De l'accrochage des tableaux dans les réserves d'un musée

Sujet proposé par Philippe Baptiste

Philippe.Baptiste@Polytechnique.fr

1 Conservation des toiles de maîtres et bin packing

Les grands musées nationaux n'exposent en général qu'une infime partie des œuvres qu'ils conservent. Pour des raisons de sécurité, de qualité de conservation et de surveillance, les toiles non exposées sont accrochées aux murs des réserves. La surface d'accrochage étant très limitée, un "bon" accrochage tend à minimiser le nombre total de murs utilisés.

Dans la suite, nous supposerons que tous les tableaux sont rectangulaires et qu'ils ne peuvent être accrochés que dans un sens (i.e., pas de rotation possible). L'objectif du projet est d'accrocher un ensemble donné de tableaux en utilisant un nombre minimal de murs, que nous nous supposerons tous identiques.

Nous définissons maintenant deux problèmes classiques d'optimisation : le problème de *bin packing mono-dimensionnel* ($1BP$) et le problème de *bin packing en deux dimensions* ($2BP$). Le second, au cœur de ce projet, est une transposition directe du problème d'accrochage. Il est fortement recommandé de lire l'introduction de l'AMS au bin packing mono-dimensionnel (http ://www.ams.org/new-in-math/cover/bins2.html).

1.1 Le problème de bin packing mono-dimensionnel ($1BP$).

Etant donnés n articles de dimension $c_1, ..., c_n$ entières et des bins (boîtes) de taille C, il s'agit de ranger tous les articles dans un nombre minimal de bins, la somme des dimensions des articles d'un même bin ne pouvant dépasser C.

Voici deux algorithmes classiques utilisés pour $1BP$. Il seront aussi utilisés dans la suite pour la résolution de $2BP$.

- L'algorithme *First Fit Decreasing* fonctionne de la manière suivante : les articles sont considérés dans l'ordre décroissant de leur longueur c_i. A chaque étape i, l'algorithme place l'article courant dans le premier bin qui peut l'accueillir. Si aucun bin ne peut l'accueillir, un nouveau bin est ouvert.
- L'algorithme *Best Fit Decreasing* fonctionne de manière similaire : la seule différence réside dans le choix du bin dans lequel l'article courant est placé. On utilise cette fois le bin le plus rempli qui puisse l'accueillir.

1.2 Le problème de bin packing en deux dimensions ($2BP$)

$2BP$ est une généralisation naturelle du $1BP$. Il s'agit de minimiser le nombre de grands rectangles (ou grandes boîtes, ou bins) identiques pour ranger une liste d'articles rectangulaires. Les dimensions du bin sont notées W et H et les articles ont une orientation fixe (i.e., pas de rotation possible). Chaque article i a une largeur $w_i \leq W$ et une hauteur $h_i \leq H$ (w_i et h_i entiers).

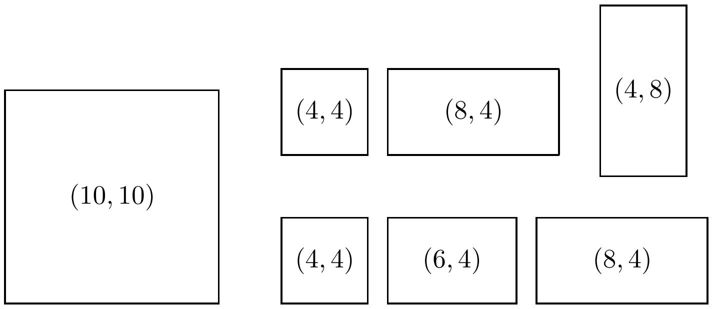

FIG. 1 – Une instance de $2BP$

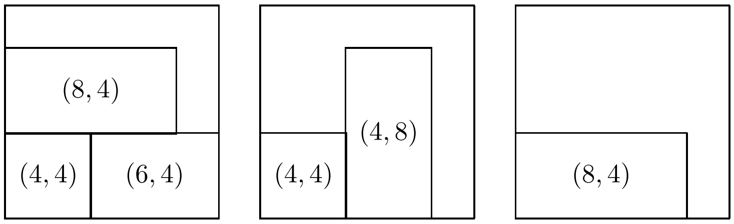

FIG. 2 – Une solution de l'instance de $2BP$

Un exemple simple de problème de bin packing en deux dimensions est illustré par la Figure 1. Un article de largeur w_i et de hauteur h_i est noté (w_i, h_i). Le bin est de taille $(10, 10)$ et six articles $\{(4, 4), (4, 4), (6, 4), (8, 4), (8, 4), (4, 8)\}$ doivent être placés. Une solution à 3 bins est représentée par la Figure 2.

1.3 Des instances

A défaut de manipuler les toiles de maîtres, vous utiliserez les instances de $2BP$ générées par Barkey & Wang et par Martello & Vigo disponibles en lignes à http ://www.lix.polytechnique.fr/˜baptiste/2dbp_instances.zip. Ce zip contient 10 fichiers `Class_01.2bp`, ..., `Class_10.2bp` qui eux-mêmes contiennent 50 instances du problème. Voici un extrait du fichier `Class_9.2bp` :

```
 9 PROBLEM CLASS
20 N. OF ITEMS
4 404 RELATIVE AND ABSOLUTE N. OF INSTANCE
100 100 HBIN,WBIN
69 60 H(I),W(I),I=1,...,N
31 84
91 18
55 78
81 77
83 57
53 90
62 65
...
55 99
```

La première ligne permet de retrouver la “classe” de l'instance et donc indirectement les paramètres qui ont permis de générer l'instance en question. Cette information nous sera utile dans la suite pour analyser les résultats classe par classe. La deuxième ligne donne le nombre d'articles de l'instance. Le premier champ de la troisième ligne identifie l'instance dans sa classe (4ème instance de la 9ème classe ici). Le deuxième champ de la troisième ligne identifie de façon absolue l'instance (la 404ème). La quatrième ligne indique la hauteur et la largeur du bin. Les lignes suivantes donnent les tailles (hauteur, largeur) des articles à placer.

2 Une interface graphique

Une solution sera représentée par un fichier texte constitué des 4 premières lignes du fichier de données, suivies d'une ligne contenant le nombre de bins utilisées puis d'une ligne par article contenant 5 champs : la hauteur, la largeur, le numéro du bin dans lequel l'article est placé et enfin les coordonnées du coin bas-gauche de l'article dans son bin.

Proposer une interface graphique permettant de charger et de visualiser une instance et une solution.

3 Deux bornes inférieures triviales

Vérifier que

$$LB_c = \left\lceil \frac{\sum_i w_i h_i}{WH} \right\rceil .$$

est une borne inférieure du nombre minimal de bins à utiliser. Montrer que

$$LB_b = |\{i : w_i > \frac{W}{2}, h_i > \frac{H}{2}\}|$$

est aussi une borne inférieure. Est-il possible d'améliorer simplement LB_b ?

Calculer les deux bornes inférieures ainsi que $LB = \max(LB_c, LB_b)$ *pour toutes les instances.*

4 L'heuristique “Bottom-Left Decreasing”

Cette méthode fonctionne en plaçant itérativement les articles dans le premier bin déjà ouvert qui peut le contenir. Quand cela n'est pas possible, on ouvre un nouveau bin. Pour cette heuristique, on considère les articles par ordre décroissant de surface et on cherche toujours à placer l'article courant le plus en bas - à gauche possible dans le bin.

Coder l'heuristique “bottom-left decreasing” et la tester sur toutes les instances. Pour les 10 classes d'instances, donner l'écart relatif moyen entre les solutions fournies par l'heuristique et LB.

5 Les heuristiques basées sur des séquences aléatoires

L'heuristique “Bottom-Left Decreasing” dépend très fortement de l'ordre initial (surface décroissante). Pour construire d'autres solutions, on applique plusieurs fois ce même algorithme en changeant l'ordre initial des articles. Plusieurs variantes sont

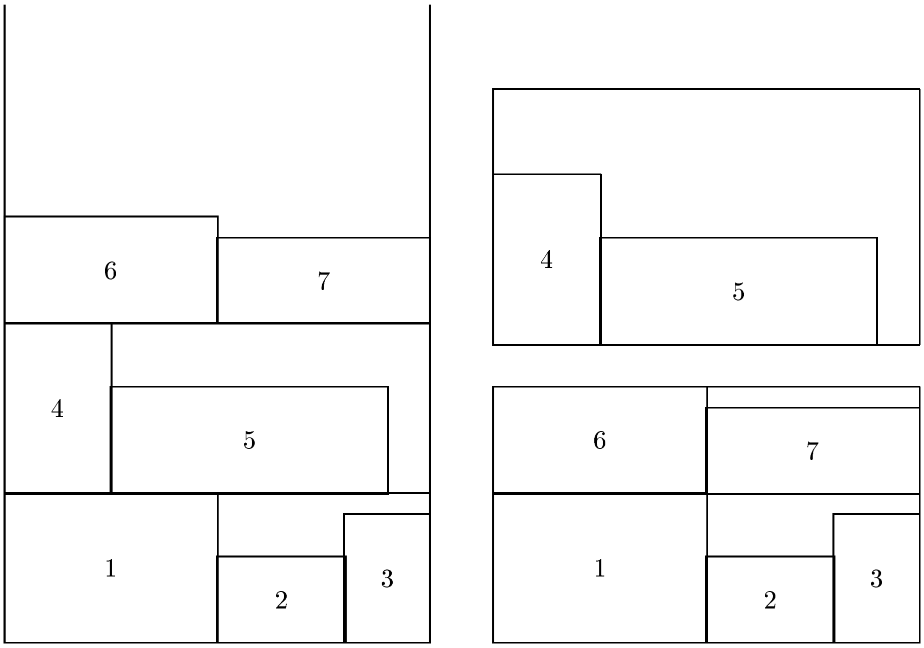

FIG. 3 – Méthode en deux phases

possibles : ordre totalement aléatoire ou ordre déterminé en fonction d'un paramètre lié aux articles (la surface, la hauteur, la largeur ou le périmètre), "pondéré" par une quantité aléatoire.

Proposer plusieurs heuristiques et plusieurs pondérations possibles. Pour les 10 classes d'instances, donner l'écart relatif moyen entre les solutions fournies par vos heuristiques et LB.

6 Des heuristiques en deux temps

Le *strip-packing* $2SP$ est une variante très étudiée du bin-packing en deux dimensions. On dispose d'une liste d'articles en deux dimension identifiés par w_i et h_i et d'une bande unique ("strip") de largeur W et de hauteur infinie. L'objectif est de ranger tous les articles de dans la bande en minimisant la hauteur totale de bande utilisée.

Les heuristiques en deux temps pour $2BP$ fonctionnent de la façon suivante :

- Résolution d'un $2SP$ à niveaux. Comme d'habitude, les articles sont placés les uns après les autres selon un ordre qui reste à déterminer. Le premier niveau est constitué de la base de la bande et les articles sont placés directement dessus. Le niveau suivant est déterminé par une ligne horizontale qui part du sommet de l'article placé le plus haut. Les articles qui restent ne seront jamais placés sous cette ligne. On itère le processus jusqu'à ce que tous les articles soient placés.
- Une fois le placement par niveau effectué, on va "regrouper" les niveaux entre eux pour construire une solution au problème $2BP$. En aucun cas, on ne change le placement des articles d'un même niveau. Par contre plusieurs niveaux peuvent être regroupés dans un même bin. Ce regroupement est alors un "simple" problème de packing ($1BP$)

La figure 3 illustre cette construction en deux temps. La figure de gauche est une solution du strip packing avec 3 niveaux définis respectivement par les lignes qui passent au dessus des articles 1, 4 et 6. La figure de droite est une solution au problème de packing $2BP$ dans lequel les niveaux 1 et 3 du strip packing on été regroupés dasn une même bin.

L'heuristique "First-Fit Decreasing Height" (FFDH) est l'une des nombreuses heuristiques en deux temps que l'on peut construire. Pour la construction d'une solution au problème de strip packing, les articles sont considérés par hauteur décroissante et chaque article est placé, calé à gauche, sur le premier niveau qui peut le contenir. Si aucun des niveaux ne peut le contenir, un nouveau niveau est créé. Pour passer du strip packing à $2BP$, on utilise l'heuristique Best Fit Decreasing (cf., §1.1) pour regrouper en un petit nombre de bins les niveaux.

Coder l'heuristique FFDH et la tester sur toutes les instances. Pour les 10 classes d'instances, donner l'écart relatif moyen entre les solutions fournies par l'heuristique et LB. Proposer des variantes de FFDH et les tester.

D'autres solutions

Les heuristiques originales et/ou les procédures de recherche de l'optimum seront fortement valorisées. Elles seront bien entendu décrites avec concision et précision et seront accompagnées d'une analyse expérimentale.

Remerciements

Merci à François Clautiaux qui nous a permis d'utiliser son manuscrit de thèse pour rédiger ce sujet.

Merci à Edward Coffman (http ://www.ee.columbia.edu/~egc), un spécialiste des problèmes de "packing", à qui le MET (The Metropolitan Museum of Art, New York) a posé une variante de ce problème.