



HAL
open science

Méthodes de décomposition pour l'optimisation discrète

Nora Touati, Lucas Létocart, Anass Nagih

► **To cite this version:**

Nora Touati, Lucas Létocart, Anass Nagih. Méthodes de décomposition pour l'optimisation discrète. 2007. hal-00147704

HAL Id: hal-00147704

<https://hal.science/hal-00147704>

Preprint submitted on 20 May 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Méthodes de décomposition pour l'optimisation discrète

Nora Touati, Lucas Létocart

LIPN UMR CNRS 7030 - Université Paris 13
99, Avenue Jean-Baptiste Clément
93430 Villetaneuse France
{nora.touati,lucas.letocart}@lipn.univ-paris13.fr

Anass Nagih

LITA EA 3097 - Université Paul Verlaine
Ile du Saulcy 57045 Metz Cedex 01
anass.nagih@univ-metz.fr

Janvier 2007

Rapport de recherche

2007-1

Résumé

Plusieurs problèmes pratiques sont modélisés par des programmes en nombres entiers. La difficulté de leur résolution réside dans la caractérisation du domaine réalisable. Face à cette difficulté, les chercheurs se sont orientés vers des approximations de ce domaine en décomposant le problème. La décomposition se traduit généralement par la relaxation des contraintes d'intégrité sur un sous-ensemble de contraintes.

Les méthodes de décomposition diffèrent sur la qualité de la borne obtenue et des méthodes de résolution utilisées ainsi que leur complexité. Nous présentons dans ce rapport quelques unes de ces méthodes, analyser leurs points forts et faibles et enfin établir une synthèse montrant les liens existants entre certaines méthodes jugées différentes.

Abstract

Many practical problems are modelled by integer programs. The difficulty of their resolution lies in the characterization of the feasible domain. Face to this difficulty, the researchers directed themselves towards approximations of this domain by decomposing the problem. The decomposition generally results in the relaxation of the integrality conditions on a subset of constraints.

Decomposition methods differ from the bound quality and from the resolution methods used like their complexity. We present in this report some of these methods, analyze their strong and weak points and finally, establish a synthesis showing the relation between some methods considered to be different.

Introduction

Les méthodes de décomposition sont généralement utilisées dans un processus de résolution de Programmes Linéaires en Nombres Entiers (*PLNE*), elles consistent à décomposer le problème en sous-problèmes plus faciles à résoudre. La décomposition entraîne la relaxation de certaines contraintes et permet donc de trouver une borne sur la valeur de (*PLNE*). Les bornes obtenues sont ensuite utilisées pour guider et accélérer la phase d'exploration par un algorithme de type branch-and-bound [26], par exemple.

Toutes les méthodes de décomposition se basent sur le même principe : le problème traité est décomposé en un ou plusieurs sous-problèmes, généralement coordonnés par un programme linéaire appelé le problème maître.

Les méthodes de résolution des nouvelles formulations décomposées alternent entre une procédure de calcul d'un majorant (en supposant que le problème (*PLNE*) a pour objectif une maximisation) par la résolution du problème maître, et une procédure qui améliore l'approximation courante par la résolution des sous-problèmes. Le principe est

donc d'approcher l'enveloppe convexe des solutions réalisables \mathcal{P} du (*PLNE*). Une approximation classique est celle de la *relaxation continue*, qui consiste simplement à relâcher les contraintes d'intégrité du (*PLNE*). La borne obtenue par cette relaxation peut s'avérer très éloignée de la valeur optimale. Cependant, elle peut être améliorée par la génération dynamique de nouvelles informations.

Nous distinguons deux types d'approximation. *Les méthodes d'approximation externe [1]* améliorent l'approximation en générant dynamiquement des plans de coupes contenant \mathcal{P} et que l'on appelle aussi inégalités valides ; dans cette approche, les inégalités valides sont générées par un programme linéaire appelé *le problème de séparation*. *Les méthodes d'approximation interne* consistent à améliorer l'approximation en générant dynamiquement les points extrêmes d'un polyèdre contenu dans \mathcal{P} , dans ce cas, chaque point extrême est généré par un programme linéaire appelé *le sous-problème*.

Nous présentons d'abord le principe général de la décomposition, puis quelques techniques de décomposition et les méthodes de résolution associées, et enfin une synthèse des liens existants entre ces méthodes.

1 Principe de la décomposition

Le principe est de décomposer les contraintes du problème en un ou plusieurs blocs indépendants qui constituent les contraintes des sous-problèmes (ou bien des problèmes de séparation). Ces derniers sont coordonnés par le problème maître.

Soit le programme linéaire suivant :

$$(PLNE) \begin{cases} z_{PLNE} = \max c^t x \\ Ax \leq a & (m \text{ contraintes}) \\ Bx \leq b & (p \text{ contraintes}) \\ x \in \mathbb{N}^n \end{cases}$$

Où $A \in \mathbb{R}^{m \times n}$, $B \in \mathbb{R}^{p \times n}$, $a \in \mathbb{R}^m$, $b \in \mathbb{R}^p$. La résolution de (*PLNE*) peut nécessiter l'utilisation d'une procédure de branch-and-bound, cette procédure est d'autant plus efficace lorsque l'on dispose de bonnes bornes sur la solution, ce qui peut diminuer considérablement la taille de l'arbre d'exploration. La procédure la plus simple est de résoudre la relaxation continue de (*PLNE*) que l'on nomme (*PL*) et qui est obtenue en relâchant les contraintes d'intégrité de (*PLNE*), la borne (*PL*) est donnée par :

$$z_{PL} = \max_{x \in \mathbb{R}^n} \{c^t x \mid Ax \leq a, Bx \leq b\} = \max_{x \in \mathcal{Q}} \{c^t x\},$$

Il est clair que $z_{PL} \geq z_{PLNE}$ car $\mathcal{P} \subseteq \mathcal{Q}$. En relâchant les contraintes d'intégrité, on obtient généralement un problème plus facile à résoudre, mais z_{PL} peut être considérablement éloignée de z_{PLNE} , nous devons donc considérer des procédures plus efficaces.

Pour améliorer la borne z_{PL} , les méthodes de décomposition construisent un nouveau polyèdre d'optimisation qui intersecte \mathcal{Q} pour obtenir une meilleure approximation. Ce nouveau polyèdre possède une description de taille exponentielle, nous devons donc générer des portions de sa description de manière dynamique.

Notons $\text{conv}(\cdot)$ une enveloppe convexe. Soit $\mathcal{P}' = \text{conv}(\{x \in \mathbb{N}^n | Ax \leq a\})$, $\mathcal{Q}' = \{x \in \mathbb{R}^n | Ax \leq a\}$, $\mathcal{P}'' = \text{conv}(\{x \in \mathbb{N}^n | Bx \leq b\})$ et $\mathcal{Q}'' = \{x \in \mathbb{R}^n | Bx \leq b\}$

On peut donc écrire : $\mathcal{Q} = \mathcal{Q}' \cap \mathcal{Q}''$ et $\mathcal{P} = \mathcal{P}' \cap \mathcal{P}''$.

La décomposition consiste à relâcher les contraintes d'intégrité sur un sous-ensemble de contraintes, cela revient donc à résoudre ($PLNE$) sur le domaine $\mathcal{D1} = \mathcal{P}' \cap \mathcal{Q}''$ ou bien $\mathcal{D2} = \mathcal{P}'' \cap \mathcal{Q}'$ (voir la figure 1). Dans ces deux cas, la décomposition n'a un sens que si $\mathcal{D1} \subset \mathcal{Q}$ (resp. $\mathcal{D2} \subset \mathcal{Q}$), de telle sorte à obtenir une borne au moins aussi bonne que z_{PL} s'il existe un algorithme efficace d'optimisation sur le domaine \mathcal{P}' (resp. \mathcal{P}'').

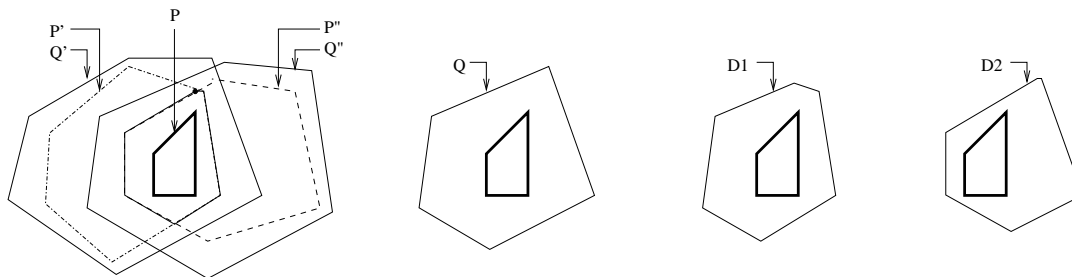


FIG. 1 – Illustration de différentes approximations du domaine \mathcal{P}

2 Relaxation lagrangienne

La relaxation lagrangienne [15, 22] est appliquée lorsque l'on reconnaît dans la matrice des contraintes, des contraintes difficiles dont la relaxation engendre des problèmes plus faciles à résoudre, ou bien pour lesquels on dispose d'outils efficaces de résolution.

Cette méthode consiste à relâcher certaines contraintes jugées compliquantes. Les contraintes relâchées sont réinjectées dans la fonction objectif, pondérées par des coefficients que l'on appelle *les multiplicateurs de Lagrange*.

Considérons le problème ($PLNE$) et notons $X = \{x \in \mathbb{N}^n | Bx \leq b\}$, un ensemble supposé borné. Admettons que “ $Ax \leq a$ ” sont les contraintes “difficiles” et “ $x \in X$ ” les contraintes “faciles”. On associe alors aux contraintes “difficiles” un vecteur de multiplicateurs $u \in \mathbb{R}_+^m$ et l'on définit la fonction lagrangienne correspondant à cette modélisation par :

$$L_x(u) = c^t x + u^t (a - Ax) \quad (1)$$

On pénalise ainsi dans l'objectif la violation des contraintes $A_i x \leq a_i$, $i = 1, \dots, m$ par l'intermédiaire des multiplicateurs de Lagrange $u_1, \dots, u_m \geq 0$.

Si on maximise, pour $u \in \mathbb{R}_+^m$ donné $L_x(u)$ sur $x \in X$ (ou sur $x \in \text{conv}(X)$), on obtient une solution optimale notée $x(u)$. C'est la valeur de cette solution qui définit la fonction duale, $\Theta(u)$:

$$\begin{aligned} \Theta(u) &= \max_{x \in X} L_x(u) \\ &= \max_{x \in \text{conv}(X)} L_x(u) \\ &= (c - u^t A)x(u) + u^t a \end{aligned} \quad (2)$$

Le dual de la relaxation lagrangienne consiste à minimiser la fonction duale :

$$(DL) \quad \begin{cases} z_{DL} &= \min_{u \in \mathbb{R}_+^m} \Theta(u) \end{cases}$$

Remarque 1

(a) Généralement, la complexité du dual de la relaxation lagrangienne est fortement liée à la complexité de calcul de $\Theta(u)$.

(b) Considérons la fonction $\phi(x) = \min_{u \in \mathbb{R}_+^m} L_x(u)$. Trivialement, $\phi(x)$ vérifie :

$$\phi(x) = \begin{cases} cx & \text{si } Ax \leq a \\ -\infty & \text{sinon} \end{cases} \quad \begin{aligned} &(a - Ax \geq 0 \Rightarrow \text{valeur optimale obtenue pour } u = 0) \\ &(\exists i/a_i - A_i x < 0 \Rightarrow u_i \text{ direction d'infinitude}) \end{aligned}$$

Le problème (PLNE) n'est autre que la maximisation de $\phi(x)$ sur X . Cette formulation établit la relation ci-dessous entre (PLNE) et (DL) :

$$\begin{aligned} (PLNE) &\equiv \begin{cases} \max_{x \in X} \phi(x) \end{cases} && \equiv \begin{cases} \max_{x \in X} \min_{u \in \mathbb{R}_+^m} L(x, u) \end{cases} \\ &\equiv \begin{cases} \min_{u \in \mathbb{R}_+^m} \max_{x \in X} L(x, u) \end{cases} && \equiv \begin{cases} \min_{u \in \mathbb{R}_+^m} \Theta(u) \end{cases} \equiv (DL) \end{aligned}$$

(c) Par définition, $\Theta(u) \geq L_x(u)$, $\forall x \in X$. De plus, si $Ax \leq a$ (solution x réalisable), alors $L_x(u) \geq c^t x \forall x \in X$. La valeur $\Theta(u)$ constitue donc pour toute valeur de $u \in \mathbb{R}_+^m$ un majorant de la valeur optimale du problème (PLNE). Le problème dual (DL) revient ainsi à trouver le plus petit majorant, i.e. le majorant qui soit le plus proche de la valeur optimale du problème d'origine.

(d) Il existe plusieurs lagrangiens pour le problème (PLNE) (i.e plusieurs choix possibles des contraintes à relaxer). Pour la sélection d'un lagrangien approprié, un compromis doit être établi entre la complexité de calcul de $\Theta(u)$ et la qualité du

saut de dualité¹.

2.1 Résolution du dual langrangien par la méthode de sous-gradient

Cette méthode consiste à mettre à jour itérativement les variables du problème (DL) (les multiplicateurs lagrangiens). Chaque itération k consiste en deux étapes :

1) Pour u^k (le multiplicateur lagrangien à la $k^{\text{ème}}$ itération), calculer $\Theta(u^k)$ pour obtenir une solution primale (x^k) et le sous-gradient de Θ au point u^k que l'on appelle g^k .

2) Calculer $u^{k+1} = (u^k + t^k g^k)$, où $t^k > 0$ est le pas de déplacement.

La fonction Θ étant convexe et linéaire par morceaux, g^k peut donc ne pas être une direction de descente, on doit alors tester lors de chaque itération k si le majorant obtenu z_{DL}^k de z_{DL} est inférieur à \bar{z}_{DL} , le meilleur majorant courant.

Si la séquence $t^k, k \geq 0$ satisfait : $t^k \rightarrow 0, \sum_{k=0}^{\infty} t^k = \infty$ alors $\lim_{k \rightarrow \infty} z_{DL}^k = z_{DL}$ [34]. Le calcul du pas de déplacement proposé par [35] est :

$$t^k = \alpha^k \frac{z_{DL} - z_{DL}^k}{|\gamma^k|^2}$$

où α^k est un paramètre qui vérifie $0 < \epsilon_1 \leq \alpha^k \leq 2 - \epsilon_2$ (ϵ_1 et ϵ_2 étant deux petits réels) et γ^k est la direction de déplacement. Sous ces conditions, la séquence de valeurs $z_{DL}^k, k \geq 1$, obtenues au cours des itérations de l'algorithme, converge vers z_{DL} . Comme la valeur de z_{DL} n'est pas connue, on utilise une approximation z_{DL}^k , mais on perd ainsi la garantie de convergence.

2.2 Résolution du dual langrangien par la méthode des plans coupants

Nous pouvons considérer une version équivalente au problème ($PLNE$) qui est le problème de séparation pour \mathcal{P} . Etant donné $x \in \mathbb{R}^n$, le problème de séparation pour \mathcal{P} consiste à décider si $x \in \mathcal{P}$ ou non et à déterminer $\alpha \in \mathbb{R}^n$ et $\beta \in \mathbb{R}$ tels que $\alpha^T y \geq \beta, \forall y \in \mathcal{P}$. S'il existe un couple $(\alpha, \beta) \in \mathbb{R}^{n+1}$ tel que $\alpha^T x < \beta$, alors $\alpha^T y \geq \beta, \forall y \in \mathcal{P}$ est une inégalité valide pour \mathcal{P} qui est dite violée par $x \in \mathbb{R}^n$. La forme optimisée du problème de séparation est de trouver l'inégalité la plus violée.

Cette méthode, appliquée à notre problème, consiste à résoudre la formulation linéaire du problème dual (DL) donnée ci-dessous :

¹On appelle un saut de dualité l'écart entre les valeurs optimales des problèmes ($PLNE$) et (DL) obtenues à une itération donnée.

$$\begin{aligned}
(DL) &\equiv \min_{u \in \mathbb{R}_+^m} \max_{x \in X} L_x(u) \\
&\equiv \min_{u \in \mathbb{R}_+^m} \max_{x \in X} cx + u^t(a - Ax) \\
&\equiv \min_{u \in \mathbb{R}_+^m} \max_{j \in J} cx^j + u^t(a - Ax^j) \\
&\equiv \begin{cases} \min & \theta \\ & \theta \geq cx^j + u^t(a - Ax^j), \quad j \in J \quad (*) \\ & u \in \mathbb{R}_+^m, \theta \in \mathbb{R} \end{cases}
\end{aligned}$$

où J est l'ensemble des indices des points extrêmes de $X = \{Bx \leq b, x \in \mathbb{N}^n\}$. Notons que la taille de la nouvelle formulation est exponentielle (égale à la cardinalité de l'ensemble J), et que ces contraintes sont des supports de la fonction Θ .

La méthode des plans coupants consiste à considérer le problème dual avec un sous-ensemble des contraintes de type $(*)$ (éventuellement vide), soit (DL') ce problème. Sa résolution fournit une solution duale \hat{u} . Pour vérifier si elle est optimale, on calcule la valeur de $\Theta(\hat{u})$. D'après la remarque (1-c), si la valeur de (DL') coïncide avec $\Theta(\hat{u})$, alors la solution courante est optimale, sinon, la contrainte $\theta \geq cx(\hat{u}) + u^t(a - Ax(\hat{u}))$ est rajoutée à (DL') qui est réoptimisé.

Remarque 2 *Le problème (DL) peut être reformulé comme suit :*

$$\begin{aligned}
\min & \theta \\
& \theta \geq \Theta(u^j) - (g^j)^t(u - u^j), \quad j \in J \\
& (\theta, u) \in \mathbb{R} \times \mathbb{R}_+^m
\end{aligned}$$

où g^j est le sous-gradient de Θ au point u^j et u^j vérifie $x^j = x(u^j)$. De manière similaire à la méthode de sous-gradient, le principe est de calculer lors de chaque itération une direction de déplacement, les deux méthodes diffèrent par le calcul du pas de déplacement et par le critère d'arrêt pris en compte.

3 Décomposition lagrangienne

La décomposition lagrangienne [19, 24] est utilisée lorsque la conjonction entre deux blocs de contraintes complique le problème. Pour le problème $(PLNE)$, la conjonction des contraintes $Ax \leq a$ et $Bx \leq b$ peut rendre sa résolution difficile. Cette méthode consiste alors à dupliquer les variables et à construire un problème équivalent :

$$(PLNE) \equiv \begin{cases} \max cx \\ Bx \leq b, \quad x \in \mathbb{N}^n \\ x = y \\ Ay \leq a, \quad y \in Y \end{cases}$$

où $Y \subset \mathbb{N}^n$. En relâchant la contrainte $x = y$ suivant le multiplicateur $u \in \mathbb{R}^n$, on obtient le lagrangien

$$L_{x,y}(u) = \max\{(c - u^t)x + u^t y \mid Ay \leq a, y \in Y, Bx \leq b, x \in \mathbb{N}^n\}$$

qui peut être formulé comme suit

$$L_{x,y}(u) = L_x(u) + L_y(u)$$

où

$$L_x(u) = \max\{(c - u^t)x \mid Bx \leq b, x \in \mathbb{N}^n\} \quad \text{et}$$

$$L_y(u) = \max\{u^t y \mid Ay \leq a, y \in Y\}$$

Le dual de la décomposition lagrangienne est donné par :

$$(DDL) \begin{cases} z_{DDL} = \min L_{x,y}(u) \\ u \in \mathbb{R}^n \end{cases}$$

Remarque 3

- (a) La décomposition lagrangienne est une relaxation lagrangienne sur un problème transformé qui décompose le problème initial en une série de sous-problèmes.
- (b) La valeur optimale du problème (DDL) est égale à la valeur optimale du problème

$$\max\{cx \mid x \in Z_x \cap Z_y\}$$

où $Z_x = \text{conv}\{x \in \mathbb{N}^n \mid Bx \leq b\}$ et $Z_y = \text{conv}\{y \in Y \mid Ay \leq a\}$, la figure 2 représente ces deux domaines.

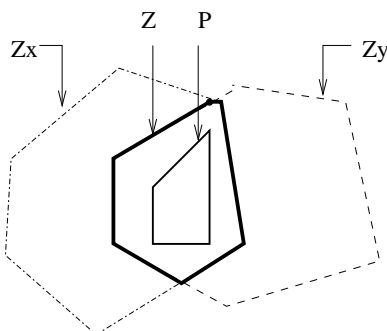


FIG. 2 – Illustration de l'approximation du domaine \mathcal{P} avec la décomposition lagrangienne.

Cette méthode a été appliquée pour la résolution du problème localisation avec capacités dans [24], les bornes obtenues sont de bonne qualité, mais la résolution est coûteuse en temps. Nous exploitons dans ce qui suit la génération de colonnes pour résoudre le dual de la substitution lagrangienne.

Les lagrangiens $L_x(u)$ et $L_y(u)$ peuvent être réécrits comme suit :

$$\begin{aligned} L_x(u) &= \{\min w_1 \mid w_1 \geq (c - u^t)x, x \in Z_x\} \\ &= \{\min w_1 \mid w_1 \geq (c - u^t)x_k, k \in K\} \end{aligned}$$

et

$$\begin{aligned} L_y(u) &= \{\min w_2 | w_2 \geq u^t y, y \in Z_y\} \\ &= \{\min w_2 | w_2 \geq u^t y_l, l \in L\} \end{aligned}$$

où K (resp. L) désignent l'ensemble des points extrêmes du polyèdre Z_x (resp. Z_y). Supposons que les ensembles Z_x et Z_y sont convexes et bornés, le problème (DDL) peut être alors linéarisé comme suit :

$$(DDL) \equiv \begin{cases} \min w_1 + w_2 \\ w_1 \geq (c - u^t)x, & x \in Z_x \\ w_2 \geq u^t y, & y \in Z_y \\ w_1, w_2 \in \mathbb{R} \\ u \in \mathbb{R}^n \end{cases} \equiv \begin{cases} \min w_1 + w_2 \\ w_1 \geq (c - u^t)x_k, & k \in K \\ w_2 \geq u^t y_l, & l \in L \\ w_1, w_2 \in \mathbb{R} \\ u \in \mathbb{R}^n \end{cases}$$

En posant $w = w_1 + w_2$ et en agrégeant les contraintes de cette formulation, on obtient :

$$(DDLA) \equiv \begin{cases} \min w \\ w + u^t(x^{(k)} - y^{(l)}) \geq cx^{(k)}, k \in K, l \in L \\ u \in \mathbb{R}^n, w \in \mathbb{R} \end{cases}$$

En ne considérant que les multiplicateurs $u \in \mathbb{R}^n$ de la forme $\lambda A \in \mathbb{R}^n$, où λ est un vecteur de \mathbb{R}^m , et en substituant dans (DDL) , on retombe sur (DL) . Autrement dit,

$$z_{DDLA} = \min_{u \in \mathbb{R}^n} L_{x,y}(u) \leq \min_{\lambda^t \in \mathbb{R}^m} L_{x,y}(\lambda^t A) = \min_{\lambda \in \mathbb{R}^m} v(DL(\lambda)) = z_{DL}.$$

4 Substitution lagrangienne

La substitution lagrangienne [20, 31, 37] amène également à une décomposition de $(PLNE)$, en alliant une copie explicite et une copie implicite de variables. Leur relaxation met en jeu deux vecteurs de multiplicateurs dont l'addition des dimensions est comprise entre m et n .

On envisage une bipartition du vecteur $x = (x', x'')$ telle que $x' \in \mathbb{R}^q$ avec $q < n - m$. On considère le vecteur $y = (y', y'')$ composé d'une copie explicite y' de x' et une copie implicite y'' de x'' , on obtient le problème suivant équivalent au problème initial, dans lequel $A = (A', A'')$:

$$(PLNE) \equiv \begin{cases} \max cx \\ Ay \leq a, & y \in Y \\ y' = x' \\ A''y'' \leq A''x'' \\ Bx \leq b, & x \in \mathbb{N}^n \\ Y \supseteq \mathbb{N}^n \end{cases}$$

En relâchant les contraintes $y' = x'$ (avec un multiplicateur $u' \in \mathbb{R}^q$) et $A''y'' \leq A''x''$ (avec un multiplicateur $u'' \in \mathbb{R}_+^m$), on obtient le lagrangien

$$L(u', u'') = L_x(u', u'') + L_y(u', u'')$$

avec

$$L_x(u', u'') = \max_{\substack{Bx \leq b \\ x \in \mathbb{N}^n}} cx - (u')^t x' - (u'')^t A'' x''$$

$$L_y(u', u'') = \max_{\substack{Ay \leq a \\ y \in Y}} (u')^t y' + (u'')^t A'' y''$$

Le dual de la substitution lagrangienne est donné par :

$$(DSL) \begin{cases} z_{DSL} = \min_{u', u''} L(u', u'') \\ u' \in \mathbb{R}^q, u'' \in \mathbb{R}_+^m \end{cases}$$

La linéarisation du problème (DSL) est donnée comme suit :

$$(DSL) \equiv \begin{cases} \min w_1 + w_2 \\ w_1 \geq cx - (u')^t x' - (u'')^t A'' x'' & x \in Z_x \\ w_2 \geq (u')^t y' + (u'')^t A'' y'' & y \in Z_y \\ w_1, w_2 \in \mathbb{R} \\ (u', u'') \in (\mathbb{R}^q, \mathbb{R}_+^m) \end{cases}$$

où $Z_x = \{x \in \mathbb{N}^n | Bx \leq b\}$ et $Z_y = \{y \in Y | Ay \leq a\}$.

Cette méthode a été appliquée pour la résolution du problème de localisation avec capacités dans [23]. Comparée à la décomposition lagrangienne, elle manipule un nombre réduit de variables duales, et permet d'obtenir les mêmes bornes et en moins de temps.

5 Décomposition de Dantzig et Wolfe

Cette méthode [10] a connus une large utilisation pour la décomposition de plusieurs problèmes complexes issus de la pratique, et est devenu un outil standard dans la programmation linéaire [15, 25, 29, 33, 39]. Nous présentons dans cette section son principe, son application à un cas particulier et quelques méthodes de résolution.

Soit $X = \{x \in \mathbb{N}^n : Bx \leq b\}$ un ensemble fini et I l'ensemble des indices de ses éléments. Le problème $(PLNE)$ peut être reformulé comme suit :

$$(PLNE_{DW}) \begin{cases} \max \sum_{i \in I} (c_i x_i) \lambda_i \\ \sum_{i \in I} (A^i x_i) \lambda_i \leq a \\ \lambda_i \in \{0, 1\}, \quad \forall i \in I \end{cases}$$

Notons que ce problème est aussi difficile à résoudre que le problème $(PLNE)$, mais leurs relaxations continues respectives sont différentes. La relaxation continue de $(PLNE)$ implique la relaxation des contraintes d'intégrité sur la globalité du problème, alors que celle de $(PLNE_{DW})$ implique la relaxation des contraintes d'intégrité sur les contraintes $Ax \leq a$ uniquement (alors qu'elles sont maintenues sur les contraintes $Bx \leq b$). La relaxation de $(PLNE_{DW})$ donne donc une meilleure borne que celle de $(PLNE)$.

La décomposition de Dantzig et Wolfe exploite cette propriété, elle repose sur la convexification de l'ensemble X et la relaxation des contraintes d'intégrité sur les variables du problème $(PLNE_{DW})$. Elle se base sur le théorème suivant [10] :

Théorème 1 *Soit $X = \{x \in \mathbb{R}^n | Bx \leq b\}$ un polyèdre borné et non vide. $x \in X$ si et seulement si x peut s'écrire comme une combinaison convexe des points extrêmes de X , i.e.*

$$X = \{x \in \mathbb{R}^n : x = \sum_{j \in J} \lambda_j x_j \mid \sum_{j \in J} \lambda_j = 1, \lambda_j \geq 0, \forall j \in J\} \quad (3)$$

où J est l'ensemble des indices des points extrêmes de X .

Ce résultat peut-être étendu au cas non borné :

Théorème 2 *Soit $X = \{x \in \mathbb{R}^n | Bx \leq b\}$ un polyèdre non vide. $x \in X$ si et seulement si x peut s'écrire comme une combinaison convexe des points extrêmes de X plus une combinaison linéaire non négative des rayons extrêmes de X , i.e.*

$$X = \{x \in \mathbb{R}^n \mid x = \sum_{j \in J} x_j \lambda_j + \sum_{r \in R} x_r \lambda_r \mid \sum_{j \in J} \lambda_j = 1, \lambda_j \geq 0, \forall j \in J \cup R\} \quad (4)$$

où J (resp. R) est l'ensemble des indices des points extrêmes (resp. des rayons extrêmes) de X .

La décomposition de Dantzig et Wolfe consiste à exprimer un polyèdre (convexe) par l'intermédiaire de ses points extrémaux. Remplaçons dans $(PLNE)$ l'expression de chaque variable $x \in X$ par la nouvelle formulation donnée dans le théorème (2), on obtient la formulation suivante :

$$\begin{cases} \max \sum_{j \in J} \lambda_j c_j x_j + \sum_{r \in R} \lambda_r c_r x_r \\ \sum_{j \in J} \lambda_j A^j x_j + \sum_{r \in R} \lambda_r A^r x_r \leq a \\ \sum_{j \in J} \lambda_j = 1 \\ \lambda_j \in \{0, 1\}, \quad \forall j \in J \\ \lambda_r \in \mathbb{N} \end{cases} \quad (5)$$

Posons $\hat{c}_j = c_j x_j$, $\hat{c}_r = c_r x_r$, $\hat{A}^j = A^j x_j$ et $\hat{A}^r = A^r x_r$, il en découle la formulation suivante :

$$\left\{ \begin{array}{l} \max \sum_{j \in J} \hat{c}_j \lambda_j + \sum_{r \in R} \hat{c}_r \lambda_r \\ \sum_{j \in J} \hat{A}^j \lambda_j + \sum_{r \in R} \hat{A}^r \lambda_r \leq a \\ \sum_{j \in J} \lambda_j = 1 \\ \lambda_j \in \{0, 1\}, \quad \forall j \in J \\ \lambda_r \in \mathbb{N} \end{array} \right. \quad (6)$$

Lorsque $R = \emptyset$ (i.e l'ensemble X est borné), le problème ci-dessus se formule comme suit :

$$\left\{ \begin{array}{l} \max z_{PM} = \sum_{j \in J} \hat{c}_j \lambda_j \\ \sum_{j \in J} \hat{A}^j \lambda_j \leq a \\ \sum_{j \in J} \lambda_j = 1 \\ \lambda_j \in \{0, 1\}, \quad \forall j \in J \end{array} \right. \quad (7)$$

Ce nouveau modèle est équivalent à (*PLNE*) et possède $m + 1$ contraintes comparativement à $m + p$ contraintes ; par contre, il contient un nombre beaucoup plus important de variables, celui-ci étant égal au nombre de points extrêmes de X .

5.1 Structure Bloc Diagonale

Dans plusieurs applications, la matrice des contraintes présente une structure de la forme de plusieurs sous-matrices dont les éléments sont non nuls. Les sous-matrices non nulles sont alors regroupées de telle sorte que des sous-ensembles indépendants de variables et de contraintes apparaissent, généralement liés par un ensemble de contraintes, alors qualifiées de liantes ou couplantes.

Supposons que la matrice B du problème (*PLNE*) soit bloc diagonale ; cela induit que le problème (*PLNE*) peut s'écrire de la façon suivante (à une permutation des lignes et des colonnes près) :

$$\left\{ \begin{array}{l} \max cx \\ \left(\begin{array}{cccc} A_1 & A_2 & \cdots & A_p \\ B_1 & 0 & \cdots & 0 \\ 0 & B_2 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & B_p \end{array} \right) x \leq \left(\begin{array}{c} a \\ b_1 \\ b_2 \\ \vdots \\ b_p \end{array} \right) \\ x \in \mathbb{N}^n \end{array} \right.$$

Plus précisément, si chaque sous-matrice B_k contient q_k lignes et n_k colonnes (bien sûr, $\sum_{k=1}^p q_k = q$ et $\sum_{k=1}^p n_k = n$), et en notant x_k (resp b_k , c_k , A_k) le vecteur des variables x (resp. le vecteur du membre droit, le vecteur de la fonction objectif, les

colonnes de la matrice de contraintes), d'indice $\sum_{\ell=1}^{k-1} (n_\ell) + 1$ à $\sum_{\ell=1}^k n_\ell$, (*PLNE*) s'écrit encore :

$$\left\{ \begin{array}{ll} \max \sum_{k=1}^p c_k x_k & \\ \sum_{k=1}^p A_k x_k \leq a & (m \text{ contraintes}) \\ B_k x_k \leq b_k \quad \forall k = 1, \dots, p & (\sum_{k=1}^p q_k = q \text{ contraintes}) \\ x_k \in \mathbb{N}^{n_k} & (\sum_{k=1}^p n_k = n \text{ variables}) \end{array} \right.$$

On peut procéder de la même manière que précédemment (en exprimant directement le problème (*PLNE*) en fonction des points extrémaux du polyèdre $\{x \in \mathbb{N}^n \mid Bx \leq b\}$), mais exploiter la structure particulière du problème semble plus adéquat. Précisément, il peut sembler intéressant d'exprimer le problème (*PLNE*) en fonction des points extrémaux de chacun des domaines $X^k = \{y \in \mathbb{N}^{n_k} \mid B_k y \leq b_k\}$, pour $k = 1, \dots, p$. L'avantage d'une telle démarche est que ces domaines, qui sont dans un espace de dimension plus petite que le domaine global, peuvent être exprimés indépendamment les uns des autres : on espère ainsi réduire la complexité d'expression des points extrémaux.

Pour écrire le problème global en fonction des points extrémaux des sous-systèmes X^k , on introduit encore quelques notations ; on définit pour chaque sous-système $k \in \{1, \dots, p\}$:

J_k : l'ensemble des indices de ses points extrémaux.

n_k : son nombre de variables ($n_k = |J_k|$).

x_k^j , $j \in J_k$: son point extrémal d'indice j .

λ_k^j , $j \in J_k$: la variable d'indice j (coefficient de la combinaison convexe des points extrémaux de X_k associé au point extrême x_k^j).

$\tilde{c}_k^j = \sum_{j \in J_k} c_k x_k^j$: coût associé à la variable λ_k^j .

$\hat{A}_j^k = \sum_{j \in J_k} A_k x_k^j$: colonne associée à la variable λ_k^j .

Toute solution réalisable du système $\{x \in \mathbb{N}^n \mid Bx \leq b\}$ s'écrit :

$$x = \left(\sum_{j \in J_1} \lambda_1^j x_1^j, \sum_{j \in J_2} \lambda_2^j x_2^j, \dots, \sum_{j \in J_p} \lambda_p^j x_p^j \right)$$

où pour k de 1 à p , la famille $(\lambda_k^j)_{j \in J_k}$ vérifie :

$$\sum_{j \in J_k} \lambda_k^j = 1, \quad \lambda_k^j \in \{0, 1\} \quad \forall j \in J_k$$

Le nouveau problème maître (\hat{Q}) construit par analogie avec la formulation de Dant-

zig et Wolfe vue précédemment est alors :

$$(\widehat{Q}) \left\{ \begin{array}{l} \max \sum_{k=1}^p \sum_{j \in J_k} \widehat{c}_k^j \lambda_k^j \\ \sum_{k=1}^p \sum_{j \in J_k} \widehat{A}_k^j \lambda_k^j \leq a \\ \sum_{j \in J_k} \lambda_k^j = 1 \quad \forall k = 1, \dots, p \\ \lambda_k \in \{0, 1\} \quad \forall k = 1, \dots, p \end{array} \right.$$

Le modèle (\widehat{Q}) possède $m+p$ contraintes, comparativement à $m+q$ contraintes pour le problème initial. Son nombre de variables, qui correspond au nombre de points extrémaux des sous-systèmes, est bien entendu toujours d'ordre exponentiel. Dans le cadre d'un schéma de génération de colonnes, on alimentera le problème maître (\widehat{Q}) à l'aide de p sous-problèmes, chacun fournissant des points extrémaux du polyèdre X^k associé. **Notons que cette dernière approche ne devient intéressante que lorsque le nombre de blocs indépendants (i.e. p) est grand.**

5.2 Problème de découpe

Le principe de décomposition de Dantzig et Wolfe a été appliqué indépendamment par Gilmore et Gomory [18] sur le problème de découpe, reformulé en un problème dont la matrice des contraintes est bloc diagonale.

Étant donné des rouleaux (de papier, d'aluminium, ...) de longueur standard L , et m demandes b_i , $i = 1 \dots m$ de pièces de longueur l_i respectivement. L'objectif est de minimiser le nombre total de planches à découper de sorte que la demande soit satisfaite.

La formulation standard est donnée par :

$$\left\{ \begin{array}{ll} \min & \sum_{p=1}^P y_p \\ & \sum_{i=1}^m x_{ip} \geq b_i \quad i = 1 \dots m \\ & \sum_{i=1}^m l_i x_{ip} \leq L y_p \quad p = 1 \dots P \\ & x_{ip} \quad \text{entier} \quad i = 1, \dots, m, p = 1 \dots P \\ & y_p \quad \in \quad \{0, 1\} \quad p = 1 \dots P \end{array} \right. \quad (8)$$

Où :

$y_p=1$ si la planche p est utilisée, $y_p=0$ sinon,

x_{ip} est le nombre de pièces de longueur l_i découpées à partir de la planche p .

P est le nombre total de planches disponibles.

Le premier ensemble de contraintes traduit la satisfaction de la demande, tandis que la deuxième caractérise une découpe réalisable.

Une deuxième modélisation est une décomposition en deux sous-problèmes : le problème *maître* et le *sous problème*. Le premier exprime de façon implicite la satisfaction de la demande ; le second explicite une découpe réalisable.

$$\begin{cases} \min & \sum_{j \in J} \lambda_j \\ & \sum_{j \in J} A^j \lambda_j \geq b \\ & \lambda_j \text{ entier, } j \in J \end{cases} \quad (9)$$

λ_j est le nombre de fois que le plan de découpe j est utilisé.

Dans ce cas, A code l'ensemble des $|J|$ plans de découpe, tel qu'un plan de découpe j est donné par $A^j = (a_1^j, a_2^j, \dots, a_m^j)$. $a_{ij} \in \mathbb{N}$ indique le nombre de planches de longueurs l_i obtenues en appliquant le plan de découpe j .

Soit π le m -vecteur multiplicateur du simplex du problème (9).

Les coûts réduits sont donnés par :

$$\bar{c}_j = 1 - \pi A^j = 1 - \sum_i \pi_i a_i^j$$

Une colonne A^j décrit un schéma réalisable si

$$\sum_{i=1}^m a_i^j l_i \leq L$$

Le sous problème est un problème de minimisation du coût réduit sur l'ensemble de toutes les colonnes correspondants aux schémas de découpes possibles ; On aura ainsi le problème suivant :

$$\begin{cases} \max & \sum_{i=1}^m \pi_i a_i \\ & \sum_{i=1}^m l_i a_i \leq L \\ & a_i \text{ entier, } i = 1, \dots, m \end{cases}$$

(Problème de sac à dos)

Remarque 4 *On peut aboutir à la formulation (9) par décomposition du problème (8) en mettant les contraintes de satisfaction de la demande dans le problème maître, et les contraintes des plans de coupes réalisables dans le sous problème. Une autre manière de faire serait de procéder inversement en mettant les contraintes de coupes réalisables dans le maître et la satisfaction de la demande dans le sous problème.*

5.3 Résolution de la formulation de Dantzig et Wolfe

Elle consiste à résoudre la relaxation continue ($\lambda_j \in [0, 1], \forall j \in J$) du problème (7), appelée le problème maître (PM). La taille des problèmes engendrés par la décomposition de Dantzig et Wolfe est exponentielle (nombre de variables égal au nombre de points extrêmes d'un polyèdre), nous devons donc générer dynamiquement ces variables jusqu'à exhiber la base optimale. Nous présentons dans ce qui suit deux méthodes basées sur ce principe et dont chacune est vue comme étant le dual de l'autre.

5.3.1 Génération de colonnes

Cette méthode a connu un énorme essor lors des dernières décennies pour la résolution de problèmes avec un nombre important de variables, notamment pour la résolution de formulations de Dantzig et Wolfe, elle a été appliquée efficacement à des problèmes variés. Une classification des applications de la génération de colonnes a été présentée dans [27].

Le principe de cette méthode est de ne pas considérer explicitement toutes les variables du problème (et par le même temps, toutes les colonnes associées de la matrice des contraintes), mais uniquement un ensemble réduit manipulant ainsi un problème maître restreint (PMR). Des variables sont ajoutées à (PMR) au fur et à mesure des itérations, jusqu'à atteindre l'optimalité (c'est-à-dire, jusqu'à ce que le problème maître restreint contienne toutes les colonnes d'une solution optimale de (PM)).

La méthode de génération de colonnes tient son nom du fait que l'on ne connaisse pas a priori les solutions (points extrémaux) du (des) sous-problème(s). Si l'on ne connaît pas ces solutions, alors on ne connaît pas leur composition et de fait, la colonne \widehat{A}^j qui leur est associée (de même d'ailleurs que leur coefficient de coût \widetilde{c}^j). Ainsi, l'introduction d'une variable λ^j dans le problème maître restreint induit l'introduction de la colonne \widehat{A}^j qui lui est associée dans la matrice des contraintes.

Considérons à présent le problème maître restreint (PMR) :

$$(PMR) \left\{ \begin{array}{l} \max_{\lambda} \sum_{j \in J'} \widetilde{c}^j \lambda^j \\ \sum_{j \in J'} \widehat{A}^j \lambda^j \leq a \\ \sum_{j \in J'} \lambda^j = 1 \\ \lambda^j \geq 0, \quad \forall j \in J' \end{array} \right.$$

où J' est un sous-ensemble strict de J . La matrice des contraintes est alors réduite aux colonnes correspondant aux points extrémaux d'indices à valeurs dans J' .

Considérons le programme linéaire (PM) et le programme restreint (PMR). Lors de chaque itération de l'algorithme du simplexe [9], en vue de l'amélioration de l'objectif de (PM), on cherche la variable hors base candidate à entrer dans la base. Dans notre cas (problème de maximisation), on cherchera une variable hors base de coût réduit positif (éventuellement, maximal).

Or, le coût réduit \bar{c}^j d'une variable hors base λ^j dans le problème (PM) s'écrit :

$$\bar{c}^j = \widetilde{c}^j - u^t \widehat{A}^j - u_0$$

où u est le multiplicateur correspondant aux contraintes couplantes ($\sum_{j \in J} \widehat{A}^j \lambda^j \leq a$) et u_0 le multiplicateur correspondant à la contrainte de convexité ($\sum_{j \in J} \lambda^j = 1$). Le problème de la recherche de la variable hors base de coût réduit maximal revient ainsi à résoudre le problème suivant appelé aussi sous-problème :

$$\bar{c}^{j*} = \max\{\hat{c}^j - u^t \hat{A}^j - u_0 \mid j \in J \setminus J'\}$$

équivalent à résoudre le problème suivant :

$$\max\{\hat{c}^j - u^t \hat{A}^j \mid j \in J\} \quad (10)$$

Si $\bar{c}^{j*} \leq 0$ (les coûts réduits de toutes les variables hors base sont négatifs ou nuls), alors la solution courante est optimale pour (PM) ; sinon, λ^{j*} entre en base et le problème maître restreint augmenté d'une colonne est réoptimisé.

Cette procédure s'arrête lorsque la base optimale (resp. ϵ -optimale) est obtenue ($\bar{c}^{j*} \geq 0$ (resp. $\bar{c}^{j*} \geq -\epsilon$)).

5.3.2 Méthode des plans coupants de Kelley

Cette technique peut être considérée comme la version duale de la génération de colonnes où la formulation décomposée engendre un nombre exponentiel de contraintes. Cette formulation est obtenue en considérant le dual du problème (PM) donné ci-dessous :

$$\begin{aligned} (DM) &\equiv \begin{cases} \min & ua + u_0 \\ & u_0 \geq \hat{c}_j - u^t \hat{A}^j \quad j \in J \\ & (u, u_0) \in (\mathbb{R}_+^m, \mathbb{R}) \end{cases} \\ &\equiv \begin{cases} \min & ua + u_0 \\ & u_0 \geq c_j x_j - u^t (A^j x_j) \quad j \in J \\ & (u, u_0) \in (\mathbb{R}_+^m, \mathbb{R}) \end{cases} \\ &\equiv \begin{cases} \min & \theta \\ & \theta + u^t (Ax_j - b) \geq cx_j, \quad j \in J \\ & u \in \mathbb{R}_+^m, \theta \in \mathbb{R} \end{cases} \end{aligned}$$

Cette formulation est équivalente à la formulation vue dans la section 2.2 et est résolue de façon équivalente.

5.4 Interprétation économique de la décomposition de Dantzig et Wolfe

Nous allons présenter dans cette section quelques propriétés de la décomposition de Dantzig et Wolfe [25].

Soient \mathcal{B} une base réalisable du problème (PM) , $x_{\mathcal{B}} = \mathcal{B}^{-1}a$ une solution de base réalisable et $u = c_{\mathcal{B}}\mathcal{B}^{-1}$ la solution duale correspondante. On s'intéresse à l'influence d'une perturbation du vecteur a sur l'objectif z_{PM} .

Supposons que la nouvelle valeur de a est $a' = a + \Delta a$. On remarque que $u = c_{\mathcal{B}}\mathcal{B}^{-1}$ ne dépend pas de a et reste donc toujours une solution duale réalisable. Dans ces conditions, si le nouveau vecteur a' vérifie $\mathcal{B}^{-1}a' \geq 0$, la base \mathcal{B} reste à la fois primale et duale

réalisable.

Pour $\|\Delta a\|$ suffisamment petit, nous avons :

$$ua' = ua + u \Delta a$$

Sachant que $z_{PM} = ua$, on en déduit :

$$\frac{\partial z_{PM}}{\partial a_i} = u_i$$

Notons que dans la formulation (PM) , $a_i, i = 1, \dots, m$ sont des ressources partagées entre les sous-systèmes, le multiplicateur u_i est appelé le coût marginal de la ressource i car il représente le gain en objectif obtenu par une augmentation de la ressource i d'une unité.

Revenons au principe de décomposition, considérons l'action du $k^{\text{ème}}$ sous-système et son effet sur l'objectif z_{PM} . Si le sous-système k choisit un vecteur d'activité x^k , cela engendrera un coût direct $c^k x^k$ et une consommation de $A^{(k)} x^k$ unités de ressources, ce qui pénalisera la consommation de ces ressources par les autres sous-systèmes (ressources rares).

Afin de bien gérer ce phénomène, on attribue des coûts marginaux u aux ressources et on force les sous-systèmes à payer lorsque des quantités de la ressource sont utilisées. Si on attribue à une ressource un coût marginal élevé, cela va décourager les sous-systèmes à utiliser de grandes quantités de cette ressource.

Il est facile de voir que le principe de décomposition coordonne les actions des sous-systèmes précisément dans ce sens. La fonction objectif du $k^{\text{ème}}$ sous-problème est :

$$z^k = c^k x^k - u A^{(k)} x^k$$

Le terme $c^k x^k$ est la contribution directe de la solution x^k au profit total z_{PM} engendré par le vecteur x^k , alors que $u A^{(k)} x^k$ est le coût total à payer par le sous-système k pour l'utilisation de $A^{(k)} x^k$ unités des ressources (l'augmentation de la consommation d'une ressource par un sous-système engendre l'augmentation du prix à payer). À cet effet, l'expression $u A^{(k)} x^k$ est précédée par un signe négatif dans la formulation de z^k .

Une autre interprétation économique de cette méthode peut être dégagée du critère d'arrêt. La résolution de la formulation de Dantzig et Wolfe s'arrête lorsque le coût réduit maximal \bar{c}^k de chaque sous-système k est négatif ou nul, cela se traduit comme suit :

$$\bar{c}^k = z^{k,*} - u_0^k \leq 0 \iff z^{k,*} \leq u_0^k$$

où $z^{k,*}$ (resp. u_0^k) est la valeur optimale (resp. le multiplicateur correspondant à la contrainte de convexité) du $k^{\text{ème}}$ sous-problème.

Notons que u_0^k est le changement de la valeur de z_{PM} engendré par un changement du second membre de la contrainte de convexité. Augmenter cette valeur d'une unité signifie que la somme des poids attribués aux solutions du sous-système est supérieure à 1. Nous pouvons conclure que u_0^k est le coût marginal de la répartition des contributions des colonnes générées dans la solution. Le test d'optimalité ci-dessus est équivalent à vérifier si le coût de la répartition courante est supérieur à celui obtenu par un changement de répartition, et dans ce cas, la répartition courante est optimale.

6 Décomposition de Benders

Cette méthode est utilisée pour la résolution des programmes linéaires à **variables mixtes** (variables entières et variables continues par exemple). Elle se base sur le principe suivant : lorsqu'au cours de la résolution, on déduit qu'une solution partielle du problème ne peut pas être complétée pour former une solution réalisable, toutes les solutions engendrées par ces solutions partielles sont ignorées.

La stratégie de Benders [4] consiste à partitionner les variables du problème en deux vecteurs x et y , elle fixe le vecteur y de telle sorte à générer un problème uniquement avec la variable x que l'on appelle le sous-problème ; si la solution optimale de ce problème révèle que la solution y est non réalisable, alors la solution duale correspondante est utilisée afin d'identifier d'autres valeurs inacceptables de y .

Soit le programme linéaire suivant à n variables et m contraintes :

$$(PB) \left\{ \begin{array}{l} \max cx + sy \\ Ax + By \leq b \\ y \in Y \\ x \in \mathbb{R}_+^{n_1} \end{array} \right.$$

$c \in \mathbb{R}^{n_1}$, $s \in \mathbb{R}^{n_2}$, $A \in \mathbb{R}^{m \times n_1}$, $B \in \mathbb{R}^{m \times n_2}$, $n = n_1 + n_2$, $b \in \mathbb{R}^m$, $n_1, n_2, n, m \in \mathbb{N}$ et Y un ensemble quelconque de vecteurs de dimension n_2 ($Y \subset \mathbb{N}^{n_2}$ par exemple, au quel cas (PB) est un problème en variables mixtes).

En fixant le vecteur y dans (PB) , on obtient le programme linéaire $(PA(y))$, soit $(DA(y))$ son dual

$$(PA(y)) \left\{ \begin{array}{l} \max cx \\ Ax \leq b - By \\ x \in \mathbb{R}_+^{n_1} \end{array} \right. \quad (DA(y)) \left\{ \begin{array}{l} \min (b - By)^t u \\ uA \geq c \\ u \in \mathbb{R}_+^m \end{array} \right.$$

Le problème d'optimisation initial admet plusieurs écritures équivalentes :

$$\begin{aligned} (PB) &\equiv \max\{sy + cx \mid Ax + By \leq b, y \in Y, x \in \mathbb{R}_+^{n_1}\} \\ &\equiv \max\{sy + \max\{cx \mid Ax \leq b - By, x \in \mathbb{R}_+^{n_1}\} \mid y \in Y\} \\ &\equiv \max\{sy + \min\{(b - By)^t u \mid uA \geq c, u \in \mathbb{R}_+^m\} \mid y \in Y\} \end{aligned}$$

En linéarisant la dernière formulation, on obtient le problème suivant, appelé problème maître de Benders :

$$(PMB) \begin{cases} \max w \\ w \leq sy + (b - By)^t u_y \\ y \in Y \\ w \in \mathbb{R} \end{cases}$$

où u_y désigne pour tout y une solution optimale de $(DA(y))$.

Le principe de résolution consiste alors à alimenter le problème (PMB) par des solutions u_y qui sont optimales pour le sous-problème dual $(DA(y))$, en fixant le vecteur y à différentes valeurs. Cet ajout se traduit par des coupes qui rétrécissent le domaine d'optimisation du problème maître.

En effet, la difficulté est d'énoncer les contraintes de type

$$w \leq sy + (b - By)^t u_y \quad (11)$$

où u_y est défini implicitement comme solution optimale du sous-problème dual. Pour éviter d'avoir à expliciter toutes ces contraintes (ce qui reviendrait à parcourir tout Y), on part d'un problème maître de Benders non contraint (problème maître restreint de Benders, noté $(PMRB)$) dans lequel on insère des contraintes (ou coupes) au fur et à mesure du déroulement de l'algorithme. La génération de ces coupes est guidée par la résolution optimale successive d'un problème dual $(DA(y))$ pour un y fixé et du problème $(PMRB)$ enrichi d'une nouvelle coupe.

Remarque 5

(a) *Expression du problème maître restreint de Benders*

À l'initialisation, le problème maître restreint s'écrit :

$$(PMRB) \begin{cases} \min w \\ y \in Y \\ w \in \mathbb{R} \end{cases}$$

À la fin de l'itération k , il s'écrit :

$$(PMRB) \begin{cases} \min w \\ w \leq sy + (b - By)^t u^\ell, \quad \ell = 1, \dots, k \\ y \in Y \\ w \in \mathbb{R} \end{cases}$$

(b) *Validité de la coupe*

On note $F(DA) = \{u \geq 0 \mid u^t A \geq c\}$ l'ensemble des solutions réalisables des sous-problèmes duaux. Par définition de $(DA(y))$, on a :

$$\forall (y, u) \in Y \times F(DA), \quad (b - By)^t u_y \leq (b - By)^t u$$

Ainsi, pour tout couple $(w, y) \in \mathbb{R} \times Y$:

$$w \leq sy + (b - By)^t u_y \Rightarrow w \leq sy + (b - By)^t u, \forall u \in F(DA)$$

Les contraintes “ $w \leq sy + (b - By)^t u_y$ ” sont donc en particulier vérifiées par toute solution réalisable du problème maître de Benders (PMB).

(c) Reprenons la formulation de Benders du problème (PB) :

$$\max\{sy + \min\{(b - By)^t u \mid u^t A \geq c, u \geq 0\} \mid y \in Y\}$$

Notons $U = \{u \in \mathbb{R}_+^m \mid u^t A \geq c\}$ supposé convexe et borné et I l'ensemble des indices de ses points extrêmes, le problème ci-dessus est équivalent à

$$\max_{y \in Y} \{sy + \min_{i \in I} \{(b - By)^t u_i\}\}$$

Une nouvelle formulation de (PMB) sera :

$$(PMB) \begin{cases} \max w \\ w \leq sy + (b - By)^t u_i, \quad i \in I \\ y \in Y \\ w \in \mathbb{R} \end{cases}$$

La décomposition de Benders a été appliquée avec succès sur plusieurs applications, on cite [13] pour la résolution du problème de programmation des mouvements des moteurs ferroviaires, [38] pour le routage aérien, [17] pour la construction de systèmes de distribution industriels et [12, 6, 7] pour la résolution de problèmes de tournées de véhicules. Cependant, cette méthode n'est pas adaptée à certaines applications [41], à cet effet, plusieurs méthodes ont été proposées pour son amélioration [41, 30, 5, 16].

7 Décomposition par adaptation du second membre des contraintes

Contrairement aux méthodes présentées précédemment qui exploitent les coûts marginaux des ressources partagées pour trouver une répartition optimale, cette technique [25] consiste à les manipuler directement.

Considérons la généralisation du problème (PLNE) où la matrice des contraintes B est bloc diagonale (section 5.1) reformulée comme suit :

$$\min \sum_{i=1}^p c^i x^i \quad \text{s.c} \quad \begin{cases} \sum_i A^{(i)} x^i \leq a \\ x^i \in X^i, \quad i = 1, \dots, p \end{cases} \quad (12)$$

Où x^i est un vecteur de dimension n_i , $c^i \in \mathbb{R}^{n_i}$, $A^{(i)} \in \mathbb{R}^{(m, n_i)}$, X^i le domaine réalisable du bloc de contraintes $B_i x^i \leq b_i$ supposé fini et non vide, pour tout $i = 1, \dots, p$,

$\sum_{i=1}^p n_i = n$ et $a \in \mathbb{R}^m$.

Ce problème peut être vu comme un problème de partitionnement des ressources a sur les p sous-systèmes. Sa résolution consiste à trouver un partitionnement réalisable, à tester s'il est optimal et à l'améliorer si possible.

Notons $y^i \in \mathbb{R}^m$ une partition de a pour le $i^{\text{ème}}$ sous-système, la condition de réalisabilité d'un partitionnement est donnée par :

$$\sum_{i=1}^p y^i \leq a \quad (13)$$

Étant donnée la partition y^i , le $i^{\text{ème}}$ sous-système peut l'utiliser à volonté, elle vérifie donc le sous-problème suivant :

$$(SP_i(y^i)) \quad \min c^i x^i \quad \text{s.c.} \quad \begin{array}{l} A^{(i)} x^i \leq y^i \\ x^i \in X^i \end{array}$$

L'ensemble des partitions réalisables est donné par :

$$y^i \in V_i = \{y^i \mid \exists x^i \in X^i \text{ tel que } A^{(i)} x^i \leq y^i\} \quad (14)$$

Afin de résoudre le problème (12), nous devons choisir des vecteurs $y^i, i = 1, \dots, p$ qui minimisent l'objectif et vérifient les contraintes (13) et (14). Le problème (12) est donc équivalent au problème ci-dessous que l'on appelle le problème maître :

$$(PMA) \quad \min W(y) = \sum_{i=1}^p W_i(y^i) \quad \text{s.c.} \quad \begin{array}{l} \sum_{i \in I} y^i \leq a \\ y^i \in V_i, \quad i = 1, \dots, p \end{array} \quad (15)$$

où $W_i(y^i)$ est la valeur optimale du $i^{\text{ème}}$ sous-problème.

L'objectif et les contraintes $y^i \in V_i, i = 1, \dots, p$ sont non linéaires, alors que les contraintes $\sum_{i \in I} y^i \leq a$ sont linéaires. Cependant, l'évaluation de la fonction W en un point y nécessite la résolution des sous-problèmes $(SP_i(y^i)), i \in I$, qui sont plus simples à résoudre que le problème d'origine et peuvent être résolus indépendamment.

Notons que si $\bar{y} = (\bar{y}^1, \dots, \bar{y}^p)$ résout (PMA) et si \bar{x}^i résout $(SP_i(y^i)), \forall i = 1, \dots, p$ alors, $\bar{x} = (\bar{x}^1, \dots, \bar{x}^p)$ résout le problème (12). Inversement, si \bar{x} résout le problème (12), alors $(A^{(1)} \bar{x}^1, \dots, A^{(p)} \bar{x}^p)$ résout (15).

Afin d'exploiter certains résultats théoriques pour la résolution du problème maître, nous énonçons les hypothèses suivantes :

1. X_i est non vide, compact et convexe.
2. Les fonctions $f^i(x^i) = c^i x^i$ et $g^i(x^i) = A^i x^i$ sont convexes et différentiables sur X_i .

3. Le problème primal possède une solution réalisable.

La convexité des fonctions f^i , g^i et X_i assure que V^i est convexe, donc le problème maître est convexe. La réalisabilité du primal implique la réalisabilité du problème maître. La compacité des ensembles X_i et la continuité des fonctions f et g assurent que le problème possède une solution optimale lorsque le problème est réalisable. La différentiabilité de toutes les fonctions permettra de formuler l'expression de la dérivée directionnelle de W^i en utilisant la programmation linéaire.

Malgré toutes ces hypothèses, W^i peut ne pas être différentiable sur tout l'ensemble V^i , ce qui engendre la plus grande difficulté pour résoudre le problème maître. Des méthodes de résolution qui se basent sur la recherche d'une direction de descente ont été présentées dans [25].

8 Hybridation des méthodes de décomposition

Une classe de méthodes de décomposition appelées méthodes de décomposition intégrées ont été présentées dans [36]. Elles sont issues de l'hybridation de méthodes de décomposition interne et externe qui offrent de meilleures bornes en moins de temps que celles obtenues par chacune des méthodes utilisées indépendamment l'une de l'autre.

Lorsque les méthodes de décomposition classiques construisent une approximation, soit interne, soit externe, les méthodes de décomposition intégrées offrent les deux approximations à la fois. Nous présentons dans ce qui suit deux techniques qui intègrent ce principe.

8.1 Price and Cut

Cette méthode consiste à intégrer la méthode des plans coupants dans la décomposition de Dantzig et Wolfe. Lorsqu'elle est intégrée dans un processus de branch-and-bound, elle est appelée le *branch-and-price-and-cut* [2, 3, 14, 40].

Cette technique alterne entre la résolution du sous-problème d'évaluation qui permet de générer des colonnes améliorantes et du sous-problème de séparation qui génère des inégalités valides améliorantes. La résolution de chacun de ces problèmes produit respectivement une description interne et une description externe du domaine réalisable (\mathcal{P}).

Le principe est d'optimiser l'objectif sur l'intersection des deux descriptions générées dynamiquement. Pour cela, on utilise une formulation de Dantzig et Wolfe dont le système de contraintes contient en plus des inégalités valides de la forme $\sum_{j \in J} (Dx_j)\lambda_j \leq d$. Ces contraintes changent dynamiquement, soit en nombre de colonnes (ajout de colonnes de coût réduit positif), soit en nombre de contraintes (ajout d'inégalités valides).

Lors de chaque itération du *price-and-cut*, on résout au choix le sous-problème d'évaluation ou le sous-problème de séparation. La procédure s'arrête lorsqu'aucune colonne de cout réduit positif, ni aucune inégalité valide, ne peut être générée.

Remarque 6 *La génération de nouvelles colonnes conduit à une augmentation de la valeur du minorant, alors que la génération de nouvelles inégalités valides conduit à la diminution de la valeur du majorant.*

8.2 Relax and Cut

Comme pour la méthode de Dantzig et Wolfe, la méthode des plans coupants peut être intégrée dans la relaxation lagrangienne [8, 11, 21, 28]. Le principe est le même que celui ci-dessus. Lors de chaque itération de cette méthode, un sous-problème d'évaluation ou un sous-problème de séparation est résolu. La mise à jour des multiplicateurs duaux se fait non pas en résolvant un problème maître, mais en utilisant un algorithme de sous-gradient.

La résolution du sous-problème d'évaluation implique la génération d'un point extrême du polyèdre défini par $\text{conv}(X)$. Cette solution permet la mise à jour de la solution duale, alors que la résolution du problème de séparation fournit une inégalité améliorante qui est rajoutée au système de contraintes de la relaxation lagrangienne du problème traité.

9 Synthèse

Nous présentons dans cette section, les relations existantes entre quelques techniques présentées précédemment.

9.1 Relation entre les méthodes lagrangiennes

Contrairement à la relaxation lagrangienne dont les contraintes relâchées n'apparaissent plus dans le système des contraintes, la relaxation agrégée permet de conserver les contraintes relâchées dans un sous-problème. On conserve donc plus d'informations avec la décomposition lagrangienne et on obtient ainsi une meilleure borne que celle de la relaxation lagrangienne. D'un autre côté, la substitution lagrangienne fournit une meilleure borne que celle de la relaxation lagrangienne ; dans un contexte de maximisation, cela se traduit par le résultat suivant [32, 24] :

$$z_{PL} \geq z_{RL} \geq z_{DSL} \geq z_{DL} \geq z_{PLNE}$$

La décomposition lagrangienne permet d'avoir une borne de meilleure qualité, mais sa mise en oeuvre nécessite plus d'effort de calcul, car on résout deux sous-problèmes lors de chaque itération contre un pour la relaxation lagrangienne ; d'un autre côté, le dual lagrangien est de taille p alors que le dual de la décomposition lagrangienne est de

taille $m + p$.

Pour remédier à cela, la substitution lagrangienne se positionne comme un compromis entre la relaxation et la décomposition lagrangienne, la taille du problème dual est alors réduit à une taille comprise entre m et n . Remarquons que ce que l'on gagne en qualité de la borne, on le perd en temps de résolution.

9.2 Relation entre la relaxation lagrangienne et la décomposition de Dantzig et Wolfe

Il est bien connu que lorsque la relaxation lagrangienne est obtenue en dualisant exactement les mêmes contraintes liantes dans la formulation de Dantzig et Wolfe, la valeur optimale du dual lagrangien et de la formulation de Dantzig et Wolfe (relâchée) sont égales. Nous montrons ci-dessous l'équivalence entre les deux formulations.

Considérons le problème (*PLNE*) et le dual lagrangien (*DL*) reformulé comme suit :

$$(DL) \equiv \begin{cases} z_{DL} = \min_{u \in \mathbb{R}_+^m} (u^t a + \max(c - u^t A)x) \\ Bx \leq b \\ x \in \mathbb{N}^n \end{cases} \\ \equiv \min_{u \in \mathbb{R}_+^m} \max_{j \in J} u^t a + (c - u^t A)x_j$$

où J est l'ensemble des points extrêmes de $X = \{x \in \mathbb{N}^n | Bx \leq b\}$. Posons $w = \max_{j \in J} u^t a + (c - u^t A)x_j$. On peut écrire (*DL*) sous la forme linéaire suivante :

$$\begin{cases} \min w \\ w \geq u^t a + (c - u^t A)x_j, & j \in J \\ (w, u) \in \mathbb{R} \times \mathbb{R}_+^m \end{cases} \quad (16)$$

Le dual de (16) s'écrit :

$$\begin{cases} \min \sum_{j \in J} (cx_j) \lambda_j \\ \sum_{j \in J} (Ax_j - a) \lambda_j \leq 0 \\ \sum_{j \in J} \lambda_j = 1 \\ \lambda_j \geq 0, & j \in J \end{cases} \quad (17)$$

Nous avons :

$$\sum_{j \in J} (Ax_j - a) \lambda_j = \sum_{j \in J} (Ax_j) \lambda_j - \sum_{j \in J} a \lambda_j$$

et

$$\sum_{j \in J} a \lambda_j = a$$

On aboutit alors au problème (18) équivalent au problème (*PM*) obtenu par la décomposition de Dantzig et Wolfe :

$$\left\{ \begin{array}{l} \min \sum_{j \in J} (cx_j) \lambda_j \\ \sum_{j \in J} (Ax_j) \lambda_j \leq a \\ \sum_{j \in J} \lambda_j = 1 \\ \lambda_j \geq 0, \quad j \in J \end{array} \right. \quad (18)$$

9.3 Relation entre la décomposition de Benders et la méthode de Dantzig et Wolfe

Montrons que ces deux méthodes peuvent être vues comme étant duales l'une par rapport à l'autre.

Considérons la relaxation continue du problème (PB) vu dans la section 6, soit (DB) son dual

$$(PB) \left\{ \begin{array}{l} \max cx + sy \\ Ax + By \leq b \\ x, y \in (\mathbb{R}^{n_1}, \mathbb{R}^{n_2}) \end{array} \right. \quad (DB) \left\{ \begin{array}{l} \max b^t u \\ Au \geq c \\ Bu \geq s \\ u \geq 0 \end{array} \right.$$

Notons $U = \{u \in \mathbb{R}_+^m \mid u^t A \geq c\}$ supposé convexe et borné. Appliquons la décomposition de Dantzig et Wolfe sur le problème (DB) en substituant ses variables par une combinaison convexe des points extrêmes de U indicés dans I . Chaque point $u \in U$ s'écrit donc comme suit :

$$u = \sum_{i \in I} \lambda_i u_i, \quad \sum_{i \in I} \lambda_i = 1, \quad \lambda_i \geq 0, \quad \forall i \in I$$

Il en découle la formulation (PBD) (le problème maître de Dantzig et Wolfe), soit $(PBDD)$ son dual :

$$(PBD) \left\{ \begin{array}{l} \max \sum_{i \in I} (b^t u_i) \lambda_i \\ \sum_{i \in I} (B^t u_i) \lambda_i \geq s \\ \sum_{i \in I} \lambda_i = 1 \\ \lambda_i \geq 0, \forall i \in I \end{array} \right. \quad (PBDD) \left\{ \begin{array}{l} \max sy + v \\ v + (By)^t u_i \leq b^t u_i, \quad \forall i \in I \\ y \in \mathbb{R}^{n_2} \\ v \in \mathbb{R} \end{array} \right.$$

Posons $w = sy + v$, en substituant cette expression dans $(PBDD)$, on aura :

$$\left\{ \begin{array}{l} \max w \\ w + (By - b)^t u_i \leq sy, \quad \forall i \in I \\ y \in \mathbb{R}_+^{n_2} \\ w \in \mathbb{R} \end{array} \right.$$

Le dual du problème maître (PBD) est équivalent au problème de Benders vu dans la remarque (5-(c)).

Soient (y^0, v^0) les solutions duales optimales du problème (*PBD*). Afin de vérifier si la solution courante est optimale, on calcule le coût réduit de chaque variable λ_i donné par

$$\bar{c}_i = b^t u_i - (y^0)^t B^t u_i - v^0$$

Sachant que u_i est un point extrême de U , maximiser \bar{c}_i , $i \in I$ revient à résoudre le problème

$$\begin{cases} \max (b - B y^0)^t u \\ A^t u \geq c \\ u \in \mathbb{R}_+^m \end{cases}$$

qui est le sous-problème de Benders.

Nous avons montré ci-dessus que la résolution d'un problème avec la décomposition de Benders revient à la résolution de son dual avec la décomposition de Dantzig et Wolfe, ces méthodes sont donc vues comme duales l'une par rapport à l'autre.

Conclusion

Nous avons présenté dans ce rapport quelques méthodes de décomposition, dont l'utilisation est de plus en plus répandue dans les applications pratiques. Elles consistent à approcher la valeur optimale d'un programme linéaire en nombres entiers (ou mixtes) par approximation de son domaine réalisable. Ces méthodes engendrent généralement des programmes linéaires de grande taille (nombre important de variables et/ou de contraintes).

Quelques-unes de ces techniques consistent à relâcher les contraintes d'intégrité sur un sous-ensemble de contraintes (cela a pour effet de diminuer la complexité du problème en manipulant deux ou plusieurs problèmes plus faciles à résoudre que le problème d'origine : un problème en nombres entiers et un problème en continu) ou encore éclater le vecteur des variables en plusieurs vecteurs pour manipuler des sous-problèmes avec un nombre réduit de variables.

Les méthodes de décomposition sont souvent indispensables pour la résolution de certains problèmes complexes ; elles permettent de calculer une solution approchée de la valeur optimale, ce qui peut conduire à accélérer la procédure de *branch-and-bound*.

Références

- [1] K. Aardal and S. Van Hoesel. Polyhedral techniques in combinatorial optimization. *Statistica Neerlandica*, 50 :3–26, 1996.

- [2] C. Barnhart, C.A. Hane, and P.H. Vance. Using branch-and-price-and-cut to solve origin-destination integer multicommodity flow problems. Operations Research, 48(2) :318–326, 2000.
- [3] G. Belov and G. Scheithauer. A branch-and-cut-and-price algorithm for one-dimensional stock cutting and two-dimensional two-stage cutting. European journal of operational research, 171(1) :85–106, 2006.
- [4] J.F. Benders. Partitioning procedures for solving mixed-variables programming problems. Numerische Mathematik, 4 :238–252, 1962.
- [5] S. Binato, M.V.F. Pereira, and S. Granville. A new benders decomposition approach to solve power transmission network design problems. Numerische Mathematik, 16(2) :235–240, 2001.
- [6] J.F. Cordeau, F. Soumis, and J. Desrosiers. A benders decomposition approach for the locomotive and car assignment problem. Transportation Science, 34(2) :133–149, 1975.
- [7] J.F. Cordeau, G. Stojkovic, F. Soumis, and J. Desrosiers. Benders decomposition for simultaneous aircraft routing and crew scheduling. Transportation Science, 35(4) :375–388, 1975.
- [8] A. Cunha, A. Lucena, N. Maculan, and M. Resende. A relax and cut algorithm for the prize collecting steiner problem in graphs. Proceedings of Mathematical Programming in Rio, pages 72–78, 2003.
- [9] G.B. Dantzig. Linear Programming and Extensions. Princeton University Press, Princeton, NJ, USA, 1963.
- [10] G.B. Dantzig and P. Wolfe. Decomposition principle for linear programs. Operations Research, 8 :101–111, 1960.
- [11] L.F. Escudero, M. Guignard, and K. Malik. A lagrangian relax-and-cut approach for the sequential ordering problem with precedence. Annals of Operations Research, 50(1) :219–237, 2005.
- [12] M.L. Fisher and R. Jaicumeur. A decomposition algorithm for large scale vehicle routing. Technical Report 78-11-05, Department of decision sciences, university of Pennsylvania, 1978.
- [13] M.G. Florian, G. Guerin, and G. Bushel. The engine scheduling problem in a railway network. INFOR J, 14 :121–138, 1976.
- [14] R. Fukasawa¹, H. Longo, J. Lysgaard, M.P. Aragao, M. Reis, E. Uchoa, and R.F. Werneck. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. Mathematical Programming, 106(3) :491–511, 2006.
- [15] A.M. Geoffrion. Lagrangean relaxation for integer programming. Mathematical Programming Stud., 2 :82–114, 1974.
- [16] A.M. Geoffrion. Generalized benders decomposition. Journal of Optimization Theory and Applications, 10(4) :237–260, 2005.

- [17] A.M. Geoffrion and G. Graves. Multicommodity distribution system design by benders decomposition. Mgmt. Sci., 5 :822–844, 1974.
- [18] P.C. Gilmore and R.E. Gomory. A linear programming approach to the cutting stock problem. Operations Research, 9 :849–859, 1961.
- [19] M. Guignard. An application of lagrangean decomposition to the resource-constrained minimum weighted arborescence problem. Networks, 20(3) :345–359, 1990.
- [20] M. Guignard. Lagrangean decomposition and lagrangean substitution for stochastic integer programming. Networks, 20(3) :345–359, 1990.
- [21] M. Guignard. Efficient cuts in lagrangean relax-and-cut schemes. European Journal of Operational Research, 105(1) :216–223, 1998.
- [22] M. Guignard. Lagrangean relaxation. In M. Resende and P. Pardalos, Oxford university press, 2004.
- [23] M. Guignard and B. Chen. Polyhedral analysis and decompositions for capacitated plant location-type problems. Discrete Applied Mathematics, 82 :79–91, 1989.
- [24] M. Guignard and S. Kim. Lagrangian decomposition : a model yielding stronger lagrangian bounds. Mathematical programming, 39 :215–228, 1987.
- [25] L.S. Lasdon. Optimization theory for large systems. Macmillan series in operations research, 1972.
- [26] E.L. Lawer and D.E. Wood. Branch-and-bound : A survey. Operations research, 14 :699–719, 1966.
- [27] M E. Lubecke and J. Desrosiers. Selected topics in column generation. Technical Report G-2002-64, Les cahiers du GERAD, 2002.
- [28] A. Lucena. Non-delayed relax-and-cut algorithms. Technical report, Working paper, Departmaneto de administracao, Universidade Federal do Rio de Janeiro, 2004.
- [29] T.L. Magnanti, J.F. Shapiro, and MH. Wagner R.T. Wong. Generalized linear programming solves the dual. Management science, 22 :1195–1203, 1976.
- [30] T.L. Magnanti and R.T. Wong. Accelerating benders decomposition : Algorithmic enhancement and medel selection criteria. Operations research, 29(3) :464–484, 1981.
- [31] P. Michelon. Méthodes lagrangiennes pour la programmation linéaire avec variables entières. Investigación Operativa, 2(2) :127–146, 1991.
- [32] P. Michelon and N. Maculan. Lagrangean decomposition for integer nonlinear programming with linear constraints. Mathematical Programming, 52(1-3) :303–313, 2005.
- [33] G.L. Nemhauser and L.A. Wolsey. Integer and combinatorial optimization. Wiley, New York, 1988.
- [34] B.T. Polyak. A general method for solving extremum problems. Soviet. Math. Dokl., 8 :593–597, 1967.

- [35] B.T. Polyak. Minimization of unsmooth functionals. USSR Comput. Math. and Math. Phys., 9 :509–521, 1969.
- [36] T.K. Ralphs and M.V. Galati. Decomposition in integer linear programming. Technical Report 04T-019, Lehigh University Industrial and Systems Engineering Department, 2005.
- [37] H. Reinoso and N. Maculan. Lagrangean decomposition for integer programming : a new scheme. INFOR, 52(2) :1–5, 1992.
- [38] R Richardson. An optimization approach to routing aircraft. Trans. Sci, 10 :52–71, 1976.
- [39] J.F. Shapiro. Mathematical programming, structures and algorithms. John Willey and Sons, Inc, 1979.
- [40] F. Vanderbeck. Exact algorithm for minimising the number of setups in the one-dimensional cutting stock problem. Operation Research, 48(6) :915–926, 2000.
- [41] R. Wong. Accelerating Benders decomposition for network design. Ph.D.thesis, Department of electrical engineering and computer science, Massachusetts Institute of technology, 1978.