

MPQ-arbres pour les Problèmes de Placement

C. Joncour, A. Pêcher, P. Valicov

IMB - IRIT - LaBRI

5 novembre 2009

Plan

Problématique

Modèle de Fekete et Schepers

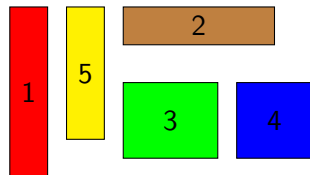
Matrices de cliques maximales/sommets

MPQ-arbres

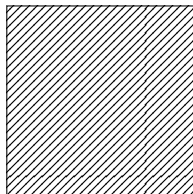
Perspectives

Problème de placement orthogonal 2D

Items

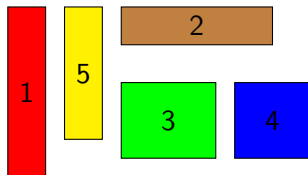


Placement dans un conteneur

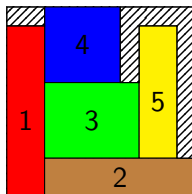


Problème de placement orthogonal 2D

Items

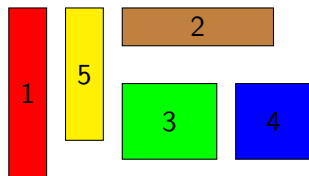


Placement dans un conteneur

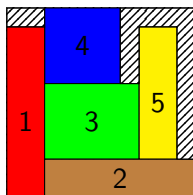


Problème de placement orthogonal 2D

Items



Placement dans un conteneur

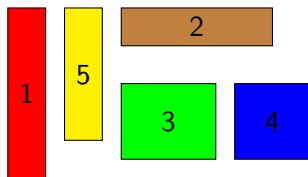


Coordonnées

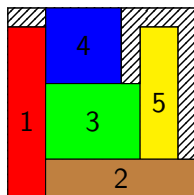
	x	y
1	0	0
2	1	0
3	1	1
4	1	3
5	3.5	1

Problème de placement orthogonal 2D

Items

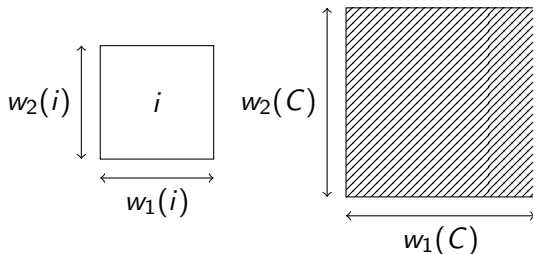


Placement dans un conteneur

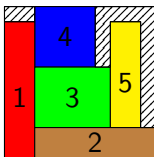


Coordonnées

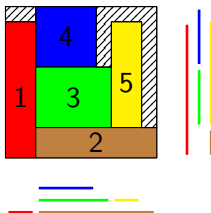
	x	y
1	0	0
2	1	0
3	1	1
4	1	3
5	3.5	1

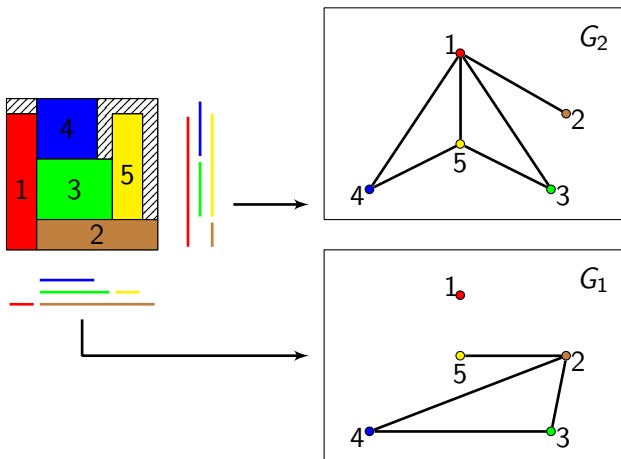


Modèle de Fekete et Schepers

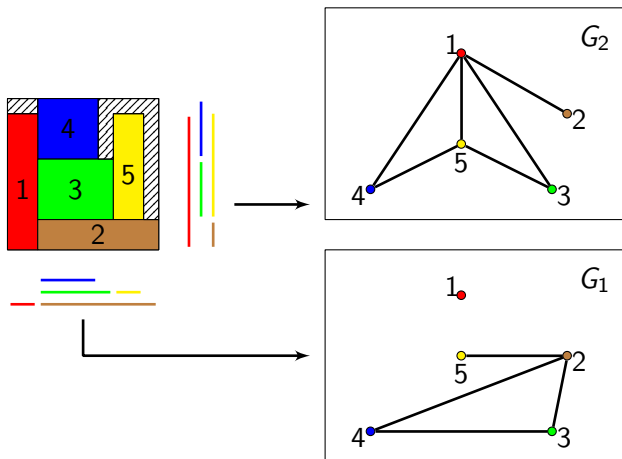


Modèle de Fekete et Schepers





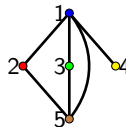
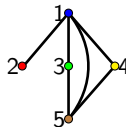
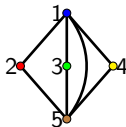
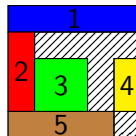
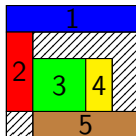
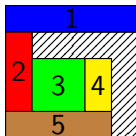
Modèle de Fekete et Schepers



$$\forall S \text{ ensemble stable de } G_d, \sum_{i \in S} w_d(i) \leq w_d(C)$$

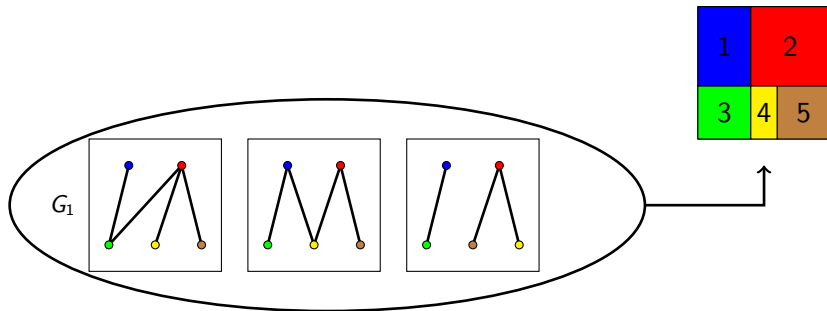
Inconvenients

Symétries



Inconvenients

Mêmes graphes d'intervalles pour une dimension



Propriété des uns consécutifs

Théorème (Fulkerson et Gross (1965))

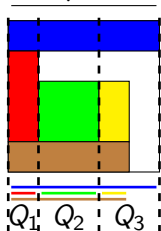
Un graphe G est d'intervalles si et seulement si la matrice des cliques/sommets de G possède la propriété des uns consécutifs sur les lignes.

Propriété des uns consécutifs

Théorème (Fulkerson et Gross (1965))

Un graphe G est d'intervalles si et seulement si la matrice des cliques/sommets de G possède la propriété des uns consécutifs sur les lignes.

Matrice de Cliques/Sommets

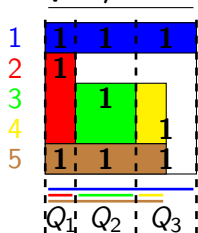


Propriété des uns consécutifs

Théorème (Fulkerson et Gross (1965))

Un graphe G est d'intervalles si et seulement si la matrice des cliques/sommets de G possède la propriété des uns consécutifs sur les lignes.

Matrice de Cliques/Sommets

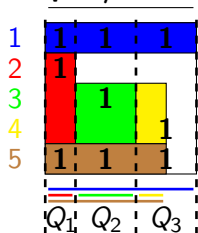


Propriété des uns consécutifs

Théorème (Fulkerson et Gross (1965))

Un graphe G est d'intervalles si et seulement si la matrice des cliques/sommets de G possède la propriété des uns consécutifs sur les lignes.

**Matrice de
Cliques/Sommets**



**Les cliques maximales
lues de gauche à droite :**

$$Q_1 = \{1, 2, 5\}$$

$$Q_2 = \{1, 3, 5\}$$

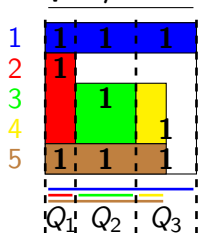
$$Q_3 = \{1, 4, 5\}$$

Propriété des uns consécutifs

Théorème (Fulkerson et Gross (1965))

Un graphe G est d'intervalles si et seulement si la matrice des cliques/sommets de G possède la propriété des uns consécutifs sur les lignes.

**Matrice de
Cliques/Sommets**



**Les cliques maximales
lues de gauche à droite :**

$$Q_1 = \{1, 2, 5\}$$

$$Q_2 = \{1, 3, 5\}$$

$$Q_3 = \{1, 4, 5\}$$

$$\begin{matrix} & Q_1 & Q_2 & Q_3 \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix} \end{matrix}$$

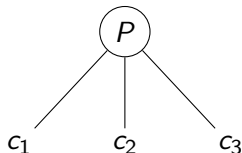
MPQ-arbres

Deux types de noeuds internes : P et Q

MPQ-arbres

Deux types de noeuds internes : P et Q

Ordres de lecture possibles :



c_1, c_2, c_3

c_1, c_3, c_2

c_2, c_1, c_3

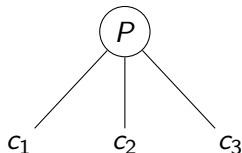
c_2, c_3, c_1

c_3, c_1, c_2

c_3, c_2, c_1

MPQ-arbres

Deux types de noeuds internes : P et Q



Ordres de lecture possibles :

c_1, c_2, c_3

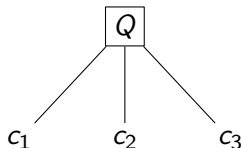
c_1, c_3, c_2

c_2, c_1, c_3

c_2, c_3, c_1

c_3, c_1, c_2

c_3, c_2, c_1

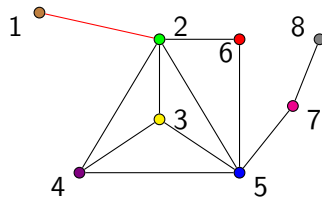
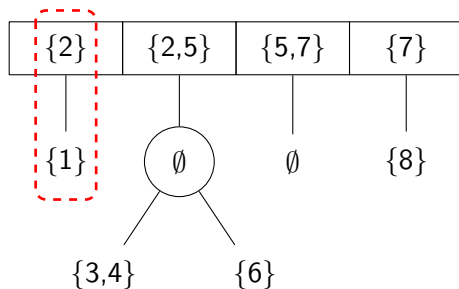


Ordres de lecture possibles :

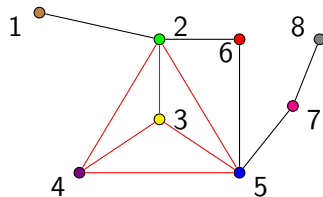
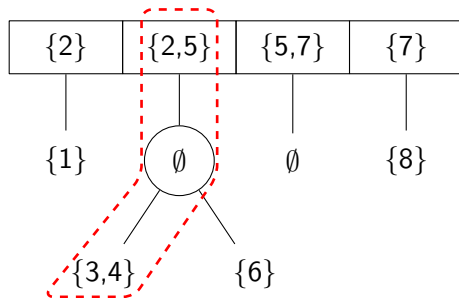
c_1, c_2, c_3

c_3, c_2, c_1

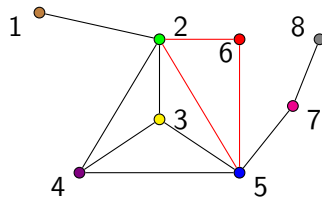
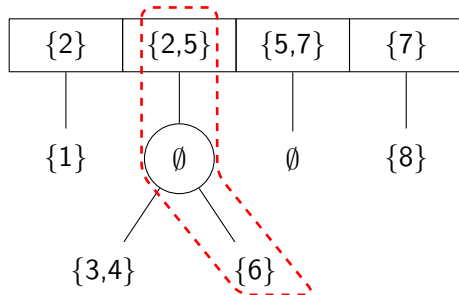
MPQ-arbres



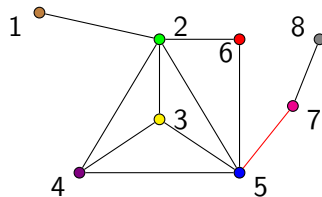
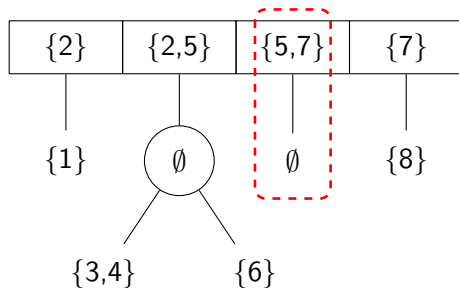
MPQ-arbres



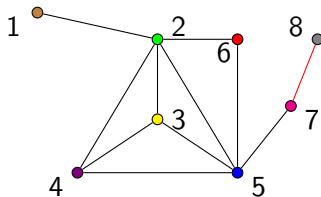
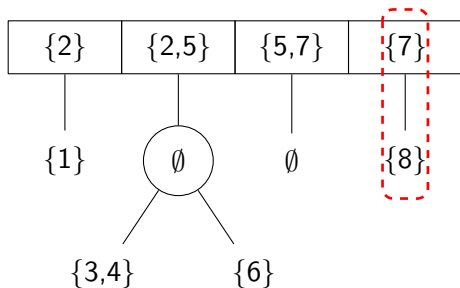
MPQ-arbres



MPQ-arbres



MPQ-arbres



MPQ-arbres

Théorème (Korte et Möhring (1989))

Un graphe G est un graphe d'intervalles si et seulement s'il existe un MPQ-arbre associé à G .

MPQ-arbres

Théorème (Korte et Möhring (1989))

Un graphe G est un graphe d'intervalles si et seulement s'il existe un MPQ-arbre associé à G .

Algorithme de reconnaissance d'un graphe d'intervalles :

- ordre sur les sommets donné par LexBFS
- ajout des sommets un par un dans le MPQ-arbre courant en suivant les patrons de Korte et Möhring
- traitement uniquement de la branche la plus à gauche de l'arbre
- arrêt si l'ajout n'est pas possible

Algorithme de génération des *MPQ*-arbres

Algorithme de génération des *MPQ*-arbres

Principe :

- générer un ordre d'insertion des sommets en testant "à la volée" les contraintes de taille
- ajouter les sommets dans le *MPQ*-arbre en utilisant un des patrons existants
- arrêt de construction d'un *MPQ*-arbre si les contraintes de tailles du conteneur ne sont pas satisfaites

Algorithme de génération des *MPQ*-arbres

Caractéristiques :

- incrémental
- réutilise les patrons existants
- uniquement la branche la plus à gauche est modifiée
- l'ordre d'insertion n'est pas connu d'avance
- possibilité de vérifier les propriétés de Fekete et Schepers au fur et à mesure

Optimisations

- ordre sur les éléments d'un ensemble associé à une feuille

Optimisations

- ordre sur les éléments d'un ensemble associé à une feuille
- ordre sur les items fils d'un noeud P

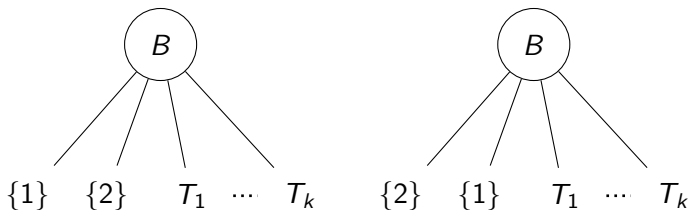
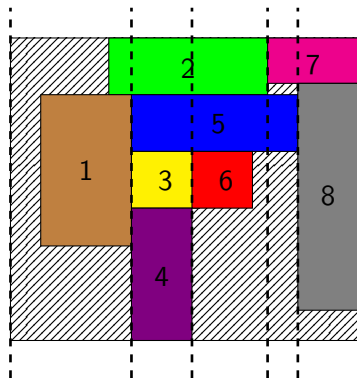
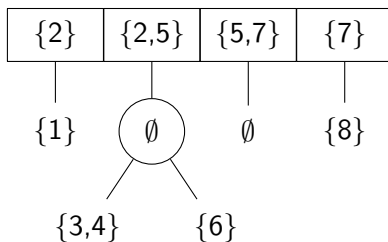


FIG.: Deux *MPQ*-arbres équivalents

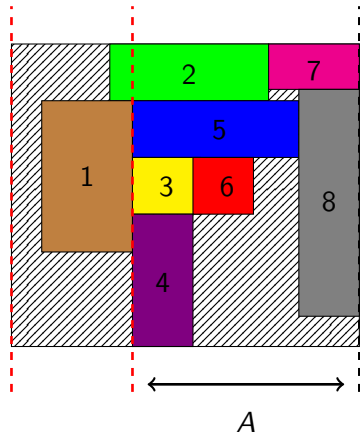
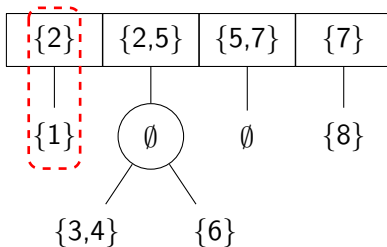
Optimisations

- aires non disponibles



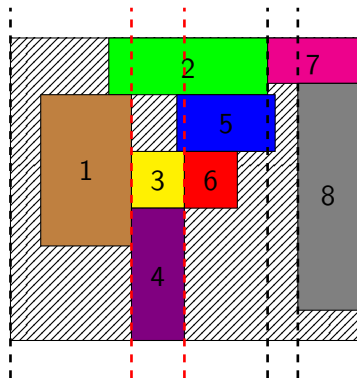
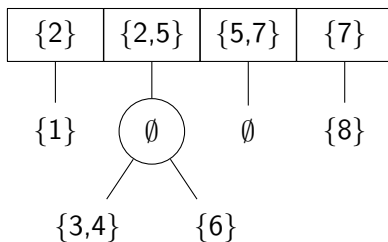
Optimisations

- aires non disponibles



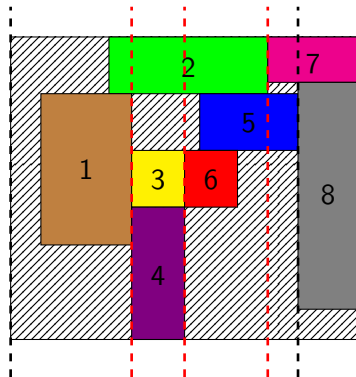
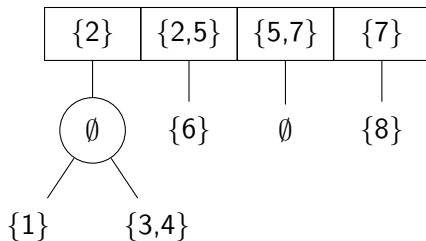
Optimisations

- réutilisation des espaces perdus



Optimisations

- réutilisation des espaces perdus



Benchmark	JPV	FS	BB	A0	A1	A2	A3	JP
ngcut1	0	0	0					0
ngcut2	0	0	0					0
ngcut3	0	0	0					0
ngcut4	0	0	0					0
ngcut5	0	0	0					0
ngcut6	0	0	0					0
ngcut7	0	0	0					0
ngcut8	0	0	0					0
ngcut9	0	0	0					0
ngcut10	0	0	0					0
ngcut11	0	0	0					0
ngcut12	0	0	0					0
gcut1	0	0	0	0	0	0	0	0
gcut2	3	0	0	0	0	25	0	0
gcut3	37	4	2	>1800	2	276	3	0
gcut4	>1800	195	46	>1800	346	>1800	376	28
gcut5	0	0	0	0	0	0	0	0
gcut6	0	0	1	0	0	9	0	0
gcut7	8	2	3	1	0	354	1	0
gcut8	1370	255	186	1202	136	>1800	108	17
gcut9	0	0	0	0	0	0	0	0
gcut10	0	0	0	0	0	6	0	0
gcut11	56	8	3	16	14	>1800	16	1
gcut12	220	109	12	63	16	>1800	25	3
gcut13	> 1800	> 1800	> 1800	> 1800	> 1800	> 1800	> 1800	> 1800

TAB.: Temps de calcul, processeur Intel Core Duo à 1.67 Ghz

Benchmark	JPV	FS	BB	A0	A1	A2	A3	JP
hccut2	0	0	0					0
hccut3	5	0	0					0
hccut4	0	0	0					0
hccut5	0	0	0					0
cgcut1	0	0	0	0	1	1	1	0
cgcut2	>1800	>1800	> 1800	>1800	>1800	533	531	39
cgcut3	20	0	95	23	23	4	4	0
okp1	1760	10	779	24	25	72	35	1
okp2	> 1800	20	288	> 1800	> 1800	1535	1559	477
okp3	> 1800	5	0	21	1	465	10	7
okp4	1290	2	14	40	2	0	4	23
okp5	> 1800	11	190	40	> 1800	513	488	3664

Temps de calcul, processeur Intel Core Duo à 1.67 Ghz

Benchmark	JPV	JP	FS	Clautiaux
E02F17	990	0	7	12
E02F20	0	2	> 1800	12
E02F22	2	0	167	4
E04F15	9	16	0	1
E04F17	1	0	13	26
E04F19	17	2	560	7
E04F20	1	1	22	3
E05F15	0	0	0	3
E05F18	1	25	0	126
E05F20	280	0	491	2
E07F15	0	0	0	1
E08F15	1	1	0	117
E20F15	0	> 1800	0	1
E00X23	> 1800	1455	> 1800	289
E03X18	1620	79	0	22
E05X15	> 1800	1219	2	0
E07X15	600	> 1800	0	1
E10X15	881	1019	0	1
E13X15	63	> 1800	0	0
E20X15	19	9	0	44

Benchmark	JPV	JP	FS	Clautiaux
E00N10	0	0	0	0
E00N15	370	1	0	2
E00N23	> 1800	66	> 1800	86
E02N20	> 1800	1236	0	1
E03N10	1	0	0	0
E03N15	> 1800	22	0	1
E03N16	> 1800	30	2	32
E03N17	604	73	0	4
E04N15	526	6	0	1
E04N17	613	110	0	1
E04N18	1066	63	10	7
E05N15	189	54	0	0
E05N17	942	141	0	1
E07N10	0	0	0	0
E07N15	73	7	0	0
E08N15	70	70	0	1
E10N10	0	0	0	0
E10N15	155	6	0	0
E13N10	1	0	0	0
E13N15	44	0	0	0
E15N10	2	1	0	0
E15N15	52	1	0	0

Perspectives

- Amélioration des temps de calcul
- Passage à un algorithme multi-dimensionnel
- Identification des bords du conteneur : problèmes d'ordonnancement