Formulations PLNE pour l'ordonnancement des chaînes sur une machine

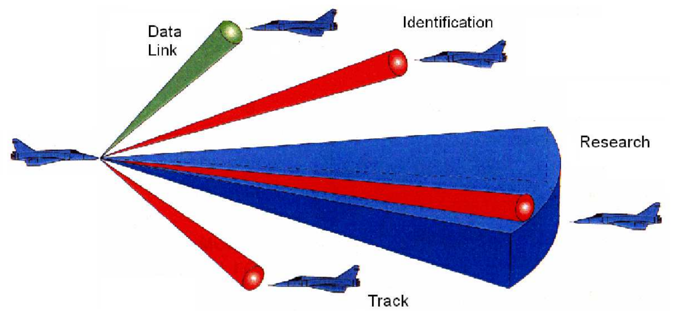
Philippe Baptiste **Ruslan Sadykov**

LIX, Ecole Polytechnique, France

ROADEF'08, Clermont-Ferrand
26 février 2008

Contenu

Ordonnancement des radars aérobornés



- Plusieurs tâches cycliques
- Un radar moderne
- On souhaite que les fréquences des tâches soient au plus proche d'optimales

Modèle mathématique I

Les données

- Horizon de temps $H = \{0, \ldots, h-1\}$
- Ensemble de tâches $N = \{1, \ldots, n\}$
- Pour chaque tâche $i \in N$:
  - Chaîne d'opérations $(O_{i0}, O_{i1}, \ldots, O_{i,n(i)})$
  - Date de début S_{i0} de l'opération O_{i0}
  - Durée d'exécution p_i d'une opération
  - Distance optimal l_i entre deux opérations
  - Fonction de pénalité $\delta_i(x) = \max\{\alpha_i(l_i - x), \beta_i(x - l_i)\}$

Variables

S_{ij} — date de début de l'opération O_{ij}, $i \in N, j \in N(i)$

Modèle mathématique II

Objective

Trouver un ordonnancement réalisable $\{S_{ij}\}_{i \in N, j \in N(i)}$ qui minimise la somme des pénalités

$$F = \sum_{i \in N} \sum_{j \in N(i)} \delta_i (S_{ij} - S_{i,j-1})$$

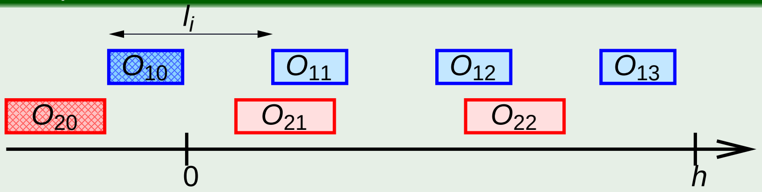
Modèle mathématique II

Objective

Trouver un ordonnancement réalisable $\{S_{ij}\}_{i \in N, j \in N(i)}$ qui minimise la somme des pénalités

$$F = \sum_{i \in N} \sum_{j \in N(i)} \delta_i (S_{ij} - S_{i,j-1})$$

Example



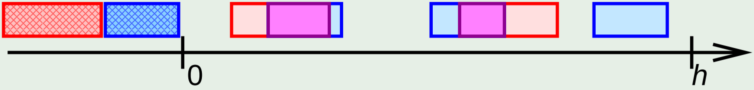
Modèle mathématique II

Objective

Trouver un ordonnancement réalisable $\{S_{ij}\}_{i\in N, j\in N(i)}$ qui minimise la somme des pénalités

$$F = \sum_{i\in N} \sum_{j\in N(i)} \delta_i (S_{ij} - S_{i,j-1})$$

Example



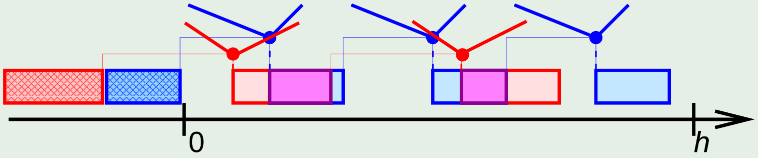
Modèle mathématique II

Objective

Trouver un ordonnancement réalisable $\{S_{ij}\}_{i\in N, j\in N(i)}$ qui minimise la somme des pénalités

$$F = \sum_{i\in N} \sum_{j\in N(i)} \delta_i(S_{ij} - S_{i,j-1})$$

Example



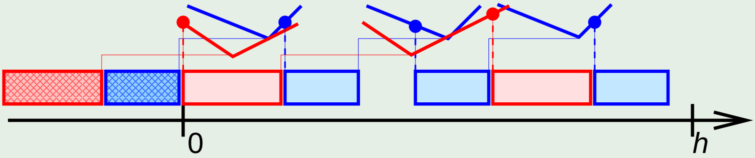
Modèle mathématique II

Objective

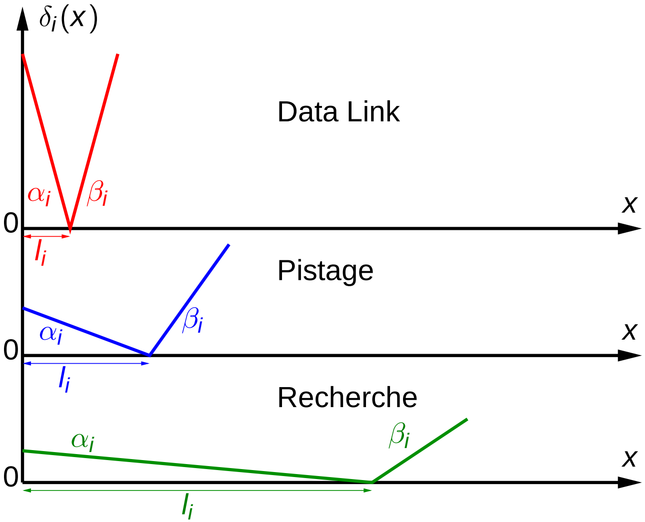
Trouver un ordonnancement réalisable $\{S_{ij}\}_{i \in N, j \in N(i)}$ qui minimise la somme des pénalités

$$F = \sum_{i \in N} \sum_{j \in N(i)} \delta_i (S_{ij} - S_{i,j-1})$$

Example



Exemples de fonctions de pénalité



Rélations avec d'autres problèmes

- Généralisation du problème juste-à-temps $1 \mid\mid \sum E_j + T_j$ (donc NP-complet [Garey, Tarjan, Wilfong, 1988])
- La version cyclique du problème est une généralisation
  - du problème d'ordonnancement non-préemptive de DCTS (*Distance-Constrained Task System*) [Han, Lin, 1992]
  - du problème de maintenance périodique (*Periodic Maintenance Problem*) [Wei, Liu, 1983]

Rélations avec d'autres problèmes

- Généralisation du problème juste-à-temps $1 \mid\mid \sum E_j + T_j$ (donc NP-complet [Garey, Tarjan, Wilfong, 1988])
- La version cyclique du problème est une généralisation
  - du problème d'ordonnancement non-préemptive de DCTS (*Distance-Constrained Task System*) [Han, Lin, 1992]
  - du problème de maintenance périodique (*Periodic Maintenance Problem*) [Wei, Liu, 1983]

Formulation (TI)

Variables

- $X_{ijt} = 1$ ssi la date de début de l'opération O_{ij} est t
- S_{ij} est égal à la date de début de l'opération O_{ij}
- $W_{ij} = \delta_i(S_{ij} - S_{i,j-1})$

Formulation (TI)

Variables

- $X_{ijt} = 1$ ssi la date de début de l'opération O_{ij} est t
- S_{ij} est égal à la date de début de l'opération O_{ij}
- $W_{ij} = \delta_i(S_{ij} - S_{i,j-1})$

Contraintes de non-chevauchement

$$\sum_{t=0}^{h-p_i} X_{ijt} = 1, \quad i \in N, j \in N(i) \cup \{0\},$$

$$\sum_{i \in N} \sum_{j \in N(i) \cup \{0\}} \sum_{t'=\max\{t-p_i+1,0\}}^{t} X_{ijt'} \leq 1, \quad t \in H.$$

Formulation (TI)

Variables

- $X_{ijt} = 1$ ssi la date de début de l'opération O_{ij} est t
- S_{ij} est égal à la date de début de l'opération O_{ij}
- $W_{ij} = \delta_i(S_{ij} - S_{i,j-1})$

Contraintes de précédence

$$S_{ij} = \sum_{t \in H} t \cdot X_{ijt}, \quad i \in N, j \in N(i) \cup \{0\},$$
$$S_{i,j-1} + p_i \leq S_{ij}, \quad i \in N, j \in N(i).$$

Formulation (TI)

Variables

- $X_{ijt} = 1$ ssi la date de début de l'opération O_{ij} est t
- S_{ij} est égal à la date de début de l'opération O_{ij}
- $W_{ij} = \delta_i(S_{ij} - S_{i,j-1})$

Fonction objective

$$\begin{aligned}
\min \quad & \sum_{i \in N} \sum_{j \in N(i)} W_{ij} \\
s.t. \quad & W_{ij} \geq \alpha_i(l_i - S_{ij} + S_{i,j-1}), \quad i \in N, j \in N(i) \\
& W_{ij} \geq \beta_i(S_{ij} - S_{i,j-1} - l_i), \quad i \in N, j \in N(i).
\end{aligned}$$

Formulation (TI)

Variables

- $X_{ijt} = 1$ ssi la date de début de l'opération O_{ij} est t
- S_{ij} est égal à la date de début de l'opération O_{ij}
- $W_{ij} = \delta_i(S_{ij} - S_{i,j-1})$

Inconvénient

On n'utilise pas le fait
que les opérations d'une même tâche sont idéntiques !

Fonction objective alternative

Notations

- $n_i(t)$ est le nombre d'opérations de tâche i commencées dans l'intervalle $[t - l_i + 1, t]$
- $\gamma_i(t) = \max\left\{\alpha_i\left(n_i(t) - 1\right), \beta_i(1 - n_i(t))\right\}$

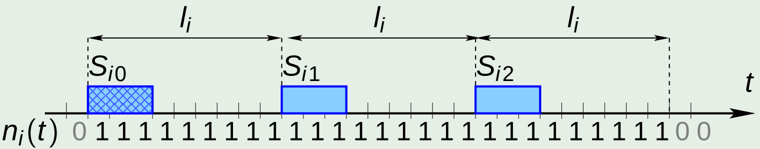
Fonction objective alternative

Notations

- $n_i(t)$ est le nombre d'opérations de tâche i commencées dans l'intervalle $[t - l_i + 1, t]$
- $\gamma_i(t) = \max\left\{\alpha_i\left(n_i(t) - 1\right), \beta_i\left(1 - n_i(t)\right)\right\}$

Example

idéalement, $n_i(t) = 1$ pour tous $t \in [S_{i0}, S_{i,n(i)} + l_i - 1]$

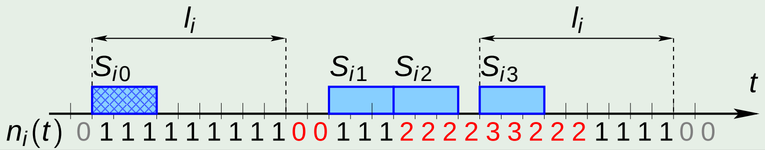


Fonction objective alternative

Notations

- $n_i(t)$ est le nombre d'opérations de tâche i commencées dans l'intervalle $[t - l_i + 1, t]$
- $\gamma_i(t) = \max\left\{\alpha_i\left(n_i(t) - 1\right), \beta_i(1 - n_i(t))\right\}$

Example



Fonction objective alternative

Notations

- $n_i(t)$ est le nombre d'opérations de tâche i commencées dans l'intervalle $[t - l_i + 1, t]$
- $\gamma_i(t) = \max\left\{\alpha_i\big(n_i(t) - 1\big), \beta_i\big(1 - n_i(t)\big)\right\}$

Theorem

Pour une tâche $i \in N$ et un ordonnancement $(S_{i0}, S_{i1}, \ldots, S_{i,n(i)})$ donnés,

$$\sum_{j \in N(i)} \delta_i(S_{ij} - S_{i,j-1}) = \sum_{t=S_{i0}}^{S_{i,n(i)}+l_i-1} \gamma_i(t).$$

Formulation (TIA)

Variables

- $X_{it} \in \mathbb{Z}_+$ est le nombre d'opérations de tâche i commencées avant ou à t
- $E_{it} \in \{0, 1\}$, $E_{it} = 1$ ssi $t \in [S_{i0}, S_{i,n(i)} + l_i - 1]$
- $W_{it} = \gamma_i(t)$

Formulation (TIA)

Variables

- $X_{it} \in \mathbb{Z}_+$ est le nombre d'opérations de tâche i commencées avant ou à t
- $E_{it} \in \{0, 1\}$, $E_{it} = 1$ ssi $t \in [S_{i0}, S_{i,n(i)} + l_i - 1]$
- $W_{it} = \gamma_i(t)$

Contraintes de non-chevauchement

$$X_{i,S_{i0}} = 1, \quad X_{i,S_{i0}-1} = 0, \quad X_{i,h-p_i} = n(i) + 1, \quad i \in N.$$
$$X_{i,t-1} \le X_{it}, \quad i \in N,\ t \in H \setminus \{0\},$$
$$\sum_{i \in N} X_{it} - \sum_{\substack{i \in N,\\ t-p_i \ge 0}} X_{i,t-p_i} \le 1, \quad t \in H.$$

Formulation (TIA)

Variables

- $X_{it} \in \mathbb{Z}_+$ est le nombre d'opérations de tâche i commencées avant ou à t
- $E_{it} \in \{0,1\}$, $E_{it} = 1$ ssi $t \in [S_{i0}, S_{i,n(i)} + l_i - 1]$
- $W_{it} = \gamma_i(t)$

Rélation entre variables X et E

$$E_{it} \geq X_{i,t-l_i+p_i} - X_{i,t-l_i},\ i \in N,\ t \in H_i,\ t \geq S_{i0} + l_i,$$
$$E_{i,t} \leq E_{i,t-1} \leq 1, \quad i \in N,\ t \in H_i \setminus \{S_{i0}\}.$$

Formulation (TIA)

Variables

- $X_{it} \in \mathbb{Z}_+$ est le nombre d'opérations de tâche i commencées avant ou à t
- $E_{it} \in \{0,1\}$, $E_{it} = 1$ ssi $t \in [S_{i0}, S_{i,n(i)} + l_i - 1]$
- $W_{it} = \gamma_i(t)$

Fonction objective

$$
\begin{aligned}
\min \quad & \sum_{i \in N} \sum_{t \in H_i} W_{it} \\
s.t. \quad & W_{it} \geq \alpha_i (X_{it} - X_{i,t-l_i} - E_{it}), \quad i \in N,\ t \in H_i, \\
& W_{it} \geq \beta_i (E_{it} - X_{it} + X_{i,t-l_i}), \quad i \in N,\ t \in H_i.
\end{aligned}
$$

Formulation (TIA)

Variables

- $X_{it} \in \mathbb{Z}_+$ est le nombre d'opérations de tâche i commencées avant ou à t
- $E_{it} \in \{0, 1\}$, $E_{it} = 1$ ssi $t \in [S_{i0}, S_{i,n(i)} + l_i - 1]$
- $W_{it} = \gamma_i(t)$

Comparaison avec (TI)

	Nombre de variables	Nombre de contraintes
(TIA)	$O(nh)$	$O(nh)$
(TI)	$O(\sum_{i \in N} n(i) \cdot h)$	$O(\sum_{i \in N} n(i) + h)$

Comparaison des formulations

Theorem

Si, pour chaque tâche $i \in N$, la date de début de la dernière opération $S_{i,n(i)}$ est fixée, alors

$$\nu_{LP}(\mathrm{TI}) \leq \nu_{LP}(\mathrm{TIA}).$$

Comparaison des formulations

Theorem

Si, pour chaque tâche $i \in N$, la date de début de la dernière opération $S_{i,n(i)}$ est fixée, alors

$$\nu_{LP}(\mathrm{TI}) \leq \nu_{LP}(\mathrm{TIA}).$$

Comparaison expérimentale

Pour les instances générées aléatoirement (les $S_{i,n(i)}$ ne sont pas fixées),

$$\frac{\nu_{LP}(\mathrm{TI})}{\nu_{LP}(\mathrm{TIA})} \approx 0.26.$$

Formulation (NC)

Notation

- S_i^k est un ordonnancement $(S_{i0}^k, S_{i1}^k, \ldots, S_{i,n(i)}^k)$ de $i \in N$
- $\chi(i,k,t) = 1$ ssi l'ordonnancement S_i^k est “actif” à t
- $X_{ik} = 1$ ssi l'ordonnacement S_i^k est choisi

Formulation (NC)

Notation

- S_i^k est un ordonnancement $(S_{i0}^k, S_{i1}^k, \ldots, S_{i,n(i)}^k)$ de $i \in N$
- $\chi(i,k,t) = 1$ ssi l'ordonnancement S_i^k est "actif" à t
- $X_{ik} = 1$ ssi l'ordonnacement S_i^k est choisi

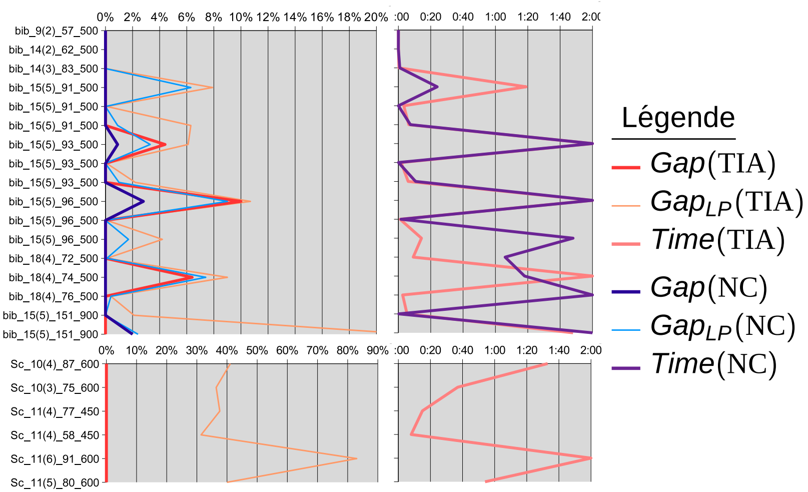
Formulation

$$\begin{aligned}
\min \quad & \sum_{i \in N} \sum_{k \in K_i} \sum_{j \in N(i)} \delta_i (S_{ij}^k - S_{i,j-1}^k) X_{ik} \\
s.t. \quad & \sum_{i \in N} \sum_{k \in K_i} \chi(i,k,t) X_{ik} \leq 1, \quad t \in H. \\
& \sum_{k \in K_i} X_{ik} = 1, \quad i \in N.
\end{aligned}$$

Algorithme de type *Branch-and-Price*

- Relaxation linéaire de (NC) : génération de colonnes
- Problème de *pricing* :
  - décomposition en n sous-problèmes
  - résolution en temps $O(nT)$ par la PD
- Branchement : division de la fenêtre de temps de S_{ij}
- Heuristique : solution fractionnaire $\longrightarrow$ entière

Instance reelles



Instances aléatoires

Horison $\in \{250, 500\}$
Densité $\in \{0.5, 0.75, 1.0\}$
Nombre total d'opérations $\in \{30, 60, 100, 150\}$
Nombre de tâches $\in \{3, 6, 15, 30\}$
Limite de temps : 1800s

Formulation	$Terminé_{LP}$	$Time_{LP}$	Gap_{LP}
(TIA)	94.9%	222s	60.0%
(NC)	93.2%	212s	57.2%

Formulation	*Terminé*	*Time*	*Gap*
(TIA)	26.1%	1458s	40.4%
(NC)	26.1%	1385s	32.4%

Conclusions

- Problème réel
- Trois formulations proposées
- Comparaison théorique et expérimentale
- Instances réeles de taille moyenne résolues optimalement

Perspectives

- Formulations facilement extensibles
  - pour le cas cyclique
  - pour le cas avec plusieurs radars idéntiques
- Un algorithme de recherche local
  - Le problème de *timing* peut-être résolu par un *Min Cost Flow* algorithme
  - Des méthodes pour estimer l'efficacité

Perspectives

- Formulations facilement extensibles
  - pour le cas cyclique
  - pour le cas avec plusieurs radars idéntiques
- Un algorithme de recherche local
  - Le problème de *timing* peut-être résolu par un *Min Cost Flow* algorithme
  - Des méthodes pour estimer l'efficacité

Perspectives

- Formulations facilement extensibles
  - pour le cas cyclique
  - pour le cas avec plusieurs radars idéntiques
- Un algorithme de recherche local
  - Le problème de *timing* peut-être résolu par un *Min Cost Flow* algorithme
  - Des méthodes pour estimer l'efficacité

Perspectives

- Formulations facilement extensibles
  - pour le cas cyclique
  - pour le cas avec plusieurs radars idéntiques
- Un algorithme de recherche local
  - Le problème de *timing* peut-être résolu par un *Min Cost Flow* algorithme
  - Des méthodes pour estimer l'efficacité