Méthode de recherche arborescente pour un problème d'ordonnancement bi-objectif

PAULINE DESSEAUX

ÉLÈVE INGÉNIEUR À L'EISTI,
ORIENTATION MATHÉMATIQUE,
OPTION AIDE À LA DÉCISION

JEUDI 15 OCTOBRE 2009

Plan

- Présentation du sujet
- L'ordonnancement
- L'optimisation multi-objectif
- Travail réalisé
- Résultats et analyse
- Conclusion

Présentation du sujet

- Problème d'ordonnancement à une machine bi-objectif :

  Chaque tâche possède une fenêtre de temps et une couleur (noir ou blanc).

  - Minimiser le retard dans l'ordonnancement des tâches
  - Minimiser le nombre de changements de couleur

- Exemple de solution :

Exemples d'application

- Tâches = processus
- Couleurs = configurations particulières
  - On cherche à regrouper les processus de même configuration

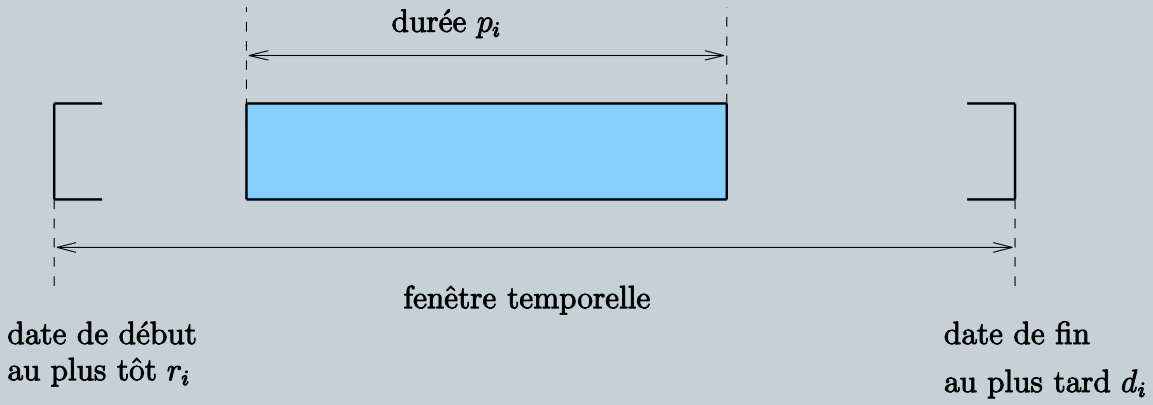
- Tâches = tâches de rénovation d'une maison
- Couleurs = artisans (électricien, plombier,…)
  - On cherche à regrouper les tâches de chaque artisan

L'ordonnancement à une machine

- La tâche i
  - Durée p_i
  - Date de début au plus tôt : r_i
  - Date de livraison : d_i
  - Date de début S_i
  - Date de fin $C_i = S_i + p_i$

Données

Variables

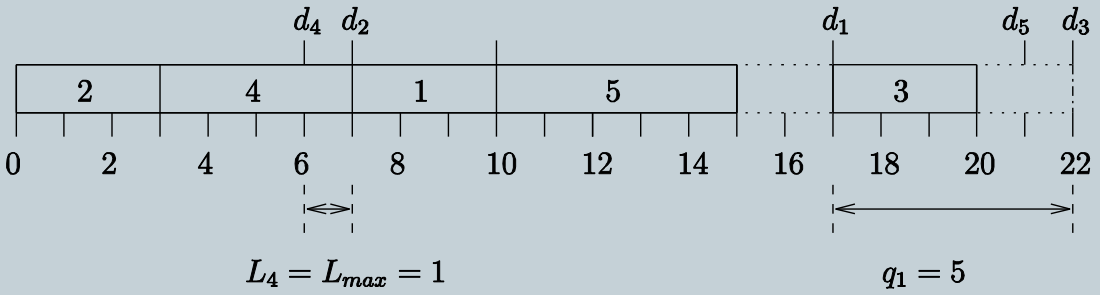


L'ordonnancement à une machine

- T=1,...,n est l'ensemble des tâches à ordonnancer
- Objectif : minimiser le retard algébrique maximal : $L_{max} = \max_{i \in T}(C_i - d_i)$
- En définissant les durées de latence q_i comme $\forall i \in T, q_i = \max_{j \in T} d_j - d_i$
- Equivalant à minimiser la durée totale avec durées de latence :

$$Cq_{max} = \max_{i \in T} Cq_i = \max_{i \in T}(C_i + q_i)$$

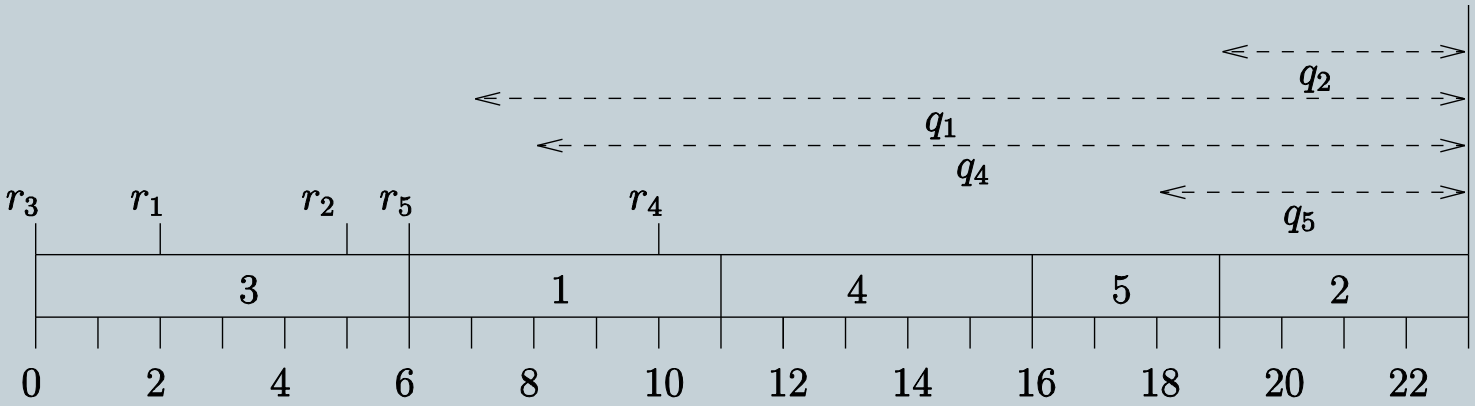
i	r_i	p_i	d_i	q_i
1	5	3	17	5
2	0	3	7	17
3	17	3	22	0
4	1	4	6	16
5	8	5	21	1



$$Cq_{max} = Cq_4 = C_4 + q_4 = 22$$

L'ordonnancement – Algorithme de Carlier (1/2)

- Minimise le Cq_{max} dans un problème à une machine
- BS: L'algorithme de Schrage ordonnance les tâches à partir des r_i et q_i : il séquence à chaque instant la tâche prête de plus grande durée de latence



- BI: Algorithme de Jackson préemptif, sert à couper des nœuds, pas à brancher.

L'ordonnancement – Algorithme de Carlier (2/2)

- Solution initiale (nœud racine) avec Schrage
- Tant que la solution optimale n'a pas été trouvée
  - Développement de l'arborescence à partir du sommet courant :

Recherche d'une tâche J_c et d'un ensemble de tâches critiques $J=[J_{c+1},\ldots,i_p]$

- i_p fixe le Cq_{max}
- J_c la plus proche de i_p de durée de latence inférieure à i_p.

J_c précède toutes les tâches de J

$$q_{J_c} \leftarrow \max\left(q_{J_c}, \sum_{i_k \in J} p_{i_k} + q_{i_p}\right)$$

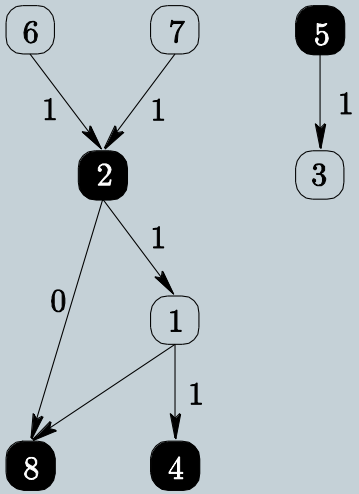
J_c suit toutes les tâches de J

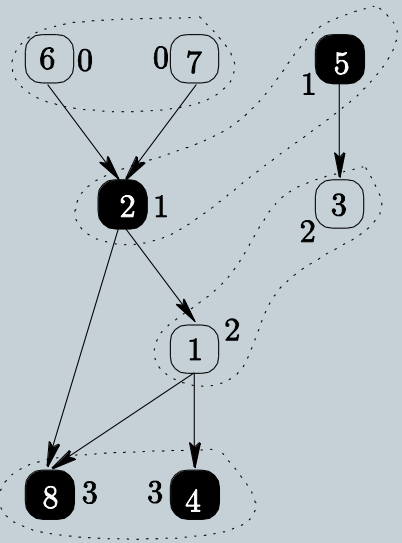
$$r_{J_c} \leftarrow \max\left(r_{J_c}, \min_{i_k \in J} r_{i_k} + \sum_{i_k \in J} p_{i_k}\right)$$

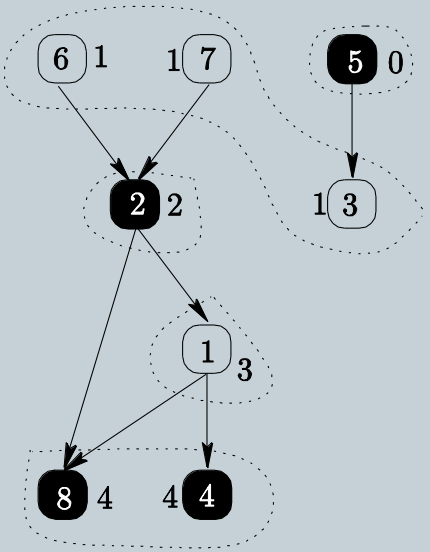
  - Repartir du nœud de plus petite borne inférieure et recalculer sa solution de Schrage

Minimisation du nombre de changements de couleurs avec contraintes de précédences

- Algorithme de Darte







Ordonnancement trié topologiquement :

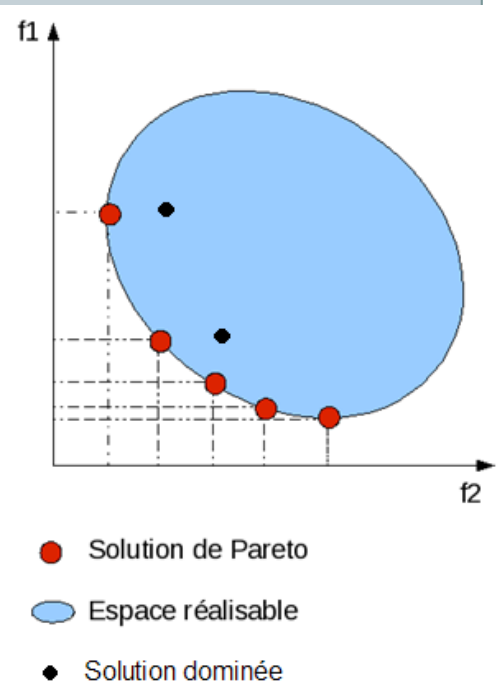
6 7 5 3 2 1 8 4

Nombre minimal de changements de couleur : 3

Ordonnancement optimal : 6 7 5 2 3 1 8 4

L'optimisation multi-objectif

- Optimiser plusieurs objectifs simultanément
- La dominance de Pareto : x_1 domine x_2 si
  - x_1 est au moins aussi bonne que x_2 dans tous les objectifs
  - x_1 est strictement meilleure que x_2 dans au moins un objectif
- La solution est l'ensemble des solutions non dominées ; ensemble *Pareto optimal*.
- Méthode exacte bi-objectif : ε-contrainte



Problème traité

- Formulation du problème :

$$(P)\begin{cases} \min Cq_{max} \\ \min(nombre\ de\ changements\ de\ couleur) \\ contraines\ d'ordonnancement\ \text{à}\ une\ machine \end{cases}$$

$\varepsilon=4$; $Cq_{max}=25$

- Données :

n : nombre de tâches à ordonnancer
r : tableau des dates de début au plus tôt
q : tableau des durées de latence
p : tableau des périodes d'exécutions
c : tableau des couleurs des tâches

$\varepsilon=3$; $Cq_{max}=40$

Méthode ε-contrainte

$$(P_\varepsilon) \begin{cases} \min Cq_{max} \\ nombre\ de\ changements\ de\ couleur \leq \varepsilon \\ contraines\ d'ordonnancement\ \text{à}\ une\ machine \end{cases}$$

On se ramène à une série de problèmes mono-objectif à résoudre

ε ← n
Tant que ε > 0 *faire*
    solution ← **Résoudre P_ε pour ε fixé**
    nbChangCouleur ← CalculeNbChangCoul(solution)
    ε ← nbChangCouleur – 1
Fin Tant que

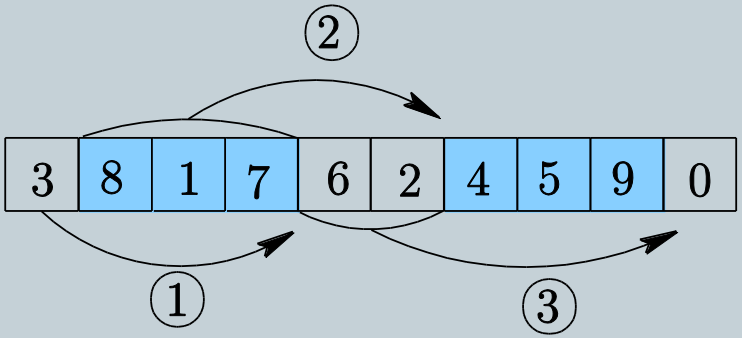
Méthode de résolution mono-objectif (P_ε)

➡ Adaptation de l'algorithme de Carlier

- Initialisation de l'arbre avec une solution réalisable
  - Heuristique ou solution précédemment trouvée
- Tant que la solution optimale n'a pas été trouvée
  - Propagation des contraintes
  - Calcul de solution réalisable (heuristiques 1 et 2)
  - Développement de l'arborescence
    - Différents modes de branchements basés sur le bloc critique
    - Ajout de contraintes de précédences et mise à jour des r_i et q_i

Heuristique 1

- Modification de l'algorithme de Schrage pour diminuer le nombre de changements de couleur.
- Réduire le nombre de changements de couleur

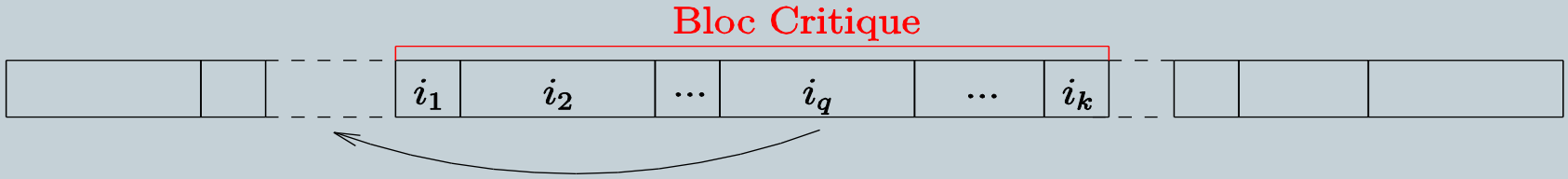


- Application de l'algorithme de Schrage sur les groupes de couleur.

Heuristique 2

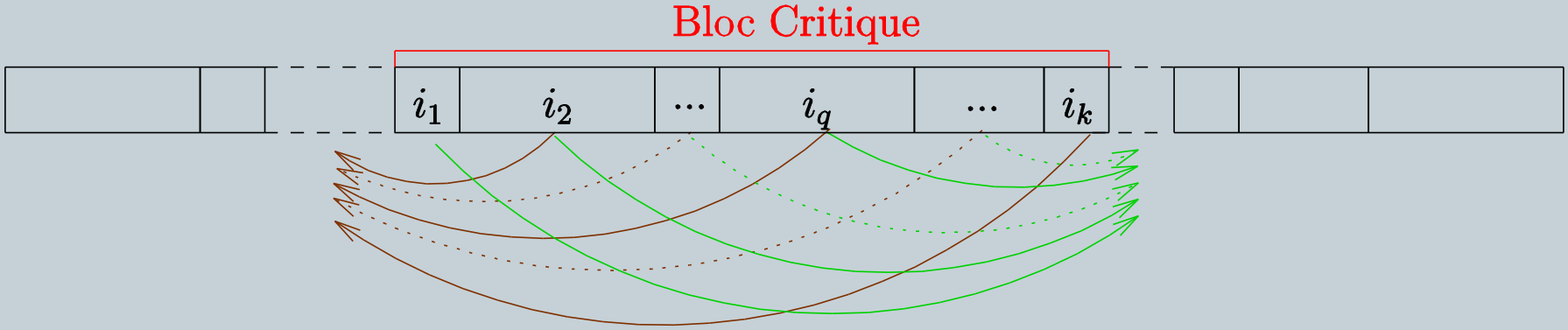
- Calcul d'une solution réalisable avec l'algorithme de Darte

- Amélioration de Cq_{max} tant que le nombre de changements de couleur est respecté



Méthodes de branchement

- Branchement k-aire



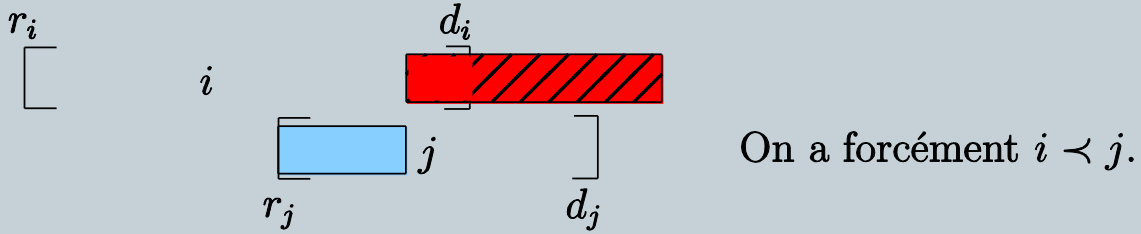
- Branchement binaire de Carlier si le Bloc Critique respecte l'ordonnancement de Schrage

Contraintes de précédences

Propagation de contraintes et Bornes Inférieures

- Ajout de contraintes de précédences :
  - Soient deux tâches i et j appartenant à un ensemble à ordonnancer

    Si $r_j+p_j+p_i > d_i$ alors i précède j



- Edge-Finding génère des contraintes de précédences
- Borne inférieure Cqmax : Jackson préemptif
- Borne inférieure changements de couleur (valeur exacte donnée par Darte)

Résultats

#jeu	n	#NC	#ND	#SolPareto	TpsExec	Arrêt prog
119	20	263	231	4	0,93	Terminé
117	30	59684	31737	5	348,45	ε=3
24	30	3020	214	2	9,07	Terminé
24	50	20809	1140	2	179,99	ε=2
53	70	14427	401	1	300,06	aucune

#jeu	n	#NC	#ND	#SolPareto	TpsExec	Arrêt prog
119	20	1615	1600	6	6,28	Terminé
117	30	19582	19536	19	130,91	Terminé
24	30	484	53	7	1,60	Terminé
24	50	8074	2431	14	424,60	ε=2
53	70	16019	1489	18	300,32	ε=5

Conclusion

- 2 objectifs très différents et très conflictuels
- Méthode exacte obtenant de bons résultats pour de petites et moyennes instances
- Prolongements possibles :
  - Trouver de nouvelles conditions pour couper des nœuds
  - Branchement sur fenêtre de temps
  - Appliquer une autre méthode bi-objectif : 2 phases