrangienne *totale*
- Relaxation lagrangienne *partielle*

Méthodes exactes

Techniques de résolution

Bornes inférieures

- Relaxation lagrangienne *totale*
- Relaxation lagrangienne *partielle*

Méthodes exactes

- Solveur MIP appliqué à la formalisation $[P]$

Techniques de résolution

Bornes inférieures

- Relaxation lagrangienne *totale*
- Relaxation lagrangienne *partielle*

Méthodes exactes

- Solveur MIP appliqué à la formalisation $[P]$
- Décomposition de Benders

Techniques de résolution

Bornes inférieures

- Relaxation lagrangienne *totale*
- Relaxation lagrangienne *partielle*

Méthodes exactes

- Solveur MIP appliqué à la formalisation $[P]$
- Décomposition de Benders
- **Décomposition et génération de coupes**

Techniques de résolution

Bornes inférieures

- Relaxation lagrangienne *totale*
- Relaxation lagrangienne *partielle*

Méthodes exactes

- Solveur MIP appliqué à la formalisation $[P]$
- Décomposition de Benders
- **Décomposition et génération de coupes**

Bornes supérieures

Techniques de résolution

Bornes inférieures

- Relaxation lagrangienne *totale*
- Relaxation lagrangienne *partielle*

Méthodes exactes

- Solveur MIP appliqué à la formalisation $[P]$
- Décomposition de Benders
- **Décomposition et génération de coupes**

Bornes supérieures

- Solveur MIP appliqué à la formalisation $[P]$, *tronqué*

Techniques de résolution

Bornes inférieures

- Relaxation lagrangienne *totale*
- Relaxation lagrangienne *partielle*

Méthodes exactes

- Solveur MIP appliqué à la formalisation $[P]$
- Décomposition de Benders
- **Décomposition et génération de coupes**

Bornes supérieures

- Solveur MIP appliqué à la formalisation $[P]$, *tronqué*
- **Décomposition et génération de coupes adaptée**

II Planification du personnel contrainte par la réalisabilité de la production

- 1 Introduction
- 2 Décomposition et génération de coupes
 - Processus général
 - Résolution du problème maître
 - Coupes initiales énergétiques
- 3 Expérimentations numériques
- 4 Conclusion

Algorithme général

- Décomposition de $[P] \rightarrow [MP]$ et $[SP]$

Algorithme général

- Décomposition de $[P] \rightarrow [MP]$ et $[SP]$
- Résolution de $[MP] \leftrightarrow \bar{y}$: roulement \rightarrow opérateur

Algorithme général

- Décomposition de $[P] \rightarrow [MP]$ et $[SP]$
- Résolution de $[MP] \leftrightarrow \bar{y}$: roulement \rightarrow opérateur
- Résolution de $[SP(\bar{y})]$: \bar{y} est réalisable ?

Algorithme général

- Décomposition de $[P] \rightarrow [MP]$ et $[SP]$
- Résolution de $[MP] \leftrightarrow \bar{y}$: roulement \rightarrow opérateur
- Résolution de $[SP(\bar{y})]$: \bar{y} est réalisable ?
 - Jobs j exécutables sur leur fenêtre d'exécution $[r_j, d_j]$?

Algorithme général

- Décomposition de $[P] \rightarrow [MP]$ et $[SP]$
- Résolution de $[MP] \leftrightarrow \bar{y}$: roulement \rightarrow opérateur
- Résolution de $[SP(\bar{y})]$: \bar{y} est réalisable ?
 - Jobs j exécutables sur leur fenêtre d'exécution $[r_j, d_j]$?
 - Affectation réalisable de compétences aux opérateurs ?

Algorithme général

- Décomposition de $[P] \rightarrow [MP]$ et $[SP]$
- Résolution de $[MP] \leftrightarrow \bar{y}$: roulement \rightarrow opérateur
- Résolution de $[SP(\bar{y})]$: \bar{y} est réalisable ?
 - Jobs j exécutables sur leur fenêtre d'exécution $[r_j, d_j]$?
 - Affectation réalisable de compétences aux opérateurs ?
- Si $[SP(\bar{y})]$ parvient à planifier tous les jobs :

Algorithme général

- Décomposition de $[P] \rightarrow [MP]$ et $[SP]$
- Résolution de $[MP] \leftrightarrow \bar{y}$: roulement \rightarrow opérateur
- Résolution de $[SP(\bar{y})]$: \bar{y} est réalisable ?
 - Jobs j exécutables sur leur fenêtre d'exécution $[r_j, d_j]$?
 - Affectation réalisable de compétences aux opérateurs ?
- Si $[SP(\bar{y})]$ parvient à planifier tous les jobs :
 - $\bar{y} \Rightarrow$ affectation de moindre coût permettant de planifier l'ensemble des jobs

Algorithme général

- Décomposition de $[P] \rightarrow [MP]$ et $[SP]$
- Résolution de $[MP] \leftrightarrow \bar{y}$: roulement \rightarrow opérateur
- Résolution de $[SP(\bar{y})]$: \bar{y} est réalisable ?
 - Jobs j exécutables sur leur fenêtre d'exécution $[r_j, d_j]$?
 - Affectation réalisable de compétences aux opérateurs ?
- Si $[SP(\bar{y})]$ parvient à planifier tous les jobs :
 - $\bar{y} \Rightarrow$ affectation de moindre coût permettant de planifier l'ensemble des jobs
 - \leftrightarrow **Optimum**

Algorithme général

- Décomposition de $[P] \rightarrow [MP]$ et $[SP]$
- Résolution de $[MP] \leftrightarrow \bar{y}$: roulement \rightarrow opérateur
- Résolution de $[SP(\bar{y})]$: \bar{y} est réalisable ?
 - Jobs j exécutables sur leur fenêtre d'exécution $[r_j, d_j]$?
 - Affectation réalisable de compétences aux opérateurs ?
- Si $[SP(\bar{y})]$ parvient à planifier tous les jobs :
 - $\bar{y} \Rightarrow$ affectation de moindre coût permettant de planifier l'ensemble des jobs
 - \hookrightarrow **Optimum**
- Sinon : Génération d'une coupe invalidant \bar{y} dans $[MP]$

Algorithme général

- Décomposition de $[P] \rightarrow [MP]$ et $[SP]$
- Résolution de $[MP] \leftrightarrow \bar{y}$: roulement \rightarrow opérateur
- Résolution de $[SP(\bar{y})]$: \bar{y} est réalisable ?
 - Jobs j exécutables sur leur fenêtre d'exécution $[r_j, d_j]$?
 - Affectation réalisable de compétences aux opérateurs ?
- Si $[SP(\bar{y})]$ parvient à planifier tous les jobs :
 - $\bar{y} \Rightarrow$ affectation de moindre coût permettant de planifier l'ensemble des jobs
 - \hookrightarrow **Optimum**
- Sinon : Génération d'une coupe invalidant \bar{y} dans $[MP]$
- Itérer jusqu'à :

Algorithme général

- Décomposition de $[P] \rightarrow [MP]$ et $[SP]$
- Résolution de $[MP] \leftrightarrow \bar{y}$: roulement \rightarrow opérateur
- Résolution de $[SP(\bar{y})]$: \bar{y} est réalisable ?
 - Jobs j exécutables sur leur fenêtre d'exécution $[r_j, d_j]$?
 - Affectation réalisable de compétences aux opérateurs ?
- Si $[SP(\bar{y})]$ parvient à planifier tous les jobs :
 - $\bar{y} \Rightarrow$ affectation de moindre coût permettant de planifier l'ensemble des jobs
 - \hookrightarrow **Optimum**
- Sinon : Génération d'une coupe invalidant \bar{y} dans $[MP]$
- Itérer jusqu'à :
 - **Optimum** (nombre fini de solutions pour $[MP]$)

Algorithme général

- Décomposition de $[P] \rightarrow [MP]$ et $[SP]$
- Résolution de $[MP] \leftrightarrow \bar{y}$: roulement \rightarrow opérateur
- Résolution de $[SP(\bar{y})]$: \bar{y} est réalisable ?
 - Jobs j exécutables sur leur fenêtre d'exécution $[r_j, d_j]$?
 - Affectation réalisable de compétences aux opérateurs ?
- Si $[SP(\bar{y})]$ parvient à planifier tous les jobs :
 - $\bar{y} \Rightarrow$ affectation de moindre coût permettant de planifier l'ensemble des jobs
 - \hookrightarrow **Optimum**
- Sinon : Génération d'une coupe invalidant \bar{y} dans $[MP]$
- Itérer jusqu'à :
 - **Optimum** (nombre fini de solutions pour $[MP]$)
 - limite de temps dépassée (borne inférieure)

Problème maître [MP]

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

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

Cut

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

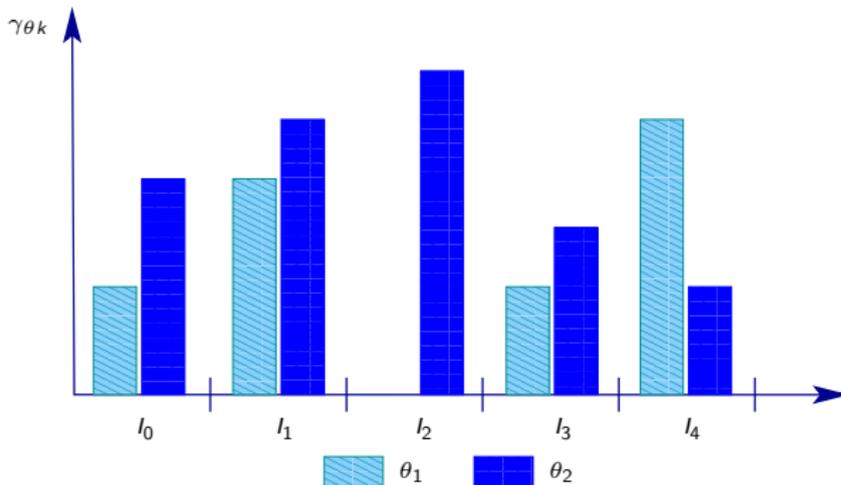
Résolution de [MP]

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

Courbe de capacité par intervalle en compétences

Courbe de capacité par intervalle en compétences

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



Sous-problème [SP(\bar{y})]

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

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

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

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

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

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

Sous-problème [SP(\bar{y})]

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

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

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

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

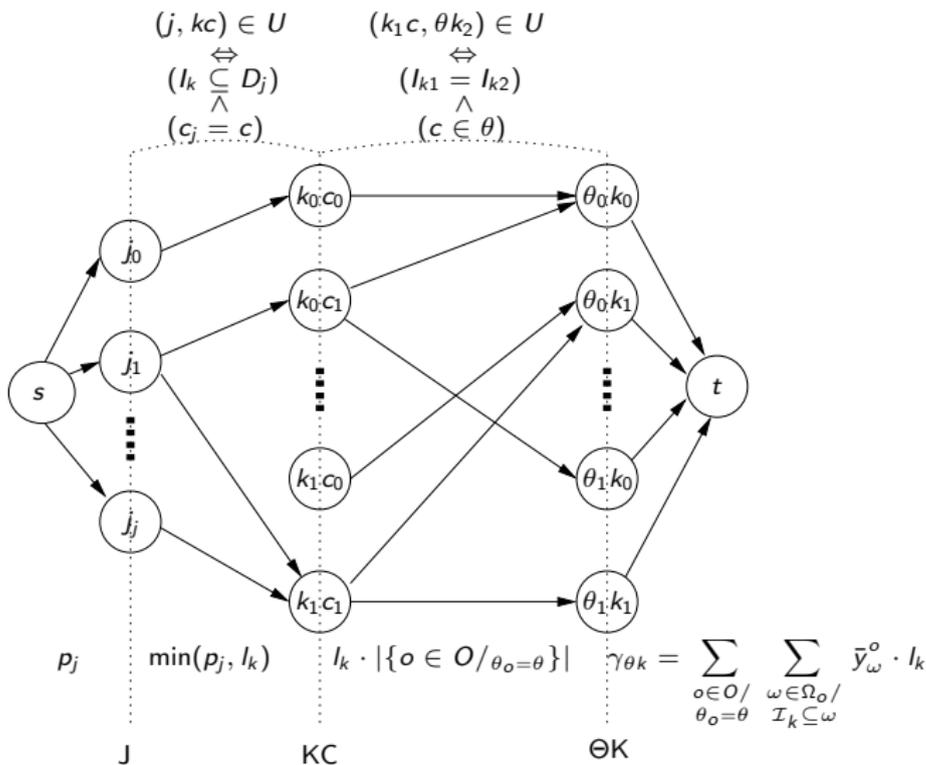
$$\forall j \in J, \forall k \in K \quad x_{jk} \in [0, \min(p_j, l_k)]$$

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

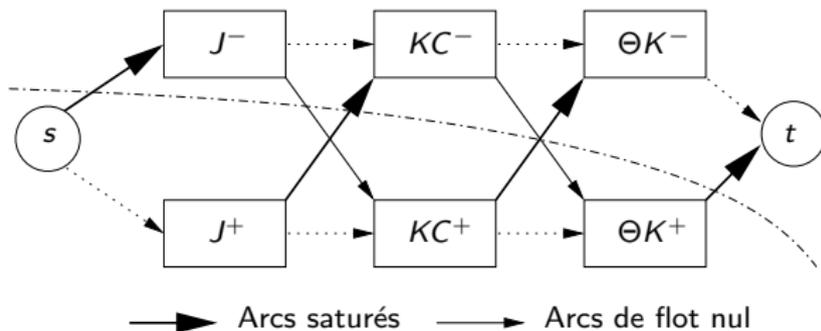
Résolution de [SP(\bar{y})]

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

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



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



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

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

Génération de coupes

Expression de la coupe de réalisabilité

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

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 un ensemble Q_r de r coupes de réalisabilité

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

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

$$\forall i \in Q_r \quad \sum_{b \in \Theta K_i^+} \sum_{\substack{o \in O / \\ \theta_o = \theta_b}} \sum_{\substack{\omega \in \Omega_o / \\ \mathcal{I}_{k_b} \subseteq \omega}} y_{\omega}^o \cdot l_{k_b} \geq \nu_i$$

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

$[MP^r] \rightarrow [MP]$ avec un ensemble Q_r de r coupes de réalisabilité

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

Caractérisation de $[MP^r]$

Sac à dos multi-choix multidimensionnel (NP-difficile)

II Planification du personnel contrainte par la réalisabilité de la production

- 1 Introduction
- 2 Décomposition et génération de coupes
 - Processus général
 - Résolution du problème maître
 - Coupes initiales énergétiques
- 3 Expérimentations numériques
- 4 Conclusion

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

Arrêt sur la solution optimale

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

Arrêt sur la solution optimale

- À chaque itération : affectation \bar{y} de moindre coût

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

Arrêt sur la solution optimale

- À chaque itération : affectation \bar{y} de moindre coût
- \hookrightarrow 1^{ère} affectation *réalisable* est **optimale**

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

Arrêt sur la solution optimale

- À chaque itération : affectation \bar{y} de moindre coût
- \hookrightarrow 1^{ère} affectation *réalisable* est **optimale**
- + : moins d'itérations / - : tps de résolution par itération

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

Arrêt sur la solution optimale

- À chaque itération : affectation \bar{y} de moindre coût
- \hookrightarrow 1^{ère} affectation *réalisable* est **optimale**
- + : moins d'itérations / - : tps de résolution par itération
- Comment ? MIP

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

Arrêt sur la solution optimale

- À chaque itération : affectation \bar{y} de moindre coût
- \hookrightarrow 1^{ère} affectation *réalisable* est **optimale**
- + : moins d'itérations / - : tps de résolution par itération
- Comment ? MIP

Arrêt sur la première solution réalisable

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

Arrêt sur la solution optimale

- À chaque itération : affectation \bar{y} de moindre coût
- \hookrightarrow 1^{ère} affectation réalisable est optimale
- + : moins d'itérations / - : tps de résolution par itération
- Comment ? MIP

Arrêt sur la première solution réalisable

- À chaque itération : affectation \bar{y} réalisable (pour $[MP^r]$)

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

Arrêt sur la solution optimale

- À chaque itération : affectation \bar{y} de moindre coût
- \hookrightarrow 1^{ère} affectation *réalisable* est **optimale**
- + : moins d'itérations / - : tps de résolution par itération
- Comment ? MIP

Arrêt sur la première solution réalisable

- À chaque itération : affectation \bar{y} réalisable (pour $[MP^r]$)
- \hookrightarrow 1^{ère} affectation *réalisable* n'est pas nécessairement **optimale**

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

Arrêt sur la solution optimale

- À chaque itération : affectation \bar{y} de moindre coût
- \hookrightarrow 1^{ère} affectation réalisable est optimale
- + : moins d'itérations / - : tps de résolution par itération
- Comment ? MIP

Arrêt sur la première solution réalisable

- À chaque itération : affectation \bar{y} réalisable (pour $[MP^r]$)
- \hookrightarrow 1^{ère} affectation réalisable n'est pas nécessairement optimale
- + : tps par itération, solution réalisable rapidement
- : contrainte de borne en plus, plus d'itérations

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

Arrêt sur la solution optimale

- À chaque itération : affectation \bar{y} de moindre coût
- \hookrightarrow 1^{ère} affectation réalisable est optimale
- + : moins d'itérations / - : tps de résolution par itération
- Comment ? MIP

Arrêt sur la première solution réalisable

- À chaque itération : affectation \bar{y} réalisable (pour $[MP^r]$)
- \hookrightarrow 1^{ère} affectation réalisable n'est pas nécessairement optimale
- + : tps par itération, solution réalisable rapidement
- : contrainte de borne en plus, plus d'itérations
- Comment ? MIP avec arrêt 1^{ère} solution ou *Feasibility Pump*

Feasibility Pump

Caractérisation

Feasibility Pump

Caractérisation

- S'applique à tout problème *MIP* :

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

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

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 *convergentes* x^* et \tilde{x} partiellement réalisables

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 *convergentes* x^* et \tilde{x} partiellement réalisables
 - x^* respecte les contraintes linéaires

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 *convergentes* x^* et \tilde{x} partiellement réalisables
 - x^* respecte les contraintes linéaires
 - \tilde{x} respecte les contraintes d'intégralité

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 *convergentes* x^* et \tilde{x} partiellement réalisables
 - x^* respecte les contraintes linéaires
 - \tilde{x} respecte les contraintes d'intégralité

Références bibliographiques

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 *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)*

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 *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)*

Feasibility Pump

Caractérisation

- S'applique à tout problème *MIP* :

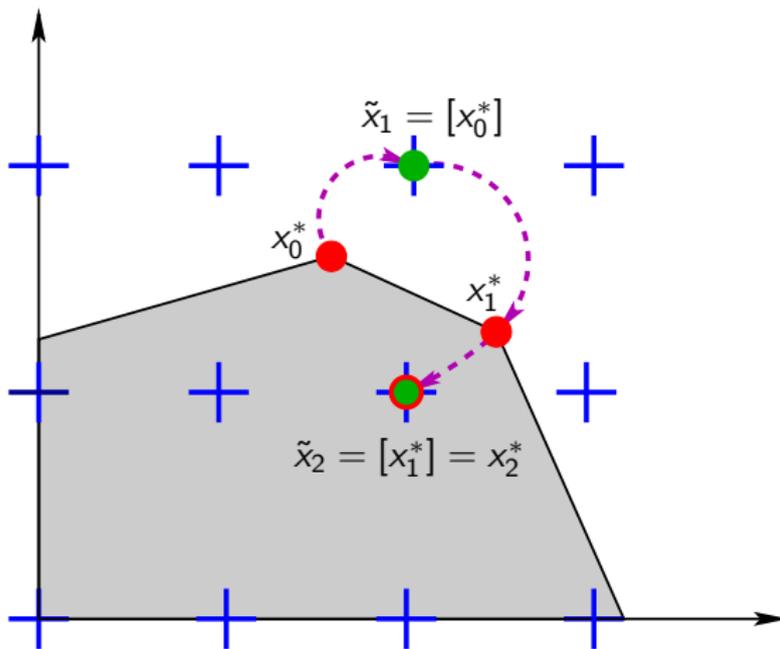
$$\min\{c^T x : Ax \geq b, \quad x_j \in \mathbb{N} \quad \forall j \in \mathcal{I}\}$$

- **But** : Trouver rapidement une solution entière réalisable
- **Principe** : Générer deux trajectoires *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)*

Illustration



II Planification du personnel contrainte par la réalisabilité de la production

- 1 Introduction
- 2 Décomposition et génération de coupes
 - Processus général
 - Résolution du problème maître
 - Coupes initiales énergétiques
- 3 Expérimentations numériques
- 4 Conclusion

Coupes initiales énergétiques

Accélérer la vitesse de convergence du processus ?

Coupes initiales énergétiques

Accélérer la vitesse de convergence du processus ?

- **Idée** : Initialiser l'ensemble *Cut* de $[MP]$

Coupes initiales énergétiques

Accélérer la vitesse de convergence du processus ?

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

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

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

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., Le Pape C. et Nuijten W. (1999)*
Problème d'ordonnancement cumulatif
(ressources peuvent exécuter plusieurs tâches en même temps)

Raisonnement énergétique

Vision énergétique du problème

Raisonnement énergétique

Vision énergétique du problème

- compétences → *ressources consommables*

Raisonnement énergétique

Vision énergétique du problème

- compétences → *ressources consommables*
- opérateurs → *fournisseurs de ressources*

Raisonnement énergétique

Vision énergétique du problème

- compétences → *ressources consommables*
- opérateurs → *fournisseurs de ressources*
- jobs → *consommateurs de ressources*

Raisonnement énergétique

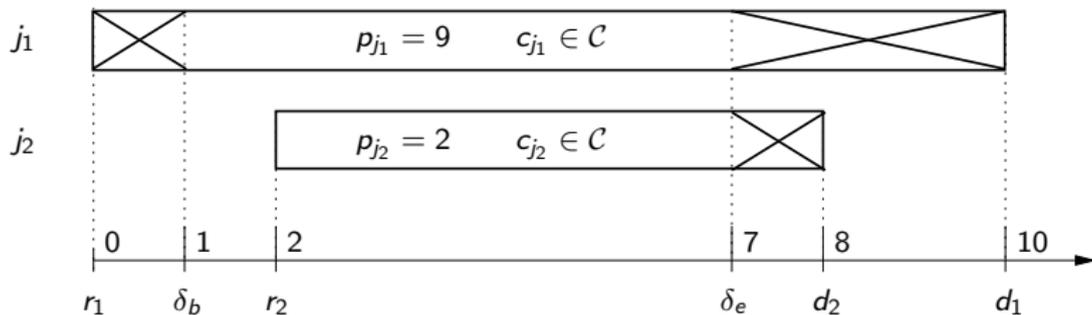
Vision énergétique du problème

- compétences \rightarrow *ressources consommables*
- opérateurs \rightarrow *fournisseurs de ressources*
- jobs \rightarrow *consommateurs de ressources*

Principe

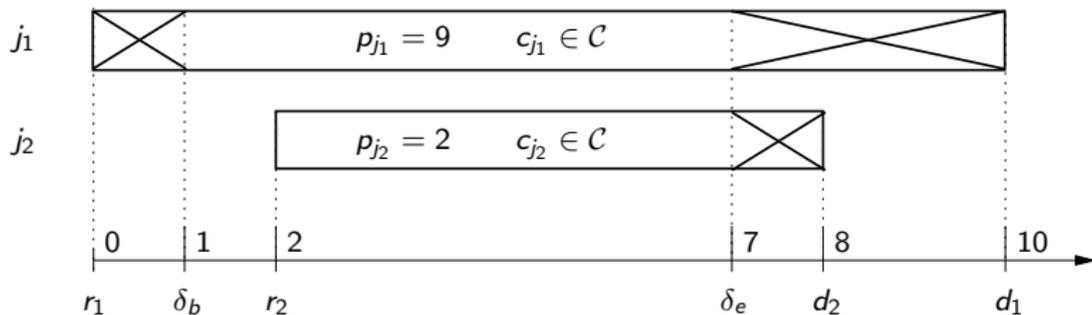
Définir des périodes de temps $[\delta_b, \delta_e]$ où il existe une **consommation minimale obligatoire** pour un sous-ensemble de compétences $\mathcal{C} \subseteq \mathcal{C}$

Illustration : coupe initiale sur $[1, 7]$ pour $\mathcal{C} = \{c_{j_1} \cup c_{j_2}\}$?



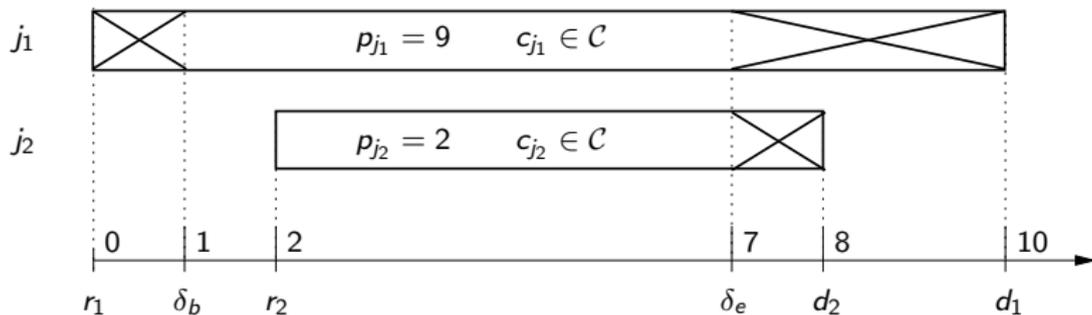
- Job j_1 : 5 unités exécutables en dehors de $[1, 7]$
 $\Rightarrow 9 - 5 = 4$ unités à exécuter sur $[1, 7]$

Illustration : coupe initiale sur $[1, 7]$ pour $\mathcal{C} = \{c_{j_1} \cup c_{j_2}\}$?



- Job j_1 : 5 unités exécutables en dehors de $[1, 7]$
 $\Rightarrow 9 - 5 = 4$ unités à exécuter sur $[1, 7]$
- Job j_2 : 1 unité exécutable en dehors de $[1, 7]$
 $\Rightarrow 2 - 1 = 1$ unité à exécuter sur $[1, 7]$

Illustration : coupe initiale sur $[1, 7]$ pour $\mathcal{C} = \{c_{j_1} \cup c_{j_2}\}$?



- Job j_1 : 5 unités exécutables en dehors de $[1, 7]$
 $\Rightarrow 9 - 5 = 4$ unités à exécuter sur $[1, 7]$
- Job j_2 : 1 unité exécutable en dehors de $[1, 7]$
 $\Rightarrow 2 - 1 = 1$ unité à exécuter sur $[1, 7]$
- \hookrightarrow consommation obligatoire de 5 unités 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 $[\delta_b, \delta_e]$

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

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

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 $[\delta_b, \delta_e]$

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

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

Structure des contraintes énergétiques

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 $[\delta_b, \delta_e]$

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

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

Structure des contraintes énergétiques

- Équivalente à celle des coupes de flot

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 $[\delta_b, \delta_e]$

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

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

Structure des contraintes énergétiques

- Équivalente à celle des coupes de flot
- $\hookrightarrow [MP^r]$ initialisé : **sac à dos multi-choix multidimensionnel**

Sélection expérimentale des coupes initiales

Précautions d'usage \Rightarrow *Sélection des coupes initiales obligatoire*

Sélection expérimentale des coupes initiales

Précautions d'usage \Rightarrow *Sélection des coupes initiales obligatoire*

- $2^{|C|}$ sous-ensembles de compétences

Sélection expérimentale des coupes initiales

Précautions d'usage \Rightarrow *Sélection des coupes initiales obligatoire*

- $2^{|C|}$ sous-ensembles de compétences
- $2^{|K|}$ périodes de temps (non forcément connexes)

Sélection expérimentale des coupes initiales

Précautions d'usage \Rightarrow *Sélection des coupes initiales obligatoire*

- $2^{|C|}$ sous-ensembles de compétences
- $2^{|K|}$ périodes de temps (non forcément connexes)

Ensembles de compétences considérés

Sélection expérimentale des coupes initiales

Précautions d'usage \Rightarrow *Sélection des coupes initiales obligatoire*

- $2^{|C|}$ sous-ensembles de compétences
- $2^{|K|}$ périodes de temps (non forcément connexes)

Ensembles de compétences considérés

- Ensemble Θ des profils de compétences

Sélection expérimentale des coupes initiales

Précautions d'usage \Rightarrow *Sélection des coupes initiales obligatoire*

- $2^{|C|}$ sous-ensembles de compétences
- $2^{|K|}$ périodes de temps (non forcément connexes)

Ensembles de compétences considérés

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

Sélection intensive des coupes initiales

Filtrage par dominance

Sélection intensive des coupes initiales

Filtrage par dominance

- *Affinement* des coupes

Sélection intensive des coupes initiales

Filtrage par dominance

- *Affinement* des coupes
- Suppression des coupes dominées

Sélection intensive des coupes initiales

Filtrage par dominance

- *Affinement* des coupes
- Suppression des coupes dominées
- Ajout des coupes non dominées à l'ensemble Cut de $[P]$

Sélection intensive des coupes initiales

Filtrage par dominance

- *Affinement* des coupes
- Suppression des coupes dominées
- Ajout des coupes non dominées à l'ensemble *Cut* de $[P]$
- Résolution de la relaxation continue de $[P]$

Sélection intensive des coupes initiales

Filtrage par dominance

- *Affinement* des coupes
- Suppression des coupes dominées
- Ajout des coupes non dominées à l'ensemble Cut de $[P]$
- Résolution de la relaxation continue de $[P]$
- Sélection empirique des contraintes :

$$\text{coupe sélectionnée} \Leftrightarrow \frac{\text{gap à l'optimum}}{\text{second membre}} \leq 12\%$$

Sélection intensive des coupes initiales

Filtrage par dominance

- *Affinement* des coupes
- Suppression des coupes dominées
- Ajout des coupes non dominées à l'ensemble Cut de $[P]$
- Résolution de la relaxation continue de $[P]$
- Sélection empirique des contraintes :

$$\text{coupe sélectionnée} \Leftrightarrow \frac{\text{gap à l'optimum}}{\text{second membre}} \leq 12\%$$

Spécificité des méthodes de coupes

Sélection intensive des coupes initiales

Filtrage par dominance

- *Affinement* des coupes
- Suppression des coupes dominées
- Ajout des coupes non dominées à l'ensemble *Cut* de $[P]$
- Résolution de la relaxation continue de $[P]$
- Sélection empirique des contraintes :

$$\text{coupe sélectionnée} \Leftrightarrow \frac{\text{gap à l'optimum}}{\text{second membre}} \leq 12\%$$

Spécificité des méthodes de coupes

- Coupes non sélectionnées conservées

Sélection intensive des coupes initiales

Filtrage par dominance

- *Affinement* des coupes
- Suppression des coupes dominées
- Ajout des coupes non dominées à l'ensemble Cut de $[P]$
- Résolution de la relaxation continue de $[P]$
- Sélection empirique des contraintes :

$$\text{coupe sélectionnée} \Leftrightarrow \frac{\text{gap à l'optimum}}{\text{second membre}} \leq 12\%$$

Spécificité des méthodes de coupes

- Coupes non sélectionnées conservées
- À chaque itération : si la solution courante du problème maître viole une de ces coupes, la coupe est ajoutée

Générateur de données

- Moduler le taux de travail sur l'horizon

Générateur de données

- Moduler le taux de travail sur l'horizon
- Dispersion des fenêtres d'exécution des jobs

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

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		

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

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 programmation : Java

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 programmation : Java
- Solveur MIP : Ilog Cplex 9.1

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 programmation : Java
- Solveur MIP : Ilog Cplex 9.1
- Processeur Intel Pentium D 930 ; 3 GHz ; 2 GB RAM

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

Instances résolues en moins de 5 minutes

		(MIP) :[Q]		(MIP) :[P]	
<i>m</i>	C	Succès	Temps(Succès)	Succès	Temps(Succès)
15	1-2	27.8%	72.8s	83.3%	31.0s
	3-5	35.8%	110.5s	86.4%	44.1s
(m = 15)		32.6%	97.7s	85.2%	39.0s
25	1-2	14.8%	151.6s	46.3%	31.7s
	3-5	13.6%	151.8s	38.3%	56.4s
(m = 25)		14.1%	151.7s	41.5%	45.4s
(Total)		23.3%	114.0s	63.3%	41.1s

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

Instances résolues en moins de 5 minutes

		(Coupe)		(Coupe) avec coupes initiales	
<i>m</i>	C	Succès	Temps(Succès)	Succès	Temps(Succès)
15	1-2	85.2%	33.2s	100.0%	5.5s
	3-5	53.1%	88.1s	80.2%	31.8s
(m = 15)		65.9%	59.8s	88.1%	19.9s
25	1-2	55.6%	17.5s	77.8%	21.1s
	3-5	40.7%	53.9s	49.4%	48.2s
(m = 25)		46.7%	36.6s	60.7%	34.3s
(Total)		56.3%	50.2s	74.4%	25.7s

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

Déviation moyenne à la meilleure borne supérieure connue
 (optimum pour 90.4% des instances)

CPU	(Coupe) version solution réalisable					
	(MIP)		[MP] \leftrightarrow MIP		[MP] \leftrightarrow Feas. Pump	
	UB*/UB	σ	UB*/UB	σ	UB*/UB	σ
1s	77.6%	21.0%	87.2%	19.0%	95.4%	10.4%
5s	92.2%	15.5%	95.6%	11.7%	99.0%	3.5%
10s	95.6%	13.2%	98.2%	6.8%	99.5%	1.1%
30s	99.1%	4.2%	99.1%	4.6%	99.8%	0.5%
60s	99.6%	0.6%	99.5%	3.2%	99.9%	0.4%
300s	99.9%	0.2%	99.9%	0.5%	100.0%	0.2%

Nb : mêmes coupes initiales ajoutées à chaque méthode

Résultats : de la qualité des méthodes exactes ...

Instances résolues en moins de 5 minutes

			(MIP)	(Benders)	(Coupe) avec coupes initiales
m	$ C $	#instances	Temps	Temps	Temps
15	1-2	41	23.7s	20.1s	3.5s
	3-5	22	14.6s	42.0s	14.8s
$(m = 15)$		63	20.6s	27.8s	7.4s
25	1-2	23	22.8s	22.4s	2.9s
	3-5	21	47.3s	58.3s	19.6s
$(m = 25)$		44	34.5s	39.5s	10.9s
$(Total)$		107	26.2s	32.6s	8.9s
% instances résolues			63.3%	46.3%	74.4%

II Planification du personnel contrainte par la réalisabilité de la production

- 1 Introduction
- 2 Décomposition et génération de coupes
- 3 Expérimentations numériques
- 4 Conclusion
 - Analyse et prolongements
 - Production scientifique

Analyse

Analyse

Analyse

Analyse

- Formalisation MIP compacte performante

Analyse

Analyse

- Formalisation MIP compacte performante
- Coupes initiales énergétiques intéressantes

Analyse

Analyse

- Formalisation MIP compacte performante
- Coupes initiales énergétiques intéressantes
- Approche par décomposition et génération de coupes très avantageuse

Analyse

Analyse

- Formalisation MIP compacte performante
- Coupes initiales énergétiques intéressantes
- Approche par décomposition et génération de coupes très avantageuse
- Résolution heuristique rapide via l'utilisation de l'heuristique de *Feasibility Pump*

Prolongements

Prolongements

Prolongements

Prolongements

- Hybrider les techniques avec des outils de Programmation Par Contraintes

Prolongements

Prolongements

- Hybrider les techniques avec des outils de Programmation Par Contraintes
- Résoudre incrémentalement le problème de *sac à dos multi-choix multidimensionnel*

Prolongements

Prolongements

- Hybrider les techniques avec des outils de Programmation Par Contraintes
- Résoudre incrémentalement le problème de *sac à dos multi-choix multidimensionnel*
- Étudier un problème intégré avec un problème d'ordonnancement de production plus complexe

II Planification du personnel contrainte par la réalisabilité de la production

- 1 Introduction
- 2 Décomposition et génération de coupes
- 3 Expérimentations numériques
- 4 Conclusion
 - Analyse et prolongements
 - Production scientifique

Production scientifique

Production scientifique

Production scientifique

Production scientifique

- 1 article paru en revue avec comité de relecture (EJOR) [6]

Production scientifique

Production scientifique

- 1 article paru en revue avec comité de relecture (EJOR) [6]
- 3 conférences internationales (FRANCORO/ROADeF, MOSIM, INCOM) [1, 2, 5]
 - ⇒ INCOM'09 : Prix du meilleur papier pour le track *Scheduling in Manufacturing Systems*

Production scientifique

Production scientifique

- 1 article paru en revue avec comité de relecture (EJOR) [6]
- 3 conférences internationales
(FRANCORO/ROADeF, MOSIM, INCOM) [1, 2, 5]
⇒ INCOM'09 : Prix du meilleur papier pour le track
Scheduling in Manufacturing Systems
- 1 conférence nationale (ROADeF) [3]

Troisième partie III

Problème de job-shop intégrant des contraintes de ressources humaines

III Planification du personnel contrainte par la réalisabilité de la production

- 5 Introduction
 - Problème
 - Formalisation MIP indexée sur le temps
 - Techniques de résolution

- 6 Branch and Cut 0-1

- 7 Expérimentations numériques

- 8 Conclusion

Présentation du problème (1/4)

Déterminer un
ordonnancement
réalisable

Job-shop

Planification d'agents

Minimiser les
coûts salariaux

Présentation du problème (2/4) - Planification d'emplois du temps

Temps

Présentation du problème (2/4) - Planification d'emplois du temps

Temps

- Horizon de planification $H = \delta \cdot \pi$ avec :

Présentation du problème (2/4) - Planification d'emplois du temps

Temps

- Horizon de planification $H = \delta \cdot \pi$ avec :
 - δ nombre de tranches horaires s

Présentation du problème (2/4) - Planification d'emplois du temps

Temps

- Horizon de planification $H = \delta \cdot \pi$ avec :
 - δ nombre de tranches horaires s
 - π durée d'un shift

Nb : Formalisation permet la modélisation de travail en 3×8

Présentation du problème (2/4) - Planification d'emplois du temps

Temps

- Horizon de planification $H = \delta \cdot \pi$ avec :
 - δ nombre de tranches horaires s
 - π durée d'un shift

Nb : Formalisation permet la modélisation de travail en 3×8

Planification d'emploi du temps d'un ensemble E de μ employés

Présentation du problème (2/4) - Planification d'emplois du temps

Temps

- Horizon de planification $H = \delta \cdot \pi$ avec :
 - δ nombre de tranches horaires s
 - π durée d'un shift

Nb : Formalisation permet la modélisation de travail en 3×8

Planification d'emploi du temps d'un ensemble E de μ employés

- A_e ensemble des machines que e maîtrise

Présentation du problème (2/4) - Planification d'emplois du temps

Temps

- Horizon de planification $H = \delta \cdot \pi$ avec :
 - δ nombre de tranches horaires s
 - π durée d'un shift

Nb : Formalisation permet la modélisation de travail en 3×8

Planification d'emploi du temps d'un ensemble E de μ employés

- A_e ensemble des machines que e maîtrise
- \mathcal{T}_e ensemble de tranches horaires où e est disponible

Présentation du problème (3/4) - Job-Shop

Job-Shop : Produire un ensemble J de n jobs sur m machines

Présentation du problème (3/4) - Job-Shop

Job-Shop : Produire un ensemble J de n jobs sur m machines

- $\forall j \in J \{O_{ji}\}_{i=1..m}$ séquence d'opérations du job j

Présentation du problème (3/4) - Job-Shop

Job-Shop : Produire un ensemble J de n jobs sur m machines

- $\forall j \in J \{O_{ji}\}_{i=1..m}$ séquence d'opérations du job j
 - machine $m_{ji} \in \{1 \dots m\}$

Présentation du problème (3/4) - Job-Shop

Job-Shop : Produire un ensemble J de n jobs sur m machines

- $\forall j \in J \{O_{ji}\}_{i=1..m}$ séquence d'opérations du job j
 - machine $m_{ji} \in \{1 \dots m\}$
 - durée p_{ji}
 - ↪ on notera ρ_{jk} la durée d'exécution de j sur la machine k

Présentation du problème (3/4) - Job-Shop

Job-Shop : Produire un ensemble J de n jobs sur m machines

- $\forall j \in J \{O_{ji}\}_{i=1..m}$ séquence d'opérations du job j
 - machine $m_{ji} \in \{1 \dots m\}$
 - durée p_{ji}
 - ↪ on notera ρ_{jk} la durée d'exécution de j sur la machine k
 - non interruptible

Présentation du problème (3/4) - Job-Shop

Job-Shop : Produire un ensemble J de n jobs sur m machines

- $\forall j \in J \{O_{ji}\}_{i=1..m}$ séquence d'opérations du job j
 - machine $m_{ji} \in \{1 \dots m\}$
 - durée p_{ji}
 - \hookrightarrow on notera ρ_{jk} la durée d'exécution de j sur la machine k
 - non interruptible
 - requiert un employé qualifié pour l'utilisation de m_{ji}

Présentation du problème (3/4) - Job-Shop

Job-Shop : Produire un ensemble J de n jobs sur m machines

- $\forall j \in J \{O_{ji}\}_{i=1..m}$ séquence d'opérations du job j
 - machine $m_{ji} \in \{1 \dots m\}$
 - durée p_{ji}
 - \hookrightarrow on notera ρ_{jk} la durée d'exécution de j sur la machine k
 - non interruptible
 - requiert un employé qualifié pour l'utilisation de m_{ji}

Ordonnancement réalisable

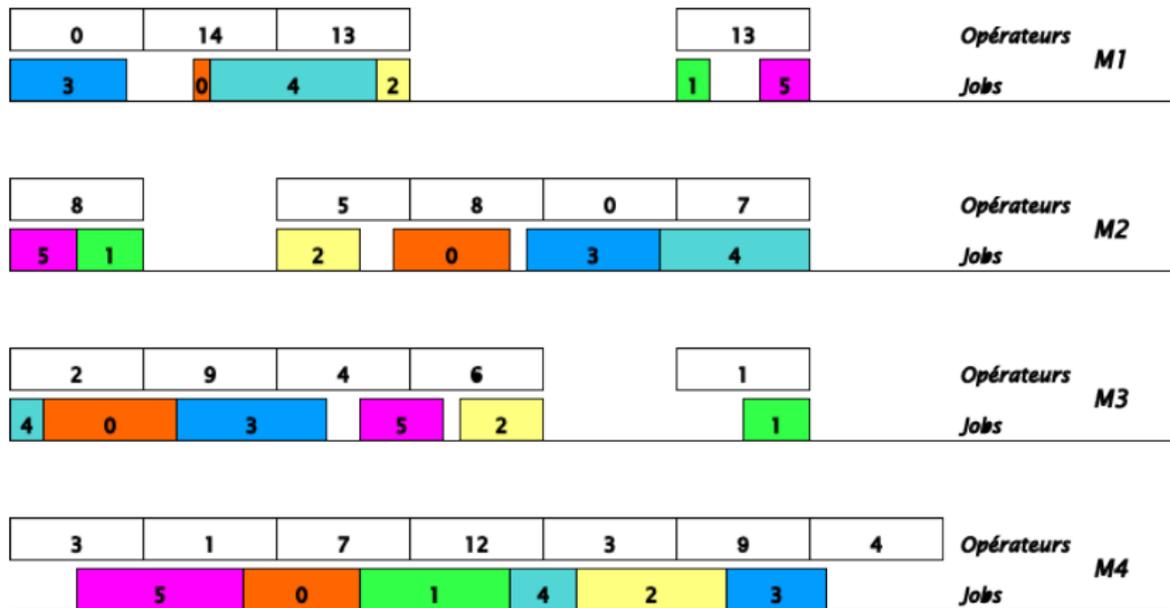
Un ordonnancement pour lequel toutes les opérations sont réalisées avec une date $C_{max} \leq H$ fixée

Présentation du problème (4/4) - Objectif

Objectif

Affecter au **moindre coût** un employé sur chaque couple (machine, tranche horaire) afin d'obtenir un *ordonnement réalisable*

Exemple (6 jobs - 4 machines - 15 employés)



Motivations (1/2) : Prolongement de la première partie

Prolongement de la première partie

Motivations (1/2) : Prolongement de la première partie

Prolongement de la première partie

- 1^{ère} partie : \rightsquigarrow Problème d'ordonnancement polynomial

Motivations (1/2) : Prolongement de la première partie

Prolongement de la première partie

- 1^{ère} partie : \rightsquigarrow Problème d'ordonnancement polynomial
- **Motivation :**
 - \hookrightarrow La décomposition et génération de coupes est-elle aussi efficace sur une problématique intégrant un problème d'ordonnancement + complexe (Job-Shop) ?

Motivations (2/2) : Confrontation à la littérature existante

Étude d'un cas traité dans la littérature

Motivations (2/2) : Confrontation à la littérature existante

Étude d'un cas traité dans la littérature

- Artigues, Gendreau, Rousseau et Vergnaud [AGRV09].
Computers and Operations Research (2009)

Motivations (2/2) : Confrontation à la littérature existante

Étude d'un cas traité dans la littérature

- Artigues, Gendreau, Rousseau et Vergnaud [AGRV09].
Computers and Operations Research (2009)
- **Motivation** : Expérimenter méthodes hybrides RO-PPC sur des problèmes couplant job-shop et planification d'agents

Motivations (2/2) : Confrontation à la littérature existante

Étude d'un cas traité dans la littérature

- Artigues, Gendreau, Rousseau et Vergnaud [AGRV09].
Computers and Operations Research (2009)
- **Motivation** : Expérimenter méthodes hybrides RO-PPC sur des problèmes couplant job-shop et planification d'agents
- *Méthode de résolution* :
↔ Résolution via un solveur de PPC avec une contrainte globale additive correspondant à la relaxation continue du problème de planification d'agents

Motivations (2/2) : Confrontation à la littérature existante

Étude d'un cas traité dans la littérature

- Artigues, Gendreau, Rousseau et Vergnaud [AGRV09].
Computers and Operations Research (2009)
- **Motivation** : Expérimenter méthodes hybrides RO-PPC sur des problèmes couplant job-shop et planification d'agents
- *Méthode de résolution* :
↪ Résolution via un solveur de PPC avec une contrainte globale additive correspondant à la relaxation continue du problème de planification d'agents
- *Notre étude* : cas spécifique de mapping activités - machines
↪ 8 des 11 instances de [AGRV09] peuvent être utilisées

III Planification du personnel contrainte par la réalisabilité de la production

- 5 Introduction
 - Problème
 - Formalisation MIP indexée sur le temps
 - Techniques de résolution
- 6 Branch and Cut 0-1
- 7 Expérimentations numériques
- 8 Conclusion

Formalisation (1/4)

Variables de décision binaires

Formalisation (1/4)

Variables de décision binaires

- $x_{eks} = 1$ ssi l'employé e travaille sur la machine k durant la tranche horaire s

Formalisation (1/4)

Variables de décision binaires

- $x_{eks} = 1$ ssi l'employé e travaille sur la machine k durant la tranche horaire s
- $y_{ikt} = 1$ ssi le job i débute son exécution sur la machine k à l'instant t

Formalisation (2/4)

Fonction-objectif

$$[P] \quad \min \Theta = \sum_{e \in E} \sum_{k \in A_e} \sum_{s \in T_e} C_{eks} \cdot X_{eks}$$

Formalisation (2/4)

Fonction-objectif

$$[P] \quad \min \Theta = \sum_{e \in E} \sum_{k \in A_e} \sum_{s \in T_e} C_{eks} \cdot X_{eks}$$

Contraintes spécifiques au problème de planification d'agents

$$\begin{aligned} \sum_{k \notin A_e} \sum_{s=0}^{\sigma} X_{eks} &= 0 & e = 1, \dots, \mu \\ \sum_{k \in A_e} \sum_{s \notin T_e} X_{eks} &= 0 & e = 1, \dots, \mu \\ \sum_{k \in A_e} (X_{eks} + X_{ek(s+1)} + X_{ek(s+2)}) &\leq 1 & e = 1, \dots, \mu \quad s = 0, \dots, \sigma - 3 \\ X_{eks} &\in \{0, 1\} & e = 1, \dots, \mu \quad k = 1, \dots, m \\ & & s = 0, \dots, \sigma - 1 \end{aligned}$$

Formalisation (3/4)

Contraintes spécifiques au problème de Job-Shop

$$\begin{aligned}
 \sum_{t=0}^{d_{ik}-\rho_{ik}} t \cdot y_{ikt} + \rho_{ik} &\leq C_{max} & i = 1, \dots, n & \quad k = m_{im} \\
 \sum_{t=r_{ik}}^{d_{ik}-\rho_{ik}} y_{ikt} &= 1 & i = 1, \dots, n & \quad k = 1, \dots, m \\
 \sum_{t=0}^{r_{ik}} y_{ikt} + \sum_{t=d_{ik}-\rho_{ik}+1}^{C_{max}} y_{ikt} &= 0 & i = 1, \dots, n & \quad k = 1, \dots, m \\
 \sum_{u=r_{ik}+\rho_{ik}}^t y_{ilu} - \sum_{u=r_{ik}}^{t-\rho_{ik}} y_{iku} &\leq 0 & i = 1, \dots, n & \quad j = 1, \dots, m-1 \\
 & & k = m_{ij} & \quad l = m_{i(j+1)} \\
 & & t = \rho_{ik} + p_{ik}, \dots, d_{il} - \rho_{il} \\
 \sum_{i=1}^n \sum_{u=\max(r_{ik}, t-\rho_{ik}+1)}^{\min(d_{ik}-\rho_{ik}, t)} y_{iku} &\leq 1 & k = 1, \dots, m & \quad t = 0, \dots, C_{max} \\
 y_{ikt} &\in \{0, 1\} & i = 1, \dots, n & \quad k = 1, \dots, m \\
 & & t = 0, \dots, C_{max}
 \end{aligned}$$

Formalisation (4/4)

Contraintes couplantes

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

$$s = \lfloor t/\pi \rfloor$$

III Planification du personnel contrainte par la réalisabilité de la production

- 5 Introduction
 - Problème
 - Formalisation MIP indexée sur le temps
 - Techniques de résolution

- 6 Branch and Cut 0-1

- 7 Expérimentations numériques

- 8 Conclusion

Techniques de résolution

Trois méthodes exactes de résolution

Techniques de résolution

Trois méthodes exactes de résolution

- Solveur MIP appliqué à la formalisation [P]

Techniques de résolution

Trois méthodes exactes de résolution

- Solveur MIP appliqué à la formalisation [P]
- Décomposition et génération de coupes de réalisabilité

Techniques de résolution

Trois méthodes exactes de résolution

- Solveur MIP appliqué à la formalisation [P]
- Décomposition et génération de coupes de réalisabilité
- **Branch and Cut 0-1 sur le travail (ou non) en chaque couple (machine, tranche horaire)**

III Planification du personnel contrainte par la réalisabilité de la production

- 5 Introduction
- 6 Branch and Cut 0-1
 - Principe
 - Coupes initiales
 - Arborescence
- 7 Expérimentations numériques
- 8 Conclusion

Principe

Idées directrices

Principe

Idées directrices

- Décomposition de $[P]$ (en relaxant les contraintes couplantes)

Principe

Idées directrices

- Décomposition de $[P]$ (en relaxant les contraintes couplantes)
 - $[Job - Shop]$ (contraintes liées aux variables y_{jkt})

Principe

Idées directrices

- Décomposition de $[P]$ (en relaxant les contraintes couplantes)
 - $[Job - Shop]$ (contraintes liées aux variables y_{jkt})
 - $[Employe]$ (contraintes liées aux variables x_{eks})

Principe

Idées directrices

- Décomposition de $[P]$ (en relaxant les contraintes couplantes)
 - $[Job - Shop]$ (contraintes liées aux variables y_{jkt})
 - $[Employe]$ (contraintes liées aux variables x_{eks})
- Fixer une distribution de couples (machine, tranche horaire) travaillés
et, par symétrie non travaillés

Principe

Idées directrices

- Décomposition de $[P]$ (en relaxant les contraintes couplantes)
 - $[Job - Shop]$ (contraintes liées aux variables y_{jkt})
 - $[Employe]$ (contraintes liées aux variables x_{eks})
- Fixer une distribution de couples (machine, tranche horaire) travaillés
et, par symétrie non travaillés
- Tester (à différents niveaux) sa réalisabilité sur

Principe

Idées directrices

- Décomposition de $[P]$ (en relaxant les contraintes couplantes)
 - $[Job - Shop]$ (contraintes liées aux variables y_{jkt})
 - $[Employe]$ (contraintes liées aux variables x_{eks})
- Fixer une distribution de couples (machine, tranche horaire) travaillés
et, par symétrie non travaillés
- Tester (à différents niveaux) sa réalisabilité sur
 - $[Job - Shop]$ via un solveur dédié

Principe

Idées directrices

- Décomposition de $[P]$ (en relaxant les contraintes couplantes)
 - $[Job - Shop]$ (contraintes liées aux variables y_{jkt})
 - $[Employe]$ (contraintes liées aux variables x_{eks})
- Fixer une distribution de couples (machine, tranche horaire) travaillés
et, par symétrie non travaillés
- Tester (à différents niveaux) sa réalisabilité sur
 - $[Job - Shop]$ via un solveur dédié
 - $[Employe]$ via un solveur de MIP

Principe

Idées directrices

- Décomposition de $[P]$ (en relaxant les contraintes couplantes)
 - $[Job - Shop]$ (contraintes liées aux variables y_{jkt})
 - $[Employe]$ (contraintes liées aux variables x_{eks})
- Fixer une distribution de couples (machine, tranche horaire) travaillés
et, par symétrie non travaillés
- Tester (à différents niveaux) sa réalisabilité sur
 - $[Job - Shop]$ via un solveur dédié
 - $[Employe]$ via un solveur de MIP
- Pour éviter exploration exhaustive :
↪ Branch and Cut 0-1, avec génération de coupes

Principe

Idées directrices

- Décomposition de $[P]$ (en relaxant les contraintes couplantes)
 - $[Job - Shop]$ (contraintes liées aux variables y_{jkt})
 - $[Employe]$ (contraintes liées aux variables x_{eks})
- Fixer une distribution de couples (machine, tranche horaire) travaillés
et, par symétrie non travaillés
- Tester (à différents niveaux) sa réalisabilité sur
 - $[Job - Shop]$ via un solveur dédié
 - $[Employe]$ via un solveur de MIP
- Pour éviter exploration exhaustive :
↪ Branch and Cut 0-1, avec génération de coupes
 - 3 ensembles de coupes initiales

Principe

Idées directrices

- Décomposition de $[P]$ (en relaxant les contraintes couplantes)
 - $[Job - Shop]$ (contraintes liées aux variables y_{jkt})
 - $[Employe]$ (contraintes liées aux variables x_{eks})
- Fixer une distribution de couples (machine, tranche horaire) travaillés
et, par symétrie non travaillés
- Tester (à différents niveaux) sa réalisabilité sur
 - $[Job - Shop]$ via un solveur dédié
 - $[Employe]$ via un solveur de MIP
- Pour éviter exploration exhaustive :
↪ Branch and Cut 0-1, avec génération de coupes
 - 3 ensembles de coupes initiales
 - de réalisabilité (en cours d'exploration)

III Planification du personnel contrainte par la réalisabilité de la production

- 5 Introduction
- 6 Branch and Cut 0-1
 - Principe
 - Coupes initiales
 - Arborescence
- 7 Expérimentations numériques
- 8 Conclusion

Coupes initiales de probing

Coupes de probing

Travail obligatoire pour le couple (machine k , tranche horaire s) ?

Coupes initiales de probing

Coupes de probing

Travail obligatoire pour le couple (machine k , tranche horaire s) ?

Comment ?

Coupes initiales de probing

Coupes de probing

Travail obligatoire pour le couple (machine k , tranche horaire s) ?

Comment ?

- Créer un job fictif j_f de durée π s'exécutant sur k durant s

Coupes initiales de probing

Coupes de probing

Travail obligatoire pour le couple (machine k , tranche horaire s) ?

Comment ?

- Créer un job fictif j_f de durée π s'exécutant sur k durant s
- Si [Job-Shop] non réalisable (\Leftrightarrow ne respecte pas C_{max}) :

Coupes initiales de probing

Coupes de probing

Travail obligatoire pour le couple (machine k , tranche horaire s) ?

Comment ?

- Créer un job fictif j_f de durée π s'exécutant sur k durant s
- Si [Job-Shop] non réalisable (\Leftrightarrow ne respecte pas C_{max}) :
 - j_f (\rightarrow absence de ressource sur (k, s)) non ordonnançable

Coupes initiales de probing

Coupes de probing

Travail obligatoire pour le couple (machine k , tranche horaire s) ?

Comment ?

- Créer un job fictif j_f de durée π s'exécutant sur k durant s
- Si [Job-Shop] non réalisable (\Leftrightarrow ne respecte pas C_{max}) :
 - j_f (\rightarrow absence de ressource sur (k, s)) non ordonnançable
 - **Un employé doit être affecté à (k, s)**

Coupes initiales de probing

Coupes de probing

Travail obligatoire pour le couple (machine k , tranche horaire s) ?

Comment ?

- Créer un job fictif j_f de durée π s'exécutant sur k durant s
- Si [Job-Shop] non réalisable (\Leftrightarrow ne respecte pas C_{max}) :
 - j_f (\rightarrow absence de ressource sur (k, s)) non ordonnançable
 - **Un employé doit être affecté à (k, s)**

Résultat

Coupes initiales de probing

Coupes de probing

Travail obligatoire pour le couple (machine k , tranche horaire s) ?

Comment ?

- Créer un job fictif j_f de durée π s'exécutant sur k durant s
- Si [Job-Shop] non réalisable (\Leftrightarrow ne respecte pas C_{max}) :
 - j_f (\rightarrow absence de ressource sur (k, s)) non ordonnançable
 - **Un employé doit être affecté à (k, s)**

Résultat

- Sous-ensemble de couples (k, s) **imposés au travail**

Coupes initiales de probing

Coupes de probing

Travail obligatoire pour le couple (machine k , tranche horaire s) ?

Comment ?

- Créer un job fictif j_f de durée π s'exécutant sur k durant s
- Si [Job-Shop] non réalisable (\Leftrightarrow ne respecte pas C_{max}) :
 - j_f (\rightarrow absence de ressource sur (k, s)) non ordonnançable
 - **Un employé doit être affecté à (k, s)**

Résultat

- Sous-ensemble de couples (k, s) **imposés au travail**
- Coupes pour [Employe] $\Rightarrow \sum_{e=1}^{\mu} x_{eks} = 1$

Coupes initiales de probing

Coupes de probing

Travail obligatoire pour le couple (machine k , tranche horaire s) ?

Comment ?

- Créer un job fictif j_f de durée π s'exécutant sur k durant s
- Si [Job-Shop] non réalisable (\Leftrightarrow ne respecte pas C_{max}) :
 - j_f (\rightarrow absence de ressource sur (k, s)) non ordonnançable
 - **Un employé doit être affecté à (k, s)**

Résultat

- Sous-ensemble de couples (k, s) **imposés au travail**
- Coupes pour [Employe] $\Rightarrow \sum_{e=1}^{\mu} x_{eks} = 1$

Généralisation

Généralisable à tout sous-ensemble de couples (k, s)

Coupes initiales de capacité

Coupes de capacité sur machine k

Nombre minimal de tranches horaires b_k travaillés sur k ?

Coupes initiales de capacité

Coupes de capacité sur machine k

Nombre minimal de tranches horaires b_k travaillés sur k ?

Comment ? b_k est le maximum de trois bornes inférieures valides

Coupes initiales de capacité

Coupes de capacité sur machine k

Nombre minimal de tranches horaires b_k travaillés sur k ?

Comment ? b_k est le maximum de trois bornes inférieures valides

- 1 Somme des couples (k, s) arbitrés par probing sur k

Coupes initiales de capacité

Coupes de capacité sur machine k

Nombre minimal de tranches horaires b_k travaillés sur k ?

Comment ? b_k est le maximum de trois bornes inférieures valides

① Somme des couples (k, s) arbitrés par probing sur k

②
$$\left\lceil \frac{\sum_{j=1}^n \rho_{jk}}{\pi} \right\rceil$$

Coupes initiales de capacité

Coupes de capacité sur machine k

Nombre minimal de tranches horaires b_k travaillés sur k ?

Comment ? b_k est le maximum de trois bornes inférieures valides

- 1 Somme des couples (k, s) arbitrés par probing sur k
- 2 $\left\lceil \frac{\sum_{j=1}^n \rho_{jk}}{\pi} \right\rceil$
- 3 Test (par probing) du nombre maximal valide de jobs fictifs

Coupes initiales de capacité

Coupes de capacité sur machine k

Nombre minimal de tranches horaires b_k travaillés sur k ?

Comment ? b_k est le maximum de trois bornes inférieures valides

- 1 Somme des couples (k, s) arbitrés par probing sur k
- 2 $\left\lceil \frac{\sum_{j=1}^n \rho_{jk}}{\pi} \right\rceil$
- 3 Test (par probing) du nombre maximal valide de jobs fictifs

Résultat : Coupes pour $[Employee]$

$$\sum_{e=1}^{\mu} \sum_{s \in \mathcal{T}_e} x_{eks} \geq b_k \quad k = 1, \dots, m$$

Coupes de mono-affectation

Coupe de mono-affectation

Assure qu'au plus un employé sera affecté à chaque couple (k, s)

Coupes de mono-affectation

Coupe de mono-affectation

Assure qu'au plus un employé sera affecté à chaque couple (k, s)

Résultat : Coupes pour $[Employee]$

$$\sum_{e \in E | (k \in A_e) \wedge (s \in T_e)} x_{eks} \leq 1 \quad k = 1, \dots, m \quad s = 0, \dots, \delta - 1$$

III Planification du personnel contrainte par la réalisabilité de la production

- 5 Introduction

- 6 Branch and Cut 0-1
 - Principe
 - Coupes initiales
 - Arborescence

- 7 Expérimentations numériques

- 8 Conclusion

Caractéristiques (1/4)

Schéma de séparation binaire

Caractéristiques (1/4)

Schéma de séparation binaire

- Fils gauche : couple (k, s) non travaillé

Caractéristiques (1/4)

Schéma de séparation binaire

- Fils gauche : couple (k, s) non travaillé
- Fils droit : couple (k, s) travaillé

Caractéristiques (1/4)

Schéma de séparation binaire

- Fils gauche : couple (k, s) non travaillé
- Fils droit : couple (k, s) travaillé

Stratégie d'exploration

Profondeur d'abord

Caractéristiques (1/4)

Schéma de séparation binaire

- Fils gauche : couple (k, s) non travaillé
- Fils droit : couple (k, s) travaillé

Stratégie d'exploration

Profondeur d'abord

Choix de la variable de branchement

Caractéristiques (1/4)

Schéma de séparation binaire

- Fils gauche : couple (k, s) non travaillé
- Fils droit : couple (k, s) travaillé

Stratégie d'exploration

Profondeur d'abord

Choix de la variable de branchement

- \bar{k} : Machine dont l'écart entre le nombre de tranches horaires imposées au travail et le nombre minimal de tranches horaires à travailler ($b_{\bar{k}}$) est maximal

Caractéristiques (1/4)

Schéma de séparation binaire

- Fils gauche : couple (k, s) non travaillé
- Fils droit : couple (k, s) travaillé

Stratégie d'exploration

Profondeur d'abord

Choix de la variable de branchement

- \bar{k} : Machine dont l'écart entre le nombre de tranches horaires imposées au travail et le nombre minimal de tranches horaires à travailler ($b_{\bar{k}}$) est maximal
- \bar{s} : Tranche horaire la plus proche de l'horizon C_{max} telle que (\bar{k}, \bar{s}) est non arbitré

Caractéristiques (2/4)

Définitions

Caractéristiques (2/4)

Définitions

- *Distribution relaxée* (moins contraignante)

Distribution de couples (k, s) tels que chaque (k, s) arbitré (par arborescence ou probing) est fixé à sa valeur et tous les autres sont **libres**

Caractéristiques (2/4)

Définitions

- *Distribution relaxée* (moins contraignante)
Distribution de couples (k, s) tels que chaque (k, s) arbitré (par arborescence ou probing) est fixé à sa valeur et tous les autres sont **libres**
- *Distribution stricte* (plus contraignante)
Distribution de couples (k, s) tels que chaque (k, s) arbitré (par arborescence ou probing) est fixé à sa valeur et tous les autres sont fixés au **repos**

Caractéristiques (2/4)

Définitions

- *Distribution relaxée* (moins contraignante)
Distribution de couples (k, s) tels que chaque (k, s) arbitré (par arborescence ou probing) est fixé à sa valeur et tous les autres sont **libres**
- *Distribution stricte* (plus contraignante)
Distribution de couples (k, s) tels que chaque (k, s) arbitré (par arborescence ou probing) est fixé à sa valeur et tous les autres sont fixés au **repos**

Évaluation

Optimum de la relaxation continue de [*Employe*] respectant la *distribution relaxée*

Caractéristiques (3/4)

Implications

Si la décision est : *le couple (\bar{k}, \bar{s}) ne doit pas être travaillé*
On essaie d'arbitrer des couples (machine, tranche horaire) en :

Caractéristiques (3/4)

Implications

Si la décision est : *le couple (\bar{k}, \bar{s}) ne doit pas être travaillé*

On essaie d'arbitrer des couples (machine, tranche horaire) en :

- utilisant les techniques de probing pour chaque couple (\bar{k}, s)

Caractéristiques (3/4)

Implications

Si la décision est : *le couple (\bar{k}, \bar{s}) ne doit pas être travaillé*

On essaie d'arbitrer des couples (machine, tranche horaire) en :

- utilisant les techniques de probing pour chaque couple (\bar{k}, s)
- vérifiant le respect de la borne $b_{\bar{k}}$ du nombre minimal de tranches horaires travaillées sur \bar{k}

Caractéristiques (3/4)

Implications

Si la décision est : *le couple (\bar{k}, \bar{s}) ne doit pas être travaillé*

On essaie d'arbitrer des couples (machine, tranche horaire) en :

- utilisant les techniques de probing pour chaque couple (\bar{k}, s)
- vérifiant le respect de la borne $b_{\bar{k}}$ du nombre minimal de tranches horaires travaillées sur \bar{k}

Règles d'élimination

Un nœud est éliminé si l'une de ces 2 conditions est vérifiée :

Caractéristiques (3/4)

Implications

Si la décision est : *le couple (\bar{k}, \bar{s}) ne doit pas être travaillé*

On essaie d'arbitrer des couples (machine, tranche horaire) en :

- utilisant les techniques de probing pour chaque couple (\bar{k}, s)
- vérifiant le respect de la borne $b_{\bar{k}}$ du nombre minimal de tranches horaires travaillées sur \bar{k}

Règles d'élimination

Un nœud est éliminé si l'une de ces 2 conditions est vérifiée :

- 1 [Employe] avec *distribution relaxée trop cher* ou non réalisable (évaluation)

Caractéristiques (3/4)

Implications

Si la décision est : *le couple (\bar{k}, \bar{s}) ne doit pas être travaillé*

On essaie d'arbitrer des couples (machine, tranche horaire) en :

- utilisant les techniques de probing pour chaque couple (\bar{k}, s)
- vérifiant le respect de la borne $b_{\bar{k}}$ du nombre minimal de tranches horaires travaillées sur \bar{k}

Règles d'élimination

Un nœud est éliminé si l'une de ces 2 conditions est vérifiée :

- ① [*Employe*] avec *distribution relaxée trop cher* ou non réalisable (évaluation)
- ② *Distribution relaxée irréalisable* pour [*Job – Shop*]

Caractéristiques (4/4)

En cas de non élimination d'un nœud

Caractéristiques (4/4)

En cas de non élimination d'un nœud

- $(\bar{x}^*, \bar{\theta}^*) \leftarrow$ optimum de [*Employe*] avec *distribution stricte*

Caractéristiques (4/4)

En cas de non élimination d'un nœud

- $(\bar{x}^*, \bar{\theta}^*) \leftarrow$ optimum de $[Employee]$ avec *distribution stricte*
 - Si \bar{x}^* est définie, on vérifie que la *distribution stricte* est réalisable pour $[Job - Shop]$

Caractéristiques (4/4)

En cas de non élimination d'un nœud

- $(\bar{x}^*, \bar{\theta}^*) \leftarrow$ optimum de $[Employee]$ avec *distribution stricte*
 - Si \bar{x}^* est définie, on vérifie que la *distribution stricte* est réalisable pour $[Job - Shop]$
 - Si *distribution stricte* valide, UB est mise à jour : $UB \leftarrow \bar{\theta}^*$

Caractéristiques (4/4)

En cas de non élimination d'un nœud

- $(\bar{x}^*, \bar{\theta}^*) \leftarrow$ optimum de $[Employee]$ avec *distribution stricte*
 - Si \bar{x}^* est définie, on vérifie que la *distribution stricte* est réalisable pour $[Job - Shop]$
 - Si *distribution stricte* valide, UB est mise à jour : $UB \leftarrow \bar{\theta}^*$
- Si UB non mise à jour : **coupe de réalisabilité** pour $[Employee]$

Caractéristiques (4/4)

En cas de non élimination d'un nœud

- $(\bar{x}^*, \bar{\theta}^*) \leftarrow$ optimum de $[Employee]$ avec *distribution stricte*
 - Si \bar{x}^* est définie, on vérifie que la *distribution stricte* est réalisable pour $[Job - Shop]$
 - Si *distribution stricte* valide, UB est mise à jour : $UB \leftarrow \bar{\theta}^*$
- Si UB non mise à jour : **coupe de réalisabilité** pour $[Employee]$

Coupe de réalisabilité pour $[Employee]$

Caractéristiques (4/4)

En cas de non élimination d'un nœud

- $(\bar{x}^*, \bar{\theta}^*) \leftarrow$ optimum de [Employe] avec *distribution stricte*
 - Si \bar{x}^* est définie, on vérifie que la *distribution stricte* est réalisable pour [Job – Shop]
 - Si *distribution stricte* valide, UB est mise à jour : $UB \leftarrow \bar{\theta}^*$
- Si UB non mise à jour : **coupe de réalisabilité** pour [Employe]

Coupe de réalisabilité pour [Employe]

- Interdire les solutions équivalentes à \bar{x}^*

Caractéristiques (4/4)

En cas de non élimination d'un nœud

- $(\bar{x}^*, \bar{\theta}^*) \leftarrow$ optimum de [Employee] avec *distribution stricte*
 - Si \bar{x}^* est définie, on vérifie que la *distribution stricte* est réalisable pour [Job – Shop]
 - Si *distribution stricte* valide, UB est mise à jour : $UB \leftarrow \bar{\theta}^*$
- Si UB non mise à jour : **coupe de réalisabilité** pour [Employee]

Coupe de réalisabilité pour [Employee]

- Interdire les solutions équivalentes à \bar{x}^*
- Forcer le travail sur un couple (k, s) non travaillé pour *distribution stricte*

Caractéristiques (4/4)

En cas de non élimination d'un nœud

- $(\bar{x}^*, \bar{\theta}^*) \leftarrow$ optimum de [Employe] avec *distribution stricte*
 - Si \bar{x}^* est définie, on vérifie que la *distribution stricte* est réalisable pour [Job – Shop]
 - Si *distribution stricte* valide, UB est mise à jour : $UB \leftarrow \bar{\theta}^*$
- Si UB non mise à jour : **coupe de réalisabilité** pour [Employe]

Coupe de réalisabilité pour [Employe]

- Interdire les solutions équivalentes à \bar{x}^*
- Forcer le travail sur un couple (k, s) non travaillé pour *distribution stricte*
- Coupe : $\sum_{(k,s)\text{non arbitré}} \sum_{e=1}^{\mu} x_{eks} \geq 1$

Données

8 instances de [AGRV09]

Instance	n	m	μ	μ_{extra}	δ	π
<i>ejs</i>	6	4	25	10	8	8
<i>ejs8X8</i>	8	8	40	20	8	8
<i>ejs10X10</i>	10	10	50	20	10	10

Outils

- Langage de développement : Java sauf pour [AGRV09] (C++)
- Solveur MIP : Ilog Cplex 12.1
- Solveurs PPC ([AGRV09]) : Ilog Solver 6.7 & Scheduler 6.7
- Solveur de job-shop dédié :
 Branch & Bound (*Carlier, Péridy, Pinson, Rivreau*) (C++)
- Processor : Intel Core 2 Quad Q6600 @ 2,40 GHz - 3 GB RAM
- Temps de résolution limite : 300 secondes

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

Impact des coupes initiales

Instance	Relaxation continue LP de $[P]$		Relaxation continue LP' de $[P]$ avec coupes initiales			
	LP/Θ^*	Temps	LP'/Θ^*	#coupes	Temps préprocess	Temps total
ejs4	92.4%	0.2s	100.0%	43	0.0s	0.1s
ejs9	87.3%	0.3s	94.6%	67	1.5s	0.3s
ejs10	97.1%	0.5s	100.0%	50	0.4s	0.2s
ejs8 \times 8 ₁	60.1%	1.6s	86.2%	94	9.5s	10.8s
ejs8 \times 8 ₂	68.0%	2.5s	91.6%	103	7.9s	9.6s
ejs8 \times 8 ₃	55.1%	1.6s	91.8%	93	4.6s	6.0s
ejs10 \times 10 ₁	55.7%	11.4s	93.1%	158	17.4s	24.4s
ejs10 \times 10 ₃	69.1%	18.1s	88.9%	164	218.8s	233.2s

Résultats : de la qualité des méthodes exactes ...

Résultats des méthodes exactes

		MIP		[AGRV09]		Branch and Cut 0-1 avec coupes initiales			
Instance	Θ^*	Θ	Temps	Θ	Temps	Θ	Temps préprocess	Temps total	#coupes init. + #coupes
ejs4	23	23	0.2s	23	0.9s	23	0.0s	0.0s	43 + 0
ejs9	24	24	11.4s	24	87.9s	24	1.5s	5.0s	67 + 44
ejs10	23	23	1.0s	23	3.5s	23	0.4s	0.6s	50 + 2
ejs8 \times 8 ₁	78	84	TL	78	64.8s	78	9.5s	37.8s	94 + 66
ejs8 \times 8 ₂	96	96	TL	96	98.9s	96	7.9s	43.3s	103 + 108
ejs8 \times 8 ₃	83	83	TL	83	29.7s	83	4.5s	9.0s	93 + 15
ejs10 \times 10 ₁	124	-1	TL	137	TL	124	17.5s	33.0s	158 + 31
ejs10 \times 10 ₃	95	-1	TL	150	TL	102	218.8s	TL	164 + 251

III Planification du personnel contrainte par la réalisabilité de la production

- 5 Introduction
- 6 Branch and Cut 0-1
- 7 Expérimentations numériques
- 8 Conclusion
 - Analyse et prolongements
 - Production scientifique

Analyse

Analyse

Analyse

Analyse

- Probing performant

Analyse

Analyse

- Probing performant
- Technique de décomposition intéressante

Analyse

Analyse

- Probing performant
- Technique de décomposition intéressante
- Approche par Branch and Cut très compétitive

Prolongements

Prolongements

Prolongements

Prolongements

- Analyser davantage les échecs de construction d'un ordonnancement réalisable

Prolongements

Prolongements

- Analyser davantage les échecs de construction d'un ordonnancement réalisable
- Générer de nouvelles coupes de réalisabilité

Prolongements

Prolongements

- Analyser davantage les échecs de construction d'un ordonnancement réalisable
- Générer de nouvelles coupes de réalisabilité
- Expérimenter d'autres schémas de branchement

III Planification du personnel contrainte par la réalisabilité de la production

- 5 Introduction
- 6 Branch and Cut 0-1
- 7 Expérimentations numériques
- 8 Conclusion
 - Analyse et prolongements
 - Production scientifique

Production scientifique

Production scientifique

Production scientifique

Production scientifique

- 1 article en cours de rédaction

Production scientifique

Production scientifique

- 1 article en cours de rédaction
- 1 conférence internationale (PMS) [7]

Production scientifique

Production scientifique

- 1 article en cours de rédaction
- 1 conférence internationale (PMS) [7]
- 1 conférence nationale (ROADeF) [4]

Partie IV

Conclusion et prolongements

IV Conclusion et prolongements

Contribution

Contribution

Contribution

- Étude de deux problématiques industrielles intégrant deux niveaux de décision

Contribution

- Étude de deux problématiques industrielles intégrant deux niveaux de décision
- Proposition d'approches exactes de décomposition et de génération de coupes

Contribution

- Étude de deux problématiques industrielles intégrant deux niveaux de décision
- Proposition d'approches exactes de décomposition et de génération de coupes
- Nombreux concepts et techniques de la Recherche Opérationnelle appréhendés

Contribution

- Étude de deux problématiques industrielles intégrant deux niveaux de décision
- Proposition d'approches exactes de décomposition et de génération de coupes
- Nombreux concepts et techniques de la Recherche Opérationnelle appréhendés
- Résultats significatifs obtenus pour les deux problématiques

Contribution

- Étude de deux problématiques industrielles intégrant deux niveaux de décision
- Proposition d'approches exactes de décomposition et de génération de coupes
- Nombreux concepts et techniques de la Recherche Opérationnelle appréhendés
- Résultats significatifs obtenus pour les deux problématiques
- Production scientifique (1 publication, 4 conférences internationales, 2 conférences nationales)

Prolongements

Prolongements

Prolongements

- Réaliser les prolongements mentionnés en conclusion des deux études

Prolongements

- Réaliser les prolongements mentionnés en conclusion des deux études
- Analyser de manière théorique les intérêts des techniques de décomposition et de génération de coupes

Prolongements

- Réaliser les prolongements mentionnés en conclusion des deux études
- Analyser de manière théorique les intérêts des techniques de décomposition et de génération de coupes
- Étudier des cas particuliers des deux problèmes pour mieux appréhender leur complexité

Partie V

Production scientifique

V Production scientifique

Production scientifique

-  Couplage planification-ordonnancement : une approche par décomposition et génération de coupes.
In *FRANCORO V / ROADeF*, Grenoble France, 02 2007.
-  Couplage planification/ordonnancement : une approche par décomposition et génération de coupes.
In *MOSIM'08*, Paris France, 2008.
-  Planification d'agents et ordonnancement de production : règles d'élimination et heuristique.
In *ROADeF*, Clermont-Ferrand France, 02 2008.
-  Résolution d'un problème de Job-Shop intégrant des contraintes de Ressources Humaines.
In *ROADeF*, Nancy France, 02 2009.
-  Near optimal and optimal solutions for an integrated employee timetabling and production scheduling problem.
In *INCOM*, Moscow Russia, 06 2009.
-  Cut generation for an integrated employee timetabling and production scheduling problem.
EJOR, 201(2) :557–567, 2010.
-  Solving an integrated Job-Shop problem with human resource constraints.
In *PMS*, Tours France, 04 2010.