Génération dynamique de couches pour la palettisation

Alexandre LE JEAN[1], Olivier BRIANT[2], Nadia BRAUNER[2]

[1] Fives Syleps, 56100 Lorient, France
alexandre.lejean@fivesgroup.com
[2] Univ. Grenoble Alpes, CNRS, Grenoble INP, G-SCOP, F-38000 Grenoble, France
{olivier.briant@grenoble-inp.fr, nadia.brauner@grenoble-inp.fr}

Mots-clés : *heuristique, génération de colonnes, palettisation.*

1 Introduction

L'entreprise Fives Syleps conçoit des solutions complètement automatisées de plateformes logistiques pour des commerces de moyennes et grandes surfaces. Ces plateformes servent de transit entre d'une part les industries produisant quelques références en grande quantité, et d'autre part les magasins qui demandent à l'inverse un très vaste nombre de références en de faibles quantités.

Divers problèmes de recherche opérationnelle sont à résoudre pour ces plateformes, déterminant les débits d'entrée et de sortie. Dans cet article, nous nous intéressons à l'un des problèmes dimensionnant le débit de sortie : le processus de palettisation. Il s'agit d'un problème d'optimisation consistant à placer un maximum d'objets sur une palette en bois à destination d'un commerce. De nombreuses contraintes industrielles doivent être prises en considération pour ces décisions, dont la stabilité. La Figure 1 illutre un exemple de palette construite avec succès.

FIG. 1 – Exemple de palette construite en production

L'accroissement de la variété des références traitées par l'entrepôt, l'ajout de nouvelles exigences et l'augmentation du débit requis ont nécessité le développement de meilleures pratiques de résolution. Les méthodes précédentes privilégiaient l'élaboration de solutions constituées essentiellement de "couches d'objets", assurant une bonne densité et la validation de contraintes industrielles. Cependant, avec la diversité actuelle des produits ces méthodes n'obtiennent plus d'excellents résultats.

Dans le cadre d'une thèse CIFRE, nous proposons une nouvelle méthode en deux étapes ayant pour objectif de remplacer celles utilisées. La première étape génère des empilements de couches à placer sur la palette, et la seconde ajoute par-dessus des objets à l'aide d'heuristiques basées sur celles de la littérature. Les couches placées sont déterminées par la résolution d'un programme linéaire en nombres entiers (PLNE) fournissant l'une des meilleures combinaisons provenant d'un catalogue donné en paramètre.

Nous détaillons dans cet article la méthode, basée sur la méthode de génération de colonnes, utilisée pour générer le catalogue fourni au PLNE résolu. Nous exposons dans un premier temps le déroulement de la méthode proposée, puis dans un second temps une stratégie pour générer des couches intéressantes selon des coefficients donnés.

2 Un problème d'empilement de couches

Nous souhaitons placer diverses références d'un ensemble I sur une palette en bois. Pour chaque référence $i \in I$, nous dénotons la quantité disponible n_i, la superficie s_i et la hauteur i. Pour y parvenir, nous construisons une solution constituée d'empilements de couches.

L'industriel a sa propre notion de couches classées en deux types : les demi-couches et les couches complètes. Dans les deux cas, il s'agit d'un espace en forme de parallélépipède rectangle contenant un ensemble d'objets qui vérifie certaines propriétés (détaillées Section 3.1). Une solution constituée uniquement de couches est représentée par deux piles de demi-couches placées côte à côte sur un empilement de couches complètes ; chacune des piles peut être vide.

Pour déterminer les couches à placer selon un ensemble de références I, nous proposons de résoudre le PLNE (1), avec $\mathscr{C}_1$ l'ensemble de toutes les couches complètes et $\mathscr{C}_{1/2}$ celui des demi-couches.

$$
\begin{aligned}
z^*_{IP}(\mathscr{C}_1, \mathscr{C}_{1/2}, I) = \min \quad & \sum_{i\in I} v_i \varepsilon_i + \alpha\Big(\sum_{c\in\mathscr{C}_1} h_c \lambda_c^1 + \sum_{c\in\mathscr{C}_{1/2}} \frac{h_c}{2}\lambda_c^{1/2}\Big) && && (1a)\\
s.c. \quad & \varepsilon_i + \sum_{c\in\mathscr{C}_1} q_i^c \lambda_c^1 + \sum_{c\in\mathscr{C}_{1/2}} q_i^c \lambda_c^{1/2} = n_i && \forall i \in I && (1b)\\
& \lambda_c^{1/2} = \lambda_c^D + \lambda_c^G && \forall c \in \mathscr{C}_{1/2} && (1c)\\
& \sum_{c\in\mathscr{C}_{1/2}} h_c(\lambda_c^G - \lambda_c^D) \ \geq\ 0 && && (1d)\\
& \sum_{c\in\mathscr{C}_1} h_c \lambda_c^1 + \sum_{c\in\mathscr{C}_{1/2}} h_c \lambda_c^G \ \leq\ h && && (1e)\\
& \varepsilon_i \in \mathbb{N} && \forall i \in I && (1f)\\
& \lambda_c^1 \in \mathbb{N} && \forall c \in \mathscr{C}_1 && (1g)\\
& \lambda_c^{1/2}, \lambda_c^G, \lambda_c^D \in \mathbb{N} && \forall c \in \mathscr{C}_{1/2} && (1h)
\end{aligned}
$$

Les variables λ_c^1 et $\lambda_c^{1/2}$ indiquent respectivement le nombre de fois où la couche complète $c \in \mathscr{C}_1$ et la demi-couche $c \in \mathscr{C}_{1/2}$ est utilisée, tandis que la variable ε_i indique le nombre d'exemplaires de la référence $i \in I$ qui ne se trouvent pas dans les couches sélectionnées.

L'objectif du PLNE est d'obtenir une solution minimisant les volumes v_i des références $i \in I$ qui ne sont pas placées en couches. Deux solutions avec le même volume d'objet sont ensuite comparées sur la hauteur h_c des couches c sélectionnées. Le coefficient α est choisi suffisamment petit pour respecter l'ordre des objectifs.

Pour chaque référence $i \in I$, le nombre d'exemplaires présents dans une couche c est notée q_i^c. Tous les exemplaires de la référence i doivent être répartis entre les exemplaires non placés (valeur de la variable ε_i) et les exemplaires dans les couches sélectionnées ce qui est vérifié par la contrainte (1b).

Les variables λ_c^D et λ_c^G partitionnent les demi-couches choisies (valeur de la variable $\lambda_c^{1/2}$) en deux piles. Ces variables ainsi que les contraintes (1c), (1d) et (1e) assurent l'existence d'une disposition des couches ne dépassant pas la hauteur maximum autorisée, notée h.

Alors qu'il serait théoriquement réalisable de résoudre le PLNE (1), deux problèmes surviennent. Premièrement, le nombre de variables est trop grand pour que le PLNE puisse être résolu avec une méthode classique. Deuxièmement, les ensembles $\mathscr{C}_1$ et $\mathscr{C}_{1/2}$ ne sont pas connus et les calculer demande trop de temps. Pour pallier ces problèmes, il est possible d'utiliser une méthode de *branch & price* [1].

Le méthode du *branch & price* est une méthode de *branch & bound* qui, pour résoudre la relaxation linéaire du PLNE à chaque nœud, utilise une méthode de *génération de colonnes* [3]. Cependant, le *branch & price* requiert du temps et il n'est pas nécessaire pour le problème industriel de résoudre le PLNE à l'optimalité ; nous proposons donc une résolution approchée : résoudre uniquement la relaxation linéaire du problème au nœud racine du *branch & price*. Cette résolution permet de générer des ensembles de couches intéressants que nous utilisons pour résoudre le PLNE par une méthode de *branch & bound* traditionnelle.

La méthode de génération de colonnes [3] est une méthode itérative générant progressivement un ensemble de variables qui permettent d'exprimer une solution optimale d'un programme linéaire appelé *problème maître*. À chaque itération, le programme linéaire est résolu en utilisant uniquement les variables générées jusqu'alors, prenant le nom de *problème maître restreint*. Les valeurs duales associées à la solution obtenue du problème maître restreint sont utilisées pour connaître le coût réduit des variables encore non générées. Un problème, appelé *problème esclave*, est alors de générer des variables dont le coût réduit est strictement négatif, c'est-à-dire des variables qui ajoutées au *problème maître restreint* permettent d'exprimer une solution de meilleure qualité.

2.1 Problème maître

Classiquement, le problème maître d'une méthode de résolution par la génération de colonnes est la relaxation linéaire du problème à résoudre, le problème (1) dans notre cas. Cependant, la méthode de génération de couches proposée étant une heuristique et la résolution de la relaxation linéaire du PLNE (1) n'étant pas requise, nous nous proposons de relâcher quelques contraintes supplémentaires pour le problème maître.

Les variables λ_c^D et λ_c^G ainsi que les contraintes (1c), (1d) et (1e) assurent l'existence d'une disposition des couches ne dépassant pas la hauteur maximum autorisée. Cependant, pour les instances industrielles données, l'existence d'une telle disposition est rare. Pour le problème maître, nous choisissons de relâcher ces variables et contraintes qui seront toutefois considérées lors de la résolution finale du PLNE (1).

Dans le problème maître que nous considérons, les variables sont liées uniquement par les contraintes (1b) ; une par référence $i \in I$. En particulier, deux contraintes sont liées si les références associées peuvent se trouver dans une même couche. Cependant, la différence de hauteur entre chaque paire d'objets présents dans une couche ne peut pas excéder une certaine valeur Δ donnée en paramètre par l'entreprise. En nous basant sur cette propriété, nous proposons de créer une partition de l'ensemble des références I en plusieurs sous ensembles I^k. Les ensembles de variables associées à chacune des contraintes sont disjoints et le problème maître peut être décomposé en plusieurs sous-problèmes maîtres chacun associé à un un ensemble I_k.

2.2 Problèmes esclaves

À chaque itération est résolu le problème maître restreint à des sous-ensembles de couches complètes et demi-couches notées respectivement $\tilde{\mathscr{C}}_1 \subseteq \mathscr{C}_1$ et $\tilde{\mathscr{C}}_{1/2} \subseteq \mathscr{C}_{1/2}$. À la suite de cette résolution des variables de plus petits coûts réduits doivent être déterminées : il s'agit des *problèmes esclaves*.

Les fonctions de coût réduit des variables λ_c^1 et $\lambda_c^{1/2}$, notées respectivement $\gamma_c^1(u)$ et $\gamma_c^{1/2}(u)$, sont données par les équations (2) avec u_i les valeurs duales des contraintes (1b). Les *problèmes esclaves* qui dépendent de ces fonctions sont décrits par les équations (3).

$$\gamma_c^1(u) = \alpha H_c - \sum_{i \in I} u_i q_i^c \qquad \forall c \in \mathscr{C}_1 \tag{2a}$$

$$\gamma_c^{1/2}(u) = \alpha \frac{H_c}{2} - \sum_{i \in I} u_i q_i^c \qquad \forall c \in \mathscr{C}_{1/2} \tag{2b}$$

$$\gamma^1_{min}(u) = \min_{c\in\mathscr{C}_1} \gamma^1_c(u) \tag{3a}$$

$$\gamma^{1/2}_{min}(u) = \min_{c\in\mathscr{C}_{1/2}} \gamma^{1/2}_c(u) \tag{3b}$$

Dès lors que les deux valeurs $\gamma^1_{min}(u)$ et $\gamma^{1/2}_{min}(u)$ sont positives ou nulles, nous avons la preuve que la solution optimale du problème maître restreint aux sous-ensembles $\tilde{\mathscr{C}}_1$ et $\tilde{\mathscr{C}}_{1/2}$ est aussi optimale pour le problème maître (i.e. avec les ensembles $\mathscr{C}_1$ et $\mathscr{C}_{1/2}$). Si tel est le cas pour l'itération en cours, l'heuristique est terminée. Sinon, des couches de coût réduit négatif sont rajoutées aux sous-ensembles $\tilde{\mathscr{C}}_1$ et $\tilde{\mathscr{C}}_{1/2}$ pour l'itération suivante. Le problème maître restreint aux nouveaux sous-ensembles $\tilde{\mathscr{C}}_1$ et $\tilde{\mathscr{C}}_{1/2}$ pourra alors être résolu de nouveau pour mettre à jour les coefficients u.

3 Génération de couches intéressantes

Nous disposons d'un ensemble de références I ainsi que des coefficients u et de deux fonctions d'évaluation $\gamma^1_c(u)$ et $\gamma^{1/2}_c(u)$ décrites par les équations (2). Nous souhaitons générer une couche avec la plus petite valeur négative possible selon sa fonction associée ou prouver qu'il n'existe aucune couche avec une valeur négative. Nous présentons dans cette partie comment générer de telles couches et ainsi résoudre les problèmes esclaves (3).

3.1 Reformulation du problème esclave

Puisque les ensembles $\mathscr{C}_1$ et $\mathscr{C}_{1/2}$ ne sont pas connus, nous proposons de reformuler les problèmes esclaves (3). Une couche est caractérisée par trois éléments : ses dimensions, les quantités d'exemplaires de chaque référence et la disposition des objets. Pour le problème esclave, la disposition des objets n'influence pas la qualité de la solution, aussi les couches seront uniquement différenciées par les quantités de chaque référence objets $i \in I$ présents et ses dimensions sur X, Y et Z notés respectivement l, w et h.

Par contraintes industrielles, les dimensions (l,w) d'une couche complète et d'une demi-couche sont respectivement (L_1, W_1) et $(L_{1/2}, W_{1/2})$. La hauteur h quant à elle doit être égale à la hauteur d'un des objets présents dans la couche. Nous proposons de créer un sous-problème esclave (4) par hauteurs possibles pour les couches et les demi-couches.

$$\gamma(u,l,w,h) = \max \sum_{i\in I_h} u_i q_i \tag{4a}$$

$$s.c. \quad q_i \le n_i \qquad \forall i \in I_h \tag{4b}$$

$$\sum_{i\in I_h} s_i q_i \ge 80\% lw \tag{4c}$$

$$chevauchement(q,l,w) \tag{4d}$$

$$\sum_{\{i\in I_h | h_i = h\}} q_i \ge 1 \tag{4e}$$

$$q_i \in \mathbb{N} \qquad \forall i \in I_h \tag{4f}$$

Ce sous-problème esclave est lié aux problèmes esclaves (3) par les équations (5) ; le coefficient α est le même que celui utilisé pour la fonction objectif du PLNE (1).

$$\gamma^1_{min}(u) = \min_h \alpha h - \gamma(u, L_1, W_1, h) \tag{5a}$$

$$\gamma^{1/2}_{min}(u) = \min_h \alpha \frac{h}{2} - \gamma(u, L_{1/2}, W_{1/2}, h) \tag{5b}$$

La variable q_i indique le nombre d'exemplaires de la référence $i \in I$ présents dans la couche qui ne peut pas excéder la quantité disponible n_i, assurée par la contrainte (4b).

L'objectif de ce PLNE est d'obtenir une couche maximisant le profit u_i des objets placés. À noter que ce profit peut être de valeur négative, nulle ou positive.

Pour une couche de hauteur h, les objets présents doivent avoir une hauteur comprise dans l'interval $[h-\Delta,\ h]$. Nous dénotons $I_h \subseteq I$ les références vérifiant cette propriété.

La somme de la superficie des objets, notée s_i, doit être au moins égale à 80% de la superficie de la couche, ce qui est vérifié par la contrainte (4c).

Dans une couche, les objets doivent être placés sans chevauchement. Nous dénotons $chevauchement(q,l,w)$ (4d) la contrainte assurant que cela soit possible pour les quantités indiquées par le vecteur de quantité q et pour une couche de longueur l et largeur w.

Une couche de hauteur h doit contenir au moins un objet cette hauteur, ce qui est assuré par la contrainte (4e).

3.2 Résolution des sous-problèmes esclaves

Le sous-problème esclave (4) est une variante d'un problème NP-difficile au sens fort connu sous le nom de *two-dimensional knapsack problem* [4]. Ici, les différences avec le problème classique sont la présence d'objets au profit négatif et d'avoir une superficie minimum (4c). Individuellement, ces différences ne sont pas traitées dans la littérature.

Dans la méthode de génération de colonnes, trouver une solution optimale n'est pas nécessaire. Il est préférable d'obtenir rapidement des solutions de bonne qualité, à une optimale nécessitant beaucoup plus de temps. La littérature est riche en méthodes heuristiques pour le *two-dimensional knapsack problem*. Nous choisissons d'utiliser des méthodes d'un temps de calcul polynomial [5], et une autre qui propose de bons résultats pour une durée de calcul raisonnable [7]. Lorsque toutes ces méthodes ont échoué à obtenir une couche de coût réduit négatif, nous utilisons une méthode exacte.

De nombreuses méthodes exactes existent dans la littérature [4]. Notre choix s'est porté sur une méthode basée sur la décomposition de Benders [2]. Cette méthode, itérative, résout à chaque itération k le problème (4) en relâchant la contrainte qui interdit des chevauchements (4d) ; une solution q^* est alors obtenue. Suite à cette résolution, un oracle retournant vrai ou faux est utilisé pour déterminer si q^* valide les contraintes relâchées, auquel cas q^* est optimale et la méthode est terminée. Autrement, des contraintes interdisant d'obtenir q^* sont ajoutées pour les itérations suivantes et une borne inférieure $\psi(u,l,w,h)$ sur la valeur $\gamma(u,l,w,h)$ est connue.

Nous proposons différentes stratégies pour faire un usage plus efficace des résolutions de l'oracle couteuses en temps. Ces stratégies utilisent l'oracle pour déterminer d'autres solutions que q^* telles que si une ne valide pas la contrainte relâchée, alors q^* non plus. De plus, chaque solution testée qui valide les contraintes est une couche réalisable dont la valeur de la fonction objectif notée $\chi(u,l,w,h)$ est une borne supérieure sur la valeur $\gamma(u,l,w,h)$.

La méthode proposée est terminée lorsque l'égalité est obtenue pour la relation (6). Nous choisissons d'interrompre prématurément la méthode dès lors que la couche associée à la borne $\chi(u,l,w,h)$ est une couche de coût réduit négatif. De même, si la borne $\psi(u,l,w,h)$ est positive ou nulle, alors nous avons la preuve que ce sous problème esclave n'admet aucune couche de coût réduit négatif, et la méthode est interrompue.

$$\psi(u,l,w,h) \leq \gamma(u,l,w,h) \leq \chi(u,l,w,h) \tag{6}$$

4 Conclusions et perspectives

Nous nous sommes intéressés à un problème industriel de palettisation dans le cadre de la gestion d'un entrepôt. Ce problème s'étant complexifié avec l'évolution des attentes, les méthodes utilisées jusqu'alors ne permettaient plus d'obtenir les résultats escomptés. Nous avons présentée dans cet article une partie d'un nouvel algorithme produisant des solutions constituées essentiellement de couches.

Des expérimentations sur 176 instances industrielles de référence ont permis de démontrer l'efficacité du nouvel algorithme, lequel place plus de volume que la méthode précédemment utilisée. Se référer à la thèse [6] pour les résultats expérimentaux. Outre le volume, l'industriel évalue les solutions selon un calcul de probabilité de chute d'objets. Pour cette évaluation, notre méthode obtient des résultats nettement supérieurs à ceux des autres méthodes. Cela peut s'expliquer notamment par la souplesse offerte par la structure en couches, et la possibilité de positionner les objets à l'intérieur de chaque couche de plusieurs façons.

L'approche par génération de colonnes que nous avons proposée diffère de celles de la littérature grâce à la structure même du problème que nous avons résolu ; en effet nous avons pu décomposer le problème maître et exprimer le problème esclave en plusieurs qui peuvent être résolus indépendamment.

En guise de perspectives, et pour améliorer la méthode proposée, il pourrait être intéressant de paralléliser les résolutions des différents sous-problèmes, puisqu'ils sont indépendants les uns des autres. De plus, dans la méthode que nous avons proposée, nous partons toujours d'un catalogue de couches vide. Or, une analyse fine des données permettrait probablement d'identifier une liste d'objets très fréquemment demandés par les clients et présente dans presque toutes les instances traitées. Ainsi, il est possible d'initialiser notre catalogue avec des couches précalculées constituées de ces objets. Cette même idée pourrait aussi être utilisée lors de la résolution des sous-problèmes esclaves. Plusieurs objets ont les mêmes dimensions et ne se différencient que par leur hauteur. Il pourrait être envisageable de sauvegarder en mémoire certains résultats du problème résolu par l'oracle de la décomposition de Benders, pour les réutiliser dans plusieurs esclaves.

Références

[1] Cynthia Barnhart, Ellis L Johnson, George L Nemhauser, Martin WP Savelsbergh, and Pamela H Vance. Branch-and-price : Column generation for solving huge integer programs. *Operations research*, 46(3) :316–329, 1998.

[2] Jacques F Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische mathematik*, 4(1) :238–252, 1962.

[3] George B Dantzig and Philip Wolfe. Decomposition principle for linear programs. *Operations research*, 8(1) :101–111, 1960.

[4] Manuel Iori, Vinícius L de Lima, Silvano Martello, Flávio K Miyazawa, and Michele Monaci. Exact solution techniques for two-dimensional cutting and packing. *European Journal of Operational Research*, 289(2) :399–415, 2021.

[5] Jukka Jylänki. A thousand ways to pack the bin – a practical approach to two-dimensional rectangle bin packing, 2010.

[6] Alexandre Le Jean. *Optimisation de plans de palettisation hétérogène*. PhD thesis, Université Grenoble Alpes, 2021.

[7] Lijun Wei, Qian Hu, Stephen CH Leung, and Ning Zhang. An improved skyline based heuristic for the 2d strip packing problem and its efficient implementation. *Computers & Operations Research*, 80 :113–127, 2017.