



HAL
open science

Sur l'accélération de la convergence de la génération de colonnes

Nora Touati, Lucas Létocart, Anass Nagih

► **To cite this version:**

Nora Touati, Lucas Létocart, Anass Nagih. Sur l'accélération de la convergence de la génération de colonnes. 2006. hal-00145748

HAL Id: hal-00145748

<https://hal.science/hal-00145748v1>

Preprint submitted on 11 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.

Sur l'accélération de la convergence de la génération de colonnes

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

15 décembre 2006

Rapport de recherche

2006-4

Résumé

L'utilisation de la méthode de génération de colonnes s'est intensifiée ces dernières décennies, en particulier pour la résolution de problèmes discrets de grande taille afin d'obtenir une borne lagrangienne. Elle est également intégrée dans la méthode de séparation et d'évaluation (*branch-and-price*) où elle est utilisée pour le calcul de la fonction d'évaluation en chaque nœud de l'arbre de recherche.

À l'instar des méthodes itératives, elle souffre du phénomène d'amortissement, particulièrement lorsqu'elle est utilisée pour résoudre des problèmes dégénérés. Le schéma de génération de colonnes offre plusieurs possibilités d'améliorations. Nous présentons dans ce rapport quelques unes de ces possibilités proposées dans la littérature, qui se basent sur des principes variés selon que l'on opère sur le plan primal ou dual, que l'accélération concerne la résolution du problème maître, du sous-problème ou du processus global, ou encore son hybridation avec d'autres méthodes de résolution.

Nous mettons l'accent à la fin de ce rapport sur les techniques de réoptimisation qui s'avèrent prometteuses pour l'accélération de la génération de colonnes, particulièrement lorsque le temps global de résolution est dominé par celui des sous-problèmes.

Abstract

The use of the method of column generation was intensified during the last decades, in particular for the resolution of huge discrete problems in order to obtain a lagrangian bound. It is also integrated in a branch and bound method (*branch-and-price*) where it is used for the computation of the evaluation function at each node of the research tree.

Like iterative methods, it suffers from the long tail effect, particularly when it is used to solve degenerated problems. The scheme of column generation offers several possibilities for improvements. We present in this report some of these possibilities suggested in the literature, which are based on varied principles according to whether one operates on the primal or dual plan, or that acceleration relates to the resolution of the master problem, the subproblem or the whole process, or his hybridization with other resolution methods.

At the end of this report, we focus our study on reoptimization techniques, which prove to be promising for the acceleration of column generation, particularly when the total time of resolution is dominated by that of the subproblems.

1 Introduction

De nombreux problèmes rencontrés dans la pratique sont complexes, de grande taille et difficiles à résoudre. Ce qui les rend hors des capacités de résolution des logiciels disponibles aujourd'hui, même les plus spécialisés. Pour pouvoir les traiter, des méthodes de décomposition de l'espace des solutions sont utilisées [1, 9, 43]. La décomposition engendre souvent des problèmes avec un nombre démesuré de variables, d'où un intérêt croissant pour la méthode dite de *génération de colonnes* [3, 10, 13, 25, 29, 31] dont le point fort est de pouvoir caractériser une solution

optimale d'un problème en manipulant un ensemble réduit de ses variables.

La génération de colonnes a été introduite indépendamment par Dantzig et Wolfe [9] d'un côté et Gilmore et Gomory [20] de l'autre. Elle consiste à résoudre alternativement un problème maître et un sous-problème, résultants de la décomposition du problème d'origine. Le principe est de sélectionner à chaque itération une variable candidate à améliorer la valeur courante du problème maître, en exploitant le cas particulier où les variables ont une structure spécifique qui permet de les définir par un sous-problème d'optimisation (plus courts chemins [10, 13], sac-à-dos [20]).

À l'instar des méthodes itératives, la génération de colonnes peut souffrir d'un problème de convergence, particulièrement lorsqu'elle est utilisée pour résoudre des problèmes dégénérés où elle peut passer beaucoup de temps à générer des colonnes qui n'améliorent que faiblement la valeur de l'objectif, ce phénomène est appelé *l'amortissement de la génération de colonnes*. La version duale de la génération de colonnes, connue sous le nom de la méthode des plans coupants de Kelley [23], est basée sur le même principe, mais utilise des techniques de résolution différentes. Du point de vue dual, le problème de convergence se traduit par des variations d'une grande amplitude des solutions duales.

Plusieurs méthodes visant à améliorer la convergence de la génération de colonnes ont été proposées dans la littérature. Elles se basent sur des principes variés selon que l'on opère sur le plan primal ou dual, que l'accélération concerne la résolution du problème maître, du sous-problème ou bien du processus global, ou encore de son hybridation avec d'autres méthodes de résolution. Les méthodes les plus connues et les plus utilisées sont les méthodes de stabilisation [4, 10, 35, 34] qui opèrent sur le processus global en ayant comme objectif la diminution du nombre d'itérations en réduisant les oscillations des valeurs des variables duales.

Nous rappelons dans la section 2 le principe de la génération de colonnes du point de vue primal et dual et ses problèmes de convergence. Dans la section 3 nous exposerons différentes méthodes de stabilisation. Enfin dans la section 4, nous présenterons d'autres techniques d'accélération et des pistes de recherches à explorer qui nous semblent prometteuses.

2 Principe de la génération de colonnes

Considérons le programme linéaire à n variables et m contraintes :

$$(P) \quad \max \sum_{j \in J} c_j x_j \quad \text{s.c.} \quad \begin{aligned} \sum_{j \in J} A^j x_j &\leq a \\ x_j &\geq 0, \quad \forall j \in J \end{aligned}$$

où J est l'ensemble des indices des variables (avec $|J| = n$), $(c_j, A^j) \in (\mathbb{R} \times \mathbb{R}^m)$, $\forall j \in J$ et $a \in \mathbb{R}^m$.

Soient \mathcal{B} une base réalisable de (P) et $x_{(\mathcal{B})}$ une solution de base associée. La solution duale correspondante est donnée par :

$$\pi = c_{(\mathcal{B})}\mathcal{B}^{-1}$$

La résolution du problème (P) par l'algorithme primal du simplexe consiste à améliorer itérativement la solution de base $x_{(\mathcal{B})}$ en sélectionnant, parmi toutes les variables hors base, une variable de coût réduit améliorant, i.e. positif pour un problème en maximisation. Le coût réduit d'une variable x_j est donné par : $\bar{c}_j = c_j - \pi A^j$. La colonne qui rentre dans la base peut être obtenue en résolvant le problème d'évaluation :

$$\bar{c}_{j^*} = \max_{j \in J} c_j - \pi A^j \quad (1)$$

Si $\bar{c}_{j^*} > 0$ alors la solution courante peut être améliorée en introduisant x_{j^*} dans la base. Sinon, la base courante \mathcal{B} est optimale pour (P) .

Si on est amené à manipuler un nombre démesuré de variables (plusieurs millions), il est alors très difficile de les énumérer explicitement et par conséquent, la résolution de (1) devient ardue, voire impossible. Dans le cas où les colonnes ont une structure particulière, on peut les définir en résolvant un problème d'optimisation [43]. Par exemple, le problème de découpe [20] dans lequel les colonnes sont des vecteurs à composantes entières non nulles, la définition d'une colonne revient à résoudre un problème de sac-à-dos à variables entières. Les méthodes de décomposition reposent sur le même principe : le problème d'origine est décomposé en un ou plusieurs sous-problèmes plus simples à résoudre, l'ensemble étant généralement coordonné par un programme linéaire appelé le problème maître.

L'algorithme de génération de colonnes est particulièrement adapté à la résolution des problèmes engendrés par la décomposition de Dantzig et Wolfe [9]. Son principe est similaire à celui de l'algorithme du simplexe et consiste à résoudre le problème (P) avec un ensemble réduit de ses variables, puis à l'alimenter itérativement avec de nouvelles colonnes jusqu'à atteindre l'optimalité. Ainsi, on construit à une itération k donnée, un problème (P^k) (appelé problème maître restreint) à partir de (P) en utilisant un sous-ensemble de colonnes $J^k \subseteq J$:

$$(P^k) \quad \max \sum_{j \in J^k} c_j x_j \quad \text{s.c.} \quad \begin{aligned} \sum_{j \in J^k} A^j x_j &\leq a \\ x_j &\geq 0, \quad \forall j \in J^k \end{aligned}$$

Soient \mathcal{B}^k une base optimale et réalisable de (P^k) , x^k et π^k les solutions primales et duales associées. On cherche donc une colonne de coût réduit positif (maximal) qui rentre dans la base en résolvant le problème d'évaluation (1) appelé sous-problème. Deux cas peuvent se présenter :

- (i) Soit $\bar{c}_{j^*} > 0$: la colonne j^* est rajoutée à l'ensemble J^k et le problème (P^k) augmenté d'une colonne, est réoptimisé sur l'ensemble $J^{k+1} = J^k \cup \{j^*\}$.
- (ii) Ou bien $\bar{c}_{j^*} \leq 0$: aucune variable ne possède un coût réduit positif et par conséquent la base \mathcal{B}^k est également optimale pour (P) .

L'algorithme suivant résume les étapes de résolution par la génération de colonnes.

Schéma générique de l'algorithme de génération de colonnes

<pre> {début} k ← 1 Initialiser (P^k) J^k ← Ensemble des indices des variables de la base de départ Répéter Résoudre (P^k) B^k base réalisable et optimale de (P^k) (x^k; π^k) ← solution optimale (primale; duale) de (P^k) c̄* ← max_{j∈J}{c_j - π^kA^j} j* ← argmax_{j∈J}{c_j - π^kA^j} J^{k+1} ← J^k ∪ {j*}, k ← k + 1 Jusqu'à c̄* ≥ 0 B^{k-1} est réalisable et optimale pour (P) {fin} </pre>

2.1 La génération de colonnes pour l'optimisation discrète

La résolution des problèmes en nombres entiers nécessite souvent l'utilisation de techniques de décompositions [9, 17, 19, 22, 43]. Nous nous intéresserons dans ce qui suit à la méthode de décomposition de Dantzig et Wolfe [9] dont l'application aboutit, d'une part à un problème maître et d'autre part à un ou plusieurs sous-problèmes [2, 3, 12, 13, 20, 21, 33]. Pour résoudre le programme en nombres entiers ainsi obtenu, la génération de colonnes peut être utilisée directement [28, 29] ou intégrée à un schéma de type *branch-and-bound* pour donner la méthode de *branch-and-price*.

Considérons le programme linéaire suivant :

$$(PNE) \quad \max cx \quad \text{s.c.} \quad \begin{array}{l} Ax \leq a \quad (m \text{ contraintes}) \\ Bx \leq b \quad (p \text{ contraintes}) \\ x \in \mathbb{N}^n \end{array}$$

où $A \in \mathbb{R}^{m \times n}$, $B \in \mathbb{R}^{p \times n}$, $a \in \mathbb{R}^m$ et $b \in \mathbb{R}^p$.

Soit $X = \{x \in \mathbb{N}^n : Bx \leq b\}$ un ensemble fini indicé par I . En notant x^i , $i \in I$, les éléments de X , le problème (PNE) peut se réécrire comme suit :

$$\max\{cx^i : Ax^i \leq a, i \in I\}$$

En associant une variable binaire λ_i , $i \in I$, à chaque élément x^i de X , on obtient la formulation du problème maître de Dantzig et Wolfe équivalent au problème (PNE) :

$$\max \sum_{i \in I} (cx^i) \lambda_i \quad \text{s.c.} \quad \begin{array}{l} \sum_{i \in I} (Ax^i) \lambda_i \leq a \\ \sum_{i \in I} \lambda_i = 1 \\ \lambda_i \in \mathbb{N}, \quad \forall i \in I \end{array} \quad (2)$$

En relâchant les contraintes d'intégrité sur les variables du problème (2) et en substituant I par J , l'ensemble des indices des points extrêmes de $Conv(X)$, on obtient la relaxation continue du problème maître de Dantzig-Wolfe :

$$(PM) \quad \max \sum_{j \in J} (cx^j) \lambda_j \quad \text{s.c.} \quad \begin{aligned} \sum_{j \in J} (Ax^j) \lambda_j &\leq a \\ \sum_{j \in J} \lambda_j &= 1 \\ \lambda_j &\geq 0, \quad \forall j \in J \end{aligned} \quad (3)$$

Le nombre de points extrêmes étant potentiellement exponentiel, on définit le problème maître (PM^k) restreint au sous-ensemble des variables d'indices dans $J^k \subseteq J$ construit itérativement avec l'algorithme de génération de colonnes (voir procédure décrite ci-dessus).

D'après le paragraphe 2, le coût réduit d'une variable λ_j , $j \in J$, est donné par :

$$cx^j - \pi(Ax^j) - \pi_0$$

et le problème d'évaluation appelé aussi le sous-problème est formulé comme suit :

$$(SP) \quad \max_{j \in J} \{(c - \pi A)x^j - \pi_0\} \Leftrightarrow \max_{x \in Conv(X)} \{(c - \pi A)x - \pi_0\}$$

où $\pi \in \mathbb{R}_+^m$ est le vecteur des variables duales correspondant aux contraintes $\sum_{j \in J} (Ax^j) \lambda_j \leq a$ (dites couplantes) et $\pi_0 \in \mathbb{R}$ la variable duale correspondant à la contrainte de convexité $\sum_{j \in J} \lambda_j = 1$.

Remarques 1

- (a) *Le problème maître peut être initialisé avec un ensemble de colonnes générées par une heuristique ou bien en utilisant des variables artificielles à l'instar de la phase 1 de l'algorithme du simplexe.*
- (b) *Il n'est pas nécessaire de résoudre les sous-problèmes jusqu'à l'optimalité, une méthode heuristique peut fournir des colonnes de coût réduit positif en un temps raisonnable [37], la résolution exacte des sous-problèmes n'est nécessaire que pour prouver l'optimalité.*
- (c) *Afin de réduire le nombre d'itérations de la génération de colonnes, il est préférable de générer lors de chaque itération un ensemble de colonnes de coûts réduits positifs (cela revient à déterminer plusieurs solutions du sous-problème de coûts réduits positifs en utilisant des heuristiques ou bien en calculant les k meilleures solutions) que de n'en générer qu'une seule.*
- (d) *Les problèmes (PNE) et (2) sont équivalents mais leurs relaxations continues respectives sont différentes. La relaxation continue de (2) correspond à la relaxation lagrangienne des contraintes $Ax \leq a$ [39].*
- (e) *D'après la remarque (1-(b)) et lorsque le sous-problème possède la propriété d'intégralité¹ (problème du plus court chemin [10] par exemple), la borne obtenue est la même que celle de la relaxation continue. Afin d'obtenir une meilleure borne, il est préférable que le sous-problème ne possède pas la propriété d'intégralité.*

¹La résolution de la relaxation continue d'un problème en nombres entiers possédant la propriété d'intégralité fournit des solutions entières.

2.2 La Génération de Colonnes dans l'espace dual

Considérons le dual du problème (PM) :

$$(DM) \quad \min \pi a + \pi_0 \quad \text{s.c.} \quad \begin{aligned} \pi_0 &\geq cx^j - \pi(Ax^j) \quad \forall j \in J \\ (\pi, \pi_0) &\in (\mathbb{R}_+^m, \mathbb{R}) \end{aligned}$$

Ce problème peut être résolu par la méthode des plans coupants de Kelley [23].

Posons $\theta = \pi a + \pi_0$, le problème (DM) peut être réécrit comme suit :

$$(D) \quad \min \theta \quad \text{s.c.} \quad \begin{aligned} \theta &\geq cx^j + \pi(a - Ax^j) \quad \forall j \in J \\ (\theta, \pi) &\in (\mathbb{R}, \mathbb{R}_+^m) \end{aligned} \quad (4)$$

il en découle les formulations équivalentes :

$$\min \theta \quad \text{s.c.} \quad \begin{aligned} \theta &\geq \Theta(\pi) \\ (\theta, \pi) &\in (\mathbb{R}, \mathbb{R}_+^m) \end{aligned} \quad \Leftrightarrow \quad \min_{\pi \in \mathbb{R}_+^m} \Theta(\pi)$$

où $\Theta(\pi) = \max_{j \in J} \{cx^j + \pi(a - Ax^j)\}$.

Remarques 2

- (a) La fonction $\pi \mapsto \Theta(\pi)$ est une fonction convexe et linéaire par morceaux.
- (b) Pour tout $\pi \in \mathbb{R}_+^m$, $M = \max_{j \in J} \{cx^j + \pi(a - Ax^j)\}$ définit un majorant de la valeur optimale du problème maître (PM) .
- (c) Le problème (4) est équivalent au dual de la relaxation lagrangienne du problème (PNE) obtenu en dualisant les contraintes $Ax \leq a$.

Par analogie avec la procédure présentée au début de la section 2, on définit le problème maître (D^k) restreint à un ensemble réduit de contraintes générées par le sous-ensemble de colonnes $J^k \subseteq J$:

$$(D^k) \quad \min \theta \quad \text{s.c.} \quad \begin{aligned} \theta &\geq cx^j + \pi(a - Ax^j) \quad \forall j \in J^k \\ (\theta, \pi) &\in (\mathbb{R}, \mathbb{R}_+^m) \end{aligned} \quad (5)$$

Ces contraintes qui sont (éventuellement) des supports de la courbe de Θ , définissent une approximation Θ^k de Θ . La valeur de $\Theta(\pi)$ est fournie par la résolution du sous-problème

$$\max_{j \in J^k} \{cx^j + \pi(a - Ax^j)\} \quad (6)$$

Soit π^k la solution optimale de (D^k) , pour vérifier si π^k est optimale pour (D) , on calcule la valeur de $\Theta(\pi^k)$ en résolvant le sous-problème (6). Deux cas peuvent se présenter :

- (i) Soit $\Theta(\pi^k) > \Theta^k(\pi^k)$, la solution π^k n'est pas optimale pour (D) et la nouvelle coupe associée à la colonne générée par le sous-problème est rajoutée à l'ensemble des coupes déjà générées pour obtenir un nouveau problème (D^{k+1}) , et le processus est réitéré.

(ii) Soit $\Theta(\pi^k) \leq \Theta^k(\pi^k)$, dans ce cas π^k est optimale pour (D) et l'algorithme s'arrête.

Chaque coupe dans le dual correspond à une colonne dans le primal, le théorème ci-dessous permet de construire une solution primale réalisable à partir des conditions d'optimalité du problème dual restreint [7].

Théorème 1 (1) La solution optimale π^k de (D^k) est caractérisée par l'existence d'une solution primale réalisable x^k satisfaisant les conditions de complémentarité :

$$Ax^k - a \leq 0, \quad \pi^k \geq 0, \quad (Ax^k - a)\pi^k = 0$$

(2) x^k coïncide avec la solution optimale de (PM^k) .

(3) x^k est optimale pour (PM) si $cx^k = \Theta(\pi)$, avec π la meilleure solution duale courante.

2.3 Convergence de la génération de colonnes

La génération de colonnes assure une très rapide amélioration de la valeur du problème maître durant les premières itérations. Cependant, elle peut passer beaucoup de temps à générer des colonnes qui n'améliorent que faiblement la valeur de l'objectif. La figure 1-(a) illustre l'effet d'amortissement lors des dernières itérations.

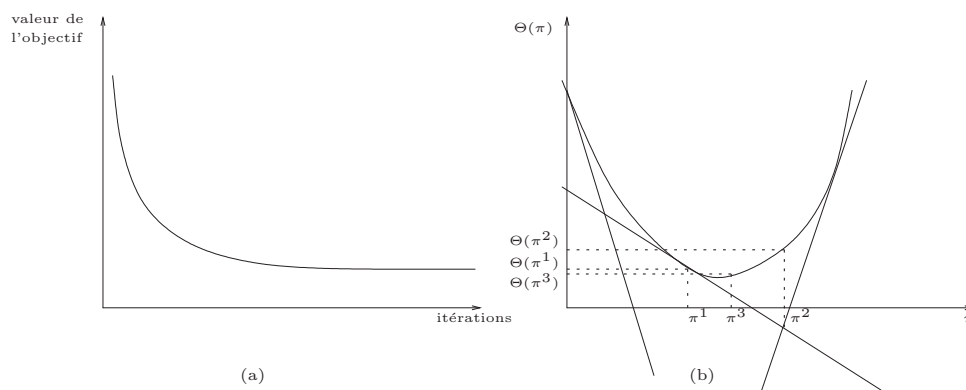


FIG. 1 – Convergence de la génération de colonnes du point de vu primal et dual

Les problèmes de convergence de la génération de colonnes proviennent essentiellement de la dégénérescence primale et de la qualité des colonnes générées qui n'appartiendront pas à la base optimale. Sous l'aspect dual, ces problèmes de convergence se traduisent par des oscillations des valeurs des variables duales où l'on observe des déplacements qui causent une détérioration de la qualité de la solution (figure 1-(b)). L'étude de la convergence de la génération de colonnes du point de vue dual a été la source du développement des méthodes de stabilisation, dont le principe de base est de générer des solutions proches de la meilleure solution courante, pour diminuer les oscillations des valeurs des variables duales. Une synthèse des différentes méthodes est présentée dans la section suivante.

3 Méthodes de stabilisation

Il existe dans la littérature plusieurs approches dites de stabilisation qui visent à diminuer les fluctuations des variables duales en guidant la méthode pour générer des solutions proches de la meilleure solution trouvée jusqu'à l'itération courante. Ces méthodes opèrent donc sur l'espace dual et ont pour effet de diminuer le nombre d'itérations. Le principe de base de ces méthodes est de restreindre la résolution du problème dual (4) au voisinage de la meilleure solution trouvée appelée *le centre de stabilité*. Nous présentons dans ce qui suit deux types de stabilisation des variables duales. Le premier, basé sur la pénalisation du problème dual, englobe les méthodes de type faisceaux et les méthodes de stabilisation par pénalités linéaires et consiste à rajouter un terme de stabilisation à l'objectif du problème dual qui a pour effet de générer des solutions proches du centre de stabilité. Le second, englobe les méthodes du centre qui déterminent une localisation (bornée, convexe et fermée) et calculent un point à l'intérieur de cet ensemble. Nous présenterons à la fin de cette section une méthode basée sur un autre principe, qui est de perturber la solution duale.

3.1 Méthodes de stabilisation par pénalisation du problème dual

La méthode du point proximal a été la première technique à introduire ce concept [44], pour l'optimisation des fonctions convexes non différentiables. Le principe est d'approcher la fonction Θ par une fonction $\widehat{\Theta}$ également convexe et possédant les mêmes points minimaux que Θ et de plus différentiable. Malheureusement, le problème à résoudre lors de chaque itération est aussi difficile que le problème d'origine. Afin de remédier à cela, nous avons besoin de réunir les informations déjà obtenues sur Θ lors des itérations précédentes pour faciliter le calcul de la prochaine approximation, d'où l'idée d'utiliser la méthode des plans coupants.

Les méthodes présentées dans cette section consistent à rajouter un terme de stabilisation à l'objectif du problème dual afin de pénaliser les solutions éloignées du centre de stabilisation. Cette pénalisation permet de réduire l'espace des réalisables et à pour effet de diminuer la complexité de résolution du problème (D). Le problème stabilisé est formulé comme suit :

$$(D_{sta}) \quad \min_{\pi \in \mathbb{R}^m} \widehat{\Theta}(\pi) \quad \text{où} \quad \widehat{\Theta}(\pi) = \Theta(\pi) + F(\pi - \bar{\pi})$$

où $\bar{\pi}$ est le centre de stabilité et F une fonction convexe qui atteint son minimum au point 0. On utilise la méthode des plans coupants pour résoudre le problème (D_{sta}). On définit le problème stabilisé (D_{sta}^k) à l'itération k comme suit :

$$(D_{sta}^k) \quad \min_{\pi \in \mathbb{R}^m} \widehat{\Theta}^k(\pi) \quad \text{où} \quad \widehat{\Theta}^k(\pi) = \Theta^k(\pi) + F(\pi - \bar{\pi}) \quad (7)$$

Les principales étapes des méthodes de stabilisation sont résumées ci-dessous :

Algorithme général de stabilisation de la génération de colonnes

{*début*}

Choisir un centre de stabilité $\bar{\pi}$ et un ensemble de colonnes de départ J^0
Poser $k \leftarrow 0$ et fixer $\varepsilon > 0$ petit

Faire

Résoudre le problème (7), soit π^k une solution optimale

Calculer $\Theta(\pi^k)$ en résolvant le sous-problème (6)

Si $(\Theta(\bar{\pi}) > \Theta(\pi^k))$ **Alors** $\bar{\pi} \leftarrow \pi^k$ **FinSi**

Enrichir Θ^k avec le plan coupant obtenu

Jusqu'à $(\Theta(\bar{\pi}) - \Theta^k(\pi^k)) < \varepsilon$

{*fin*}

L'algorithme s'arrête lorsque l'écart entre le majorant $\Theta(\bar{\pi})$ et le minorant $\Theta^k(\pi^k)$ est suffisamment petit (c'est-à-dire inférieur à ε et la solution obtenue est ε -optimale).

Résolution du problème stabilisé

Ajouter des contraintes au dual restreint correspond à rajouter des colonnes au problème maître restreint, le théorème 1 permet de calculer une solution primale réalisable du problème maître restreint à partir d'une solution duale. Notons qu'une solution π^k du problème (D_{sta}^k) n'est pas forcément optimale pour (D^k) , afin d'obtenir une solution primale réalisable, le problème maître restreint doit être résolu en incluant toutes les colonnes générées. Afin d'éviter ce calcul, il serait avantageux de formuler le dual du problème (7) en fonction des variables d'origine, cela devient possible en utilisant les fonctions conjuguées et la dualité de Fenchel [7].

Le problème (D_{sta}^k) peut être reformulé comme suit :

$$\min_{(\pi, \gamma) \in \mathbb{R}_+^m \times \mathbb{R}^m} \Theta^k(\pi) + F(\gamma), \quad \gamma = \pi - \hat{\pi}$$

Le dual de la relaxation lagrangienne de cette nouvelle formulation obtenue en relâchant les contraintes $\gamma = \pi - \hat{\pi}$ suivant le multiplicateur $y \in \mathbb{R}^m$ est appelé le dual de Fenchel :

$$\begin{aligned} (DF^k) &\equiv \max_{y \in \mathbb{R}^m} \min_{(\pi, \gamma) \in \mathbb{R}_+^m \times \mathbb{R}^m} \Theta^k(\pi) - y\pi + F(\gamma) + y\gamma \\ &\equiv \max_{y \in \mathbb{R}^m} [\min_{\pi \in \mathbb{R}_+^m} (\Theta^k(\pi) - y\pi) + \min_{\gamma \in \mathbb{R}^m} (F(\gamma) + y\gamma)] \\ &\equiv \max_{y \in \mathbb{R}^m} [-\max_{\pi \in \mathbb{R}_+^m} (y\pi - \Theta^k(\pi)) - \max_{\gamma \in \mathbb{R}^m} ((-y)\gamma - F(\gamma))] \end{aligned}$$

On définit T^* , la fonction conjuguée d'une fonction T comme suit :

$$T^*(u) = \max_v uv - T(v)$$

Remarquons que " $\max_{\pi \in \mathbb{R}_+^m} (y\pi - \Theta^k(\pi)) = \Theta^{k*}(y)$ " et " $\max_{\gamma \in \mathbb{R}^m} ((-y)\gamma - F(\gamma)) = F^*(-y)$ ". Le dual de fenchel du problème (D_{sta}^k) peut donc se reformuler comme suit :

$$(DF^k) \equiv \max_{y \in \mathbb{R}^m} -\Theta^*(y) - F^*(-y)$$

Le terme $-\Theta^*(y)$ peut être développé comme suit :

$$\begin{aligned} -\Theta^*(y) &= -\max\{y\pi - \Theta^k(\pi) \mid \pi \in \mathbb{R}_+^m\} \\ &\equiv -\max_{\pi \in \mathbb{R}_+^m} \{y\pi - \max_{j \in J^k} \{cx^j + \pi(a - A^j x^j)\}\} \\ &\equiv -\max_{\pi \in \mathbb{R}_+^m} \{y\pi - \max_{x \in \text{Conv}(X^k)} \{cx + \pi(a - Ax)\}\} \\ &\equiv -\max_{\pi \in \mathbb{R}_+^m} \{y\pi + \min_{x \in \text{Conv}(X^k)} \{-cx - \pi(a - Ax)\}\} \\ &\equiv -\min_{x \in \text{Conv}(X^k)} \{-cx + \max_{\pi \in \mathbb{R}_+^m} \{\pi(y + a - Ax)\}\} \end{aligned}$$

Le problème (DF^k) devient :

$$\begin{aligned} \max \quad & cx + y\hat{\pi} - F^*(y) \\ & Ax - a + y = 0 \\ & x \in \text{Conv}(X^k) \\ & y \in \mathbb{R}^m \end{aligned} \tag{8}$$

Une correspondance similaire que celle du théorème (1) peut être établie entre les problèmes (8), (PM^k) et (PM) (voir [7]).

3.1.1 Les méthodes de type faisceaux

Ces méthodes ont été introduites dans [26] et [51]. Elles ont été conçues à l'origine pour l'optimisation des fonctions convexes non différentiables avec ou sans contraintes. Elles peuvent être vues comme des méthodes de stabilisation de la méthode des plans coupants.

Soit $\bar{\pi}$ la meilleure solution obtenue jusqu'à l'itération courante et $g(\bar{\pi})$ un sous-gradient de Θ en $\bar{\pi}$ vérifiant donc la propriété suivante :

$$\Theta(\pi) \geq \Theta(\bar{\pi}) + g(\bar{\pi})(\pi - \bar{\pi}) \quad \forall \pi \in \mathbb{R}_+^m$$

Le principe de base des méthodes des faisceaux est de collecter un faisceau d'informations sous la forme $\{(\Theta(\pi^j), g(\pi^j)), j \in J^k\}$ généré jusqu'à l'itération k pour tenter de calculer une direction de descente pour Θ au point $\bar{\pi}$. Étant donné que la direction obtenue n'est pas forcément une direction de descente [6], cette méthode consiste à se déplacer vers une meilleure solution si la valeur de l'objectif est améliorée significativement (étape de descente) ou à rester dans le point courant (étape nulle) et utiliser le sous-gradient généré afin d'enrichir le faisceau courant pour tenter de trouver une meilleure direction.

L'approximation Θ^k de Θ est donnée par

$$\Theta^k(\pi) = \max_{j \in J^k} \{\Theta(\pi^j) + g^j(\pi - \pi^j)\}$$

Où $g^j = g(\pi^j)$. En posant $d = \pi - \bar{\pi}$, $\Theta^k(\pi)$ s'écrira :

$$\Theta_{\bar{\pi}}^k(d) = \max_{j \in J^k} \{\alpha^j + dg^j\} + \Theta(\bar{\pi})$$

où $\alpha^j = \Theta(\pi^j) + g^j(\bar{\pi} - \pi^j) - \Theta(\bar{\pi})$

Le prochain point désiré est $\bar{\pi} + d^*$ où d^* est solution optimale du problème

$$\min_d \{\Theta_{\bar{\pi}}^k(d)\} = \min_{\theta, d} \{ \theta : \theta \geq \alpha^j + dg^j, j \in J^k \} + \Theta(\bar{\pi})$$

Remarques 3

- (a) La transformation ci-dessus est une translation de Θ au repère d'origine $(\bar{\pi}, \Theta(\bar{\pi}))$.
- (b) La méthode des plans coupants peut être vue comme une méthode de calcul d'une direction de recherche de nouvelles solutions. Si $\Theta_{\bar{\pi}}^k(d^*) = \Theta(\bar{\pi} + d^*)$ alors $\pi = \bar{\pi} + d^*$ est optimale pour (D) , sinon, $(\Theta(\pi), g(\pi))$ est rajouté au faisceau courant.

Étant donné que Θ est non différentiable, chaque point $\bar{\pi}$ est caractérisé par un sous-différentiel $\partial\Theta(\bar{\pi})$, ensemble de tous les sous-gradients de Θ en $\bar{\pi}$. La recherche d'une direction de descente pour la fonction Θ au point $\bar{\pi}$ revient à la recherche d'un vecteur de norme minimale dans $\partial\Theta(\bar{\pi})$ qui est malheureusement difficile à caractériser [18]. Une idée est alors de calculer une approximation de $\partial\Theta(\bar{\pi})$. La méthode la plus simple est la méthode de sous-gradient (ou de manière équivalente la méthode des plans coupants) qui utilise un sous-gradient $g(\bar{\pi})$ comme approximation de $\partial\Theta(\bar{\pi})$. Cette méthode est simple et d'une large utilisation, mais elle souffre d'un problème de convergence.

Afin d'approcher $\partial\Theta(\bar{\pi})$, d'autres méthodes se basent sur le principe suivant : on espère que les sous-gradients générés en des points proches de $\bar{\pi}$ sont proches de certains sous-gradients de $\partial\Theta(\bar{\pi})$. Au lieu d'utiliser un seul sous-gradient, on utilise un faisceau de sous-gradients autour de $\bar{\pi}$, cet ensemble est appelé l' ε -sous-différentiel de Θ en $\bar{\pi}$ et est noté $\partial_\varepsilon\Theta(\bar{\pi})$ (pour plus de détails, voir [18]).

Lorsque la fonction à optimiser est convexe et linéaire par morceaux (dans notre cas Θ), le problème de la recherche d'une direction de descente est le dual du problème d'origine (4) auquel on rajoute un terme quadratique à l'objectif, cette technique a donné lieu à la première variante des méthodes des faisceaux pour la stabilisation de la méthode des plans coupants qui est présentée ci-dessous.

i) Stabilisation par pénalisation quadratique

Le problème résolu lors de chaque itération est le suivant :

$$\min_{\pi \in \mathbb{R}^m} \{ \Theta^k(\pi) + \frac{1}{2t} \| \pi - \bar{\pi} \|^2 \}$$

Ce problème est équivalent au problème (D) auquel on a rajouté le terme de stabilisation $\frac{1}{2t} \| \pi - \bar{\pi} \|^2$, qui a pour objectif de pénaliser les solutions éloignées de $\bar{\pi}$. Le prochain point est obtenu en résolvant le problème quadratique :

$$\min \theta + \frac{1}{2t} \| \pi - \bar{\pi} \|^2 : \quad \theta \leq \Theta(\bar{\pi}) + g^j(\pi - \bar{\pi}) \quad \forall j \in J^k \\ (\theta, \pi) \in \mathbb{R} \times \mathbb{R}_+^m$$

Le paramètre t est appelé coefficient de stabilisation et a pour rôle de régler l'effet de stabilisation. Si t est très grand, la stabilisation est faible (i.e. on tolère les

éloignements de $\bar{\pi}$), si t est très petit, la stabilisation est forte et on risque d'avoir des améliorations très faibles de la qualité de la solution ; la figure 2 montre l'effet de la stabilisation en fonction du paramètre t .

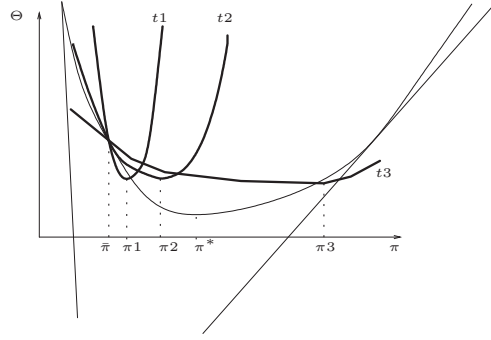


FIG. 2 – Stabilisation avec différentes valeurs de t : $t1 < t2 < t3$

ii) Stabilisation par région de confiance (Trust region)

Cette méthode a été introduite dans [45], elle offre une technique simple pour la mise à jour de t . Dans le cas d'une étape de descente, élargir t peut être préférable au déplacement vers un nouveau point, car l'approximation Θ^k de Θ peut être meilleure à l'extérieur de la région de confiance définie par la fonction de pénalisation. Dans le cas d'une étape nulle, décroître t peut être préférable à l'insertion d'un nouveau sous-gradient.

iii) Stabilisation par niveaux (Level stabilization)

Cette méthode a été introduite dans [27], le principe est de contrôler la valeur de l'objectif avec un niveau l_k lors de chaque itération k . Le problème stabilisé a la forme suivante :

$$\min_{\pi \in \mathbb{R}_+^m} \left\{ \frac{1}{2} \|\pi - \bar{\pi}\|^2 : \Theta^k(\pi) \leq l_k \right\} \quad (9)$$

Le principe est le suivant :

Calculer l'écart $\Delta(k) = \bar{\theta}^k - \underline{\theta}^k$, tel que $\bar{\theta}^k$ (resp. $\underline{\theta}^k$) est le meilleur majorant (resp. minorant) de Θ à l'itération k . π^k est une solution au plus $\Delta(k)$ -optimale pour Θ , l'objectif est donc de forcer l'écart entre le majorant et le minorant à s'annuler. Ayant $\lambda \in [0, 1]$ on résout (9) avec la valeur

$$l_k = \lambda \bar{\theta}^k + (1 - \lambda) \underline{\theta}^k = \underline{\theta}^k + \lambda \Delta(k)$$

Fixer $\lambda = 1$ entrainera $l_k = \bar{\theta}^k$, cela revient à fixer $\Theta(\bar{\pi})$ comme majorant, donc aucune information supplémentaire n'est apportée au problème et la solution optimale restera $\bar{\pi}$, d'un autre côté, si $\lambda = 0$ on obtient la même solution que la méthode des plans coupants, donc la même convergence. Il est donc conseillé d'attribuer de petites valeurs à λ . Des règles simples peuvent être appliquées pour mettre à jour l_k (voir [27]). Cette technique permet d'éviter les pas nuls et donc

de diminuer le nombre d'itérations de la génération de colonnes.

3.1.2 Méthodes de stabilisation par pénalisations linéaires

Cette classe de méthodes englobe celles qui utilisent une fonction de stabilisation linéaire par morceaux, nous présentons ci-dessous quelques variantes.

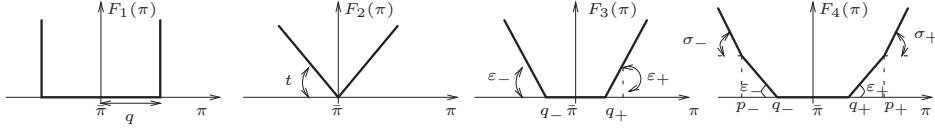


FIG. 3 – Quelques fonctions de stabilisation

i) Méthode du Boxstep

La méthode du boxstep a été proposée dans [32], le problème (4) est résolu sur une « boîte » (un hyper-cube) de largeur $2q$, $q \in \mathbb{R}_+$ centrée au centre de stabilité $\bar{\pi}$. Le modèle stabilisé est formulé comme suit :

$$\min_{\pi \in \mathbb{R}^m} \Theta^k(\pi) + F_1(\pi - \bar{\pi})$$

La fonction de stabilisation F_1 est illustrée dans la figure (3). Posons $\bar{\pi} - q = q_-$ et $\bar{\pi} + q = q_+$. La formulation linéaire du problème stabilisé est donnée par (DB) dont le dual est (PB) :

$$(DB) \begin{cases} \min \theta \\ \theta \geq cx^j + \pi(a - Ax^j) & j \in J^k \\ q_- \leq \pi_i \leq q_+ & i = 1, \dots, m \\ (\theta, \pi) \in (\mathbb{R}, \mathbb{R}_+^m) \end{cases} \quad (PB) \begin{cases} \max \sum_{j \in J} cx^j \lambda_j + q_- y_- - q_+ y_+ \\ \sum_{j \in J} Ax^j \lambda_j + y_- - y_+ \leq a \\ \sum_{j \in J} \lambda_j = 1 \\ \lambda_j \in \mathbb{R}_+, \forall j \in J \\ y_-, y_+ \in \mathbb{R}_+^m \end{cases}$$

La largeur de la boîte doit être fixée efficacement. Si q est grand, il n'y a pas d'effet de stabilisation et si q est petit, la convergence devient lente. Contrairement à la méthode des plans coupants, cette méthode ne fournit pas, lors de chaque itération, un minorant de la valeur de la solution optimale.

ii) Méthode proximale avec une norme L_1

Les auteurs de [24] ont utilisé la fonction $F_2(\pi) = t\|\pi - \bar{\pi}\|_1$ (voir la figure 3) comme fonction de stabilisation. Le problème stabilisé s'écrit :

$$\min_{\pi \in \mathbb{R}^m} \Theta^k + t\|\pi - \bar{\pi}\|_1$$

iii) Stabilisation avec une fonction linéaire à 3 morceaux

Cette méthode a été proposée dans [35], elle consiste à combiner la méthode du boxstep avec la méthode des perturbations. Alors que dans la méthode du boxstep,

les nouvelles solutions doivent appartenir à une boîte, cette méthode autorise des solutions à l'extérieur de la boîte mais impose un prix à payer. Nous présentons ci-dessous le problème perturbé (PP) et le problème stabilisé (PB) par la méthode du boxstep du problème (4) :

$$(PP) \left\{ \begin{array}{l} \max \sum_{j \in J} cx^j \lambda_j \\ \sum_{j \in J} Ax^j \lambda_j + y_- - y_+ \leq a \\ \sum_{j \in J} \lambda_j = 1 \\ y_- \leq \varepsilon_- \\ y_+ \leq \varepsilon_+ \\ \lambda_j \geq 0, \forall j \in J \\ y_-, y_+ \in \mathbb{R}_+^m \end{array} \right. \quad (PB) \left\{ \begin{array}{l} \max \sum_{j \in J} cx^j \lambda_j + q_- y_- - q_+ y_+ \\ \sum_{j \in J} Ax^j \lambda_j + y_- - y_+ \leq a \\ \sum_{j \in J} \lambda_j = 1 \\ \lambda_j \geq 0, \forall j \in J \\ y_-, y_+ \in \mathbb{R}_+^m \end{array} \right.$$

Où y_- et y_+ sont des variables de surplus et d'écart, auxquelles on associe respectivement des majorants ε_- , $\varepsilon_+ \in \mathbb{R}_+^m$ dans (PP) et des coûts q_- , $q_+ \in \mathbb{R}_+^m$ dans (PB).

De la combinaison des deux problèmes ci-dessus découle la formulation stabilisée (SP), soit (SD) le dual correspondant :

$$(SP) \left\{ \begin{array}{l} \max \sum_{j \in J} cx^j \lambda_j + q_- y_- - q_+ y_+ \\ \sum_{j \in J} Ax^j \lambda_j + y_- - y_+ \leq a \\ \sum_{j \in J} \lambda_j = 1 \\ y_- \leq \varepsilon_- \\ y_+ \leq \varepsilon_+ \\ \lambda_j \geq 0, \forall j \in J \\ y_-, y_+ \in \mathbb{R}_+^m \end{array} \right. \quad (SD) \left\{ \begin{array}{l} \min \pi a + \pi_0 + \varepsilon_- \omega_- + \varepsilon_+ \omega_+ \\ \pi^j Ax^j + \pi_0 \leq cx^j, j \in J \\ q_- - \omega_- \leq \pi \leq q_+ + \omega_+ \\ \pi, \omega_+, \omega_- \in \mathbb{R}_+^m \end{array} \right.$$

La figure 3 présente la fonction de stabilisation, linéaire à 3 morceaux F_3 .

Remarques 4

- $[q_-, q_+]$ est l'intervalle dans lequel la pénalité est nulle ($\omega_- = \omega_+ = 0$) et ε_- et ε_+ sont les vecteurs de pénalité correspondants à π suivant que l'on se trouve à gauche ou à droite de l'intervalle $[q_-, q_+]$.
- Lorsque la pénalité est infinie, cette méthode est équivalente à la méthode du boxstep.
- Si $q_+ = q_- = \bar{\pi}$ et $\varepsilon_+ = \varepsilon_- = (\varepsilon, \dots, \varepsilon)$, on obtient une méthode de stabilisation utilisant la norme linéaire $\|\cdot\|_1$.

iv) Stabilisation avec une fonction linéaire à 5 morceaux

Les auteurs de [5] ont proposé une méthode de stabilisation par une fonction linéaire à 5 morceaux, représentée par la fonction F_4 sur la figure 3. Les problèmes stabilisés, primal et dual, sont donnés ci-dessous :

$$(SPP) \left\{ \begin{array}{l} \max \sum_{j \in J} cx^j \lambda_j + p_- z_- + q_- y_- - q_+ y_+ - p_- z_- \\ \sum_{j \in J} Ax^j \lambda_j - z_- - y_- + y_+ + z_+ \leq a \\ \sum_{j \in J} \lambda_j = 1 \\ y_- \leq \varepsilon_-, z_- \leq \sigma_- \\ y_+ \leq \varepsilon_+, z_+ \leq \sigma_+ \\ \lambda_j \geq 0, \forall j \in J \\ z_-, y_-, z_+, y_+ \in \mathbb{R}_+^m \end{array} \right. \quad (10)$$

$$(SDD) \left\{ \begin{array}{l} \min \pi a + \pi_0 + \sigma_- \beta_- + \varepsilon_- \omega_- + \varepsilon_+ \omega_+ + \sigma_+ \beta_+ \\ \pi_j Ax^j + \pi_0 \leq cx^j, \quad j \in J \\ q_- - \omega_- \leq \pi \leq q_+ + \omega_+ \\ p_- - \beta_- \leq \pi \leq p_+ + \beta_+ \\ \omega_+, \omega_-, \beta_+, \beta_- \geq 0 \end{array} \right. \quad (11)$$

Remarques 5

- (a) les problèmes (SPP) et (SDD) possèdent respectivement $4m$ variables et $4m$ contraintes de plus que (PM) et (D).
- (b) Si une solution optimale π^* de (D) appartient à l'intervalle $[q_-, q_+]$ alors les problèmes (D) et (SDD) ont les mêmes solutions optimales.
- (c) Si $q_- = p_- = -M$, $q_+ = p_+ = +M$ et $\varepsilon_- = \varepsilon_+ = +M$, avec $M \rightarrow \infty$ on se trouve dans le cas où l'on pénalise la non-réalisabilité par des coûts très élevés (méthode du big M).
- (d) Plus le nombre de morceaux est grand, plus la courbure de Θ est bien approchée, mais le nombre des paramètres du problème devient important, cela implique donc plus d'efforts pour la mise à jour de ceux-ci.

3.2 Les méthodes du centre

À la différence de la méthode de Kelley, les méthodes du centre déterminent une localisation (bornée, convexe et fermée) et calculent un point à l'intérieur de cet ensemble (voir la figure 4). Ces méthodes combinent deux techniques d'optimisation : la méthode des plans coupants et la méthode des points intérieurs.

Soit le problème (4), on définit l'ensemble de localisation

$$\mathcal{L}^k(e) = \{(\theta, \pi) : \begin{array}{l} \theta \leq e \\ \theta \geq \Theta(\pi^j) + g^j(\pi - \pi^j) \quad \forall j \in J^k \\ (\pi, \theta) \in (\mathbb{R}_+^m \times \mathbb{R}) \end{array} \}$$

comme la meilleure approximation à l'itération k du support de la fonction Θ (π^j est la solution duale avec laquelle la $j^{\text{ème}}$ colonne est calculée). Les méthodes du centre consistent alors à trouver un point central dans l'ensemble de localisation. Elles diffèrent principalement par les propriétés du centre prises en compte.

3.2.1 Méthode ACCPM (Analytic Center Cutting Plane Method)

Le premier développement de la méthode ACCPM revient à [21]. Le principe est de trouver le point suivant en calculant le centre analytique de l'ensemble de localisation reformulé comme suit :

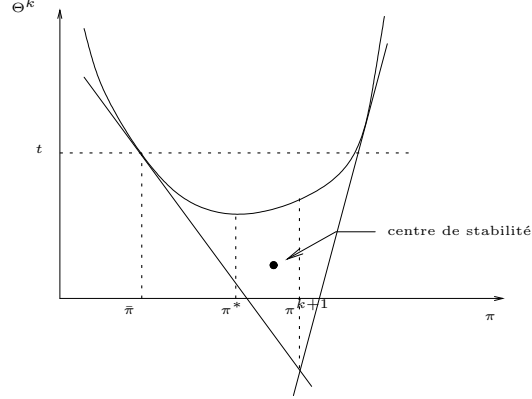


FIG. 4 – Processus des plans coupants

$$\mathcal{L}^k = \{y \quad : \quad M_j^T y \leq m_j, \quad j = 1, \dots, k\} \quad (12)$$

où $y = (\pi, \theta) \in (\mathbb{R}_+^m, \mathbb{R})$, $M_j \in \mathbb{R}^{m+1}$, $m_j \in \mathbb{R}$.

On note s_j la variable d'écart de la $j^{\text{ème}}$ contrainte de \mathcal{L}^k . Le centre analytique est l'unique solution qui optimise $\Psi = \sum_{j=1}^k s_j$ pour $y \in \mathcal{L}^k$. Notons que :

$$e^\Psi = e^{\sum_{j=1}^k s_j} = \prod_{j=1}^k s_j$$

On en déduit que le centre analytique est le point qui maximise le produit des variables d'écart de \mathcal{L}^k , ce qui se traduit géométriquement par le point qui maximise le produit des distances vers toutes les facettes du polytope \mathcal{L}^k . Notons par ailleurs que l'ajout d'une contrainte redondante au polytope \mathcal{L}^k ne change pas la géométrie du polytope, mais change le centre analytique.

3.2.2 Méthodes d'Elzinga-Moore et pc^3pa

La méthode d'Elzinga-Moore [15], trouve le prochain point comme le centre de la plus grande hypersphère contenue dans \mathcal{L}^k . Ce point est appelé le centre de Chebichev et peut être calculé en résolvant un programme linéaire de la forme suivante [38] :

$$\begin{aligned} \max \quad & \sigma \\ & \theta + \sigma \leq e \\ & \Theta(\pi^j) + g^j(\pi - \pi^j) + \sigma \sqrt{1 + \|g^j\|^2} \leq 0, \quad j \in J^k \\ & \sigma, \theta \in \mathbb{R}, \quad \pi \in \mathbb{R}_+^m \end{aligned} \quad (13)$$

où σ est le rayon de hypersphère. Cette méthode est facile à mettre en oeuvre et assure pour certains problèmes de meilleures performances que la méthode des plans coupants. Elle a pu être améliorée dans [40], dérivant la méthode pc^3pa (Proximal Chebyshev Center Cutting Plane Algorithm), cette amélioration consiste d'une

part à remédier au problème de compacité de l'ensemble de localisation, et d'autre part aux insuffisances des approximations linéaires de la fonction duale (indépendamment de la technique de calcul du point courant).

Le principe de la méthode pc^3pa est similaire à celui des faisceaux pour la stabilisation de la méthode des plans coupants, et consiste à forcer la résolution à calculer des solutions proches de la meilleure solution obtenue. Le problème (13) est donc remplacé par le problème perturbé :

$$\begin{aligned} \max \quad & \sigma - \frac{t}{2} \|\pi - \pi^k\|^2 \\ & \theta + \sigma \leq \Theta(\pi^k) \\ & \Theta(\pi^j) + g^j(\pi - \pi^j) + \sigma \sqrt{1 + \|g^j\|^2} \leq 0, \quad j \in J^k \\ & \sigma, \theta \in \mathbb{R}, \quad \pi \in \mathbb{R}_+^m \end{aligned}$$

où $t > 0$ est un paramètre de pénalisation et π^k est la solution obtenue à l'itération précédente (supposée k). Notons que l'ajout du terme quadratique rend le problème compact.

3.3 Méthode pondérée de Dantzig et Wolfe

Cette méthode a été introduite dans [50], le principe est analogue à celui décrit précédemment (à savoir, calculer des solutions duales proches de la meilleure solution duale courante), mais la procédure de résolution est différente.

À chaque itération k , la solution duale utilisée pour résoudre les sous-problèmes n'est plus π^k , mais

$$\tilde{\pi}^k = \frac{1}{s_k} \pi^k + \frac{s_k - 1}{s_k} \bar{\pi}$$

telle que $s_k = \min\{\bar{c}, \alpha\}$, $\bar{c} \in \mathbb{N}$ est une constante supérieure ou égale à 2 et $\alpha = (k + \text{le nombre d'améliorations du majorant de la solution optimale de } \Theta)/2$.

Le choix de la constante \bar{c} est un compromis entre, d'une part, éviter les solutions trop éloignées de $\bar{\pi}$ (s_k doit être petit), et d'autre part, considérer de nouvelles informations π^k (s_k doit être relativement grand).

4 Autres techniques pour accélérer la convergence de la génération de colonnes

Nous présentons dans cette section d'autres techniques, qui visent à accélérer la convergence de la génération de colonnes, se basant sur deux principes. Le premier est de tirer profit des avantages, comme la rapidité de résolution ou la qualité des solutions calculées, qu'offrent certaines méthodes d'optimisation. Le second est d'utiliser des techniques de réoptimisation qui sont adaptées au cas des processus itératifs et qui visent à diminuer le temps de résolution des problèmes à chaque itération.

4.1 La relaxation lagrangienne au sein d'un schéma de génération de colonnes

Cette méthode [8] consiste à modifier le schéma de la génération de colonnes en effectuant lors de chaque itération une résolution exacte et plusieurs résolutions approchées des sous-problèmes. Dans la résolution exacte, les sous-problèmes sont résolus par rapport à la solution duale obtenue en résolvant le problème maître avec l'algorithme du simplexe (boucle externe). Dans les résolutions approchées, les sous-problèmes sont résolus par rapport aux solutions duales obtenues en résolvant la relaxation lagrangienne du problème de départ (en relâchant les mêmes contraintes) avec l'algorithme de sous-gradient. Les étapes principales sont résumées ci-dessous :

Relaxation lagrangienne dans un schéma de génération de colonnes
<p><i>{Début}</i> initialiser le problème maître et la précision ε. <i>{début de la boucle externe}</i> Résoudre le problème maître, soit V_{PM} sa valeur optimale. Résoudre le sous-problème, soit V_{SP} sa valeur optimale. Si $V_{SP} - V_{PM} < \varepsilon$ Alors Arrêter, la solution obtenue est ε-optimale. Sinon <i>{Début de la boucle interne}</i> Générer un ensemble de solutions duales avec l'algorithme de sous-gradient appliqué au dual lagrangien du problème maître (en relâchant les contraintes liantes). Résoudre le sous-problème avec ces nouvelles variables duales et générer de nouvelles colonnes. <i>{Fin de la boucle interne}</i> Rajouter les colonnes générées dans la boucle interne au problème maître si elles n'ont pas encore été générées. <i>{Fin de la boucle externe}</i> <i>{Fin}</i></p>

Le critère d'arrêt de la boucle externe est celui de la génération de colonnes. Le critère d'arrêt de la boucle interne peut être un nombre d'itérations fixé, lié au nombre de colonnes à rajouter au problème maître lors de chaque itération. Afin de ne pas produire des colonnes déjà générées, une procédure est appliquée à la solution duale afin de générer de nouvelles solutions (pour plus de détails, voir [8]). Cette méthode permet de combiner entre la rapidité de l'algorithme de sous-gradient et l'exactitude de l'algorithme de Dantzig et Wolfe.

4.2 La relaxation agrégée au sein d'un schéma de génération de colonnes

Lors de chaque itération de la génération de colonnes, la résolution du sous-problème fournit un majorant sur la valeur optimale du problème maître (cas d'un

problème en maximisation), cette technique [30, 46] tente de générer de meilleurs majorants, donc de meilleures colonnes en utilisant la relaxation composite (lagrangienne/agrégée).

Soient (PNE) , (PM) , (DM) et (SP) les problèmes définis dans la section 2.

Le majorant de la valeur optimale du problème maître obtenu lors de chaque itération pour une solution duale π est donné par

$$M = \max_{j \in J} \{cx^j + \pi(a - A^j x^j)\}$$

En agrégeant les contraintes $Ax \leq a$ du problème (PNE) avec le multiplicateur $\sigma \in \mathbb{R}^m$, on obtient le problème suivant :

$$(PA) \quad \max cx : \begin{array}{l} \sigma Ax \leq \sigma a \\ Bx \leq b \\ x \in \mathbb{N}^n \end{array}$$

Soit $\pi^s \in R_+$ le multiplicateur lagrangien correspondant à la contrainte agrégée $\sigma Ax \leq \sigma a$. La relaxation composite de (PNE) est donnée par :

$$\max \{cx + \sigma \pi(a - Ax) : Bx \leq b, x \in \mathbb{N}^n\}$$

σ est appelé le multiplicateur composite, sa valeur optimale est obtenue en résolvant le problème dual ci-dessous en fixant la valeur de π [30] :

$$\min_{\sigma \geq 0} \Theta(\pi)_\sigma \tag{14}$$

Soit $X = \{x \in \mathbb{N}^n : Bx \leq b\}$ l'ensemble défini dans la section 2, la formulation de Dantzig et Wolfe de (PA) est donnée par (PS) , soit (DS) son dual

$$(PS) \left\{ \begin{array}{l} \max \sum_{j \in J} (cx^j) \lambda_j \\ \sum_{j \in J} \sigma(Ax^j) \lambda_j \leq \sigma a \\ \sum_{j \in J} \lambda_j = 1 \\ \lambda_j \geq 0, \quad \forall j \in J \end{array} \right. \quad (DS) \left\{ \begin{array}{l} \min \pi^s a + \pi_0^s \\ \pi_0^s \geq cx^j - \pi^s \sigma(Ax^j) \quad \forall j \in J \\ (\pi^s, \pi_0^s) \in (\mathbb{R}_+^m, \mathbb{R}) \end{array} \right.$$

Le sous-problème possède alors la forme suivante :

$$(SPS) \quad \Theta(\pi^s)_\sigma = \max_{j \in J} \{(cx^j - \pi^s \sigma Ax^j) - \pi_0^s\}$$

où π^s est le multiplicateur de la contrainte agrégée et π_0^s le multiplicateur de la contrainte de convexité. On peut écrire d'une manière équivalente :

$$\Theta(\pi^s)_\sigma \geq \{cx - \pi^s \sigma Ax\} - \pi_0^s, \forall x \in X$$

et

$$\Theta(\pi^s)_\sigma + \pi_0^s \geq \{cx - \pi^s \sigma Ax\}, \forall x \in X$$

Remarquons que $(\pi^s \sigma, \Theta(\pi^s)_\sigma + \pi_0^s)$ est une solution réalisable pour (DM) , donc

$$M^s = \pi^s \sigma a + \Theta(\pi^s)_\sigma + \pi_0^s = \max_{x \in X} \{cx^j - \pi^s \sigma(a - Ax^j)\}$$

est un majorant de la valeur de (PM) . Selon la valeur de σ , on distingue les cas suivants :

- Pour $\sigma = I = (1, 1, \dots, 1)$, $M^s = \max_{x \in X} \{cx^j - \pi^s I(a - Ax^j)\}$ est la borne lagrangienne.
- Pour $\sigma = \sigma^*$ solution de (14), M^s est la meilleure borne composite.

Il en découle que

$$\max_{x \in X} \{cx + \pi^s \sigma^*(a - Ax)\} \leq \max_{x \in X} \{cx + \pi^s(a - Ax)\}$$

lorsque $\sigma^* \neq 1$.

Les problèmes (SP) et (SPS) ne fournissent pas les mêmes colonnes. Celles fournies par (SPS) sont meilleures car elles offrent de meilleurs majorants de la valeur optimale du problème maître.

4.3 La réoptimisation

La réoptimisation consiste à étudier les possibilités de résolution efficace d'un problème (P') obtenu à partir d'un autre problème (P) en modifiant une partie de ses données. Plusieurs travaux se sont intéressés à la réoptimisation dans le cadre de diverses situations, nous en présentons quelques-uns ci-dessous.

4.3.1 Réoptimisation dans la résolution d'une suite de problèmes de plus court chemin

Plusieurs travaux se sont intéressés à la réoptimisation dans la résolution de plusieurs instances d'un problème de plus court chemin, lorsque certaines données changent. Dans [42] est proposé le premier algorithme de réoptimisation pour la résolution d'une suite de problèmes de plus court chemin lorsque le coût d'un ou plusieurs arcs diminuent. Le cas où une perturbation du coût est appliquée à un seul arc est étudié dans [36] pour les graphes orientés et dans [14] pour les graphes non orientés. Le cas général où le coût d'un ou plusieurs arcs changent est étudié dans [41].

4.3.2 Réoptimisation dans la résolution du problème du plus court chemin avec fenêtres de temps

Dans [11], les auteurs se sont intéressés à la réoptimisation dans la résolution du problème du plus court chemin avec fenêtres de temps (PCCFT) pour le calcul de chemins disjoints de coûts minimaux. Cette technique peut être intégrée dans un schéma de génération de colonnes pour la résolution du problème de tournées de véhicules avec contraintes de temps où le problème maître est de type couverture de tâches, et le sous-problème de type (PCCFT).

Le problème (PCCFT) consiste à trouver la route la moins coûteuse reliant 2 nœuds d'un réseau en visitant chaque nœud i dans un intervalle temporel $[a_i, b_i]$. Soient $G = (N, A)$ un réseau où N est l'ensemble des nœuds représentant les

tâches, s est le nœud source et p le nœud puits. À chaque arc $(i, j) \in A$ sont associés un coût c_{ij} qui inclut le coût de la tâche i plus le coût d'enchaînement des tâches i et j , et une durée positive d_{ij} qui inclut la durée de la tâche i plus la durée d'enchaînement des tâches i et j . Ce problème est formulé comme suit :

$$\left\{ \begin{array}{l} \min \sum_{(i,j) \in A} c_{ij} x_{ij} \\ \sum_{j \in N} x_{ij} - \sum_{j \in N} x_{ji} = +1, \quad i = s, \\ \sum_{j \in N} x_{ij} - \sum_{j \in N} x_{ji} = 0, \quad i \in N ; i \neq s ; i \neq p, \\ \sum_{j \in N} x_{ij} - \sum_{j \in N} x_{ji} = -1, \quad i = p \\ x_{ij} \geq 0, \quad (i, j) \in A \\ t_i + d_{ij} \leq t_j, \quad (i, j) \in A \\ a_i \leq t_i \leq b_i, \quad i \in N \end{array} \right. \quad (15)$$

où x_{ij} représente la quantité de flux acheminée sur l'arc (i, j) et t_i la date d'arrivée au nœud i .

On s'intéressera à la résolution de ce problème par la programmation dynamique. À chaque chemin réalisable X_{sj} allant du nœud s au nœud j est associée une étiquette (temps, coût) correspondant respectivement à la date d'arrivée au nœud j et au coût du chemin X_{sj} . Ces étiquettes sont notées (T_j^k, C_j^k) pour indiquer les caractéristiques du $k^{\text{ème}}$ chemin de s à j .

Soit Z^0 le sous-problème de départ et S^0 l'ensemble des nœuds du chemin optimal calculé avec l'algorithme de la programmation dynamique. Etant donné que l'objectif est de générer des chemins disjoints, tous les nœuds appartenant à S^0 sont supprimés du réseau, on cherche donc à calculer un chemin optimal dans le réseau réduit, on appellera le nouveau sous-problème Z^1 . Pour chaque nœud j , l'ensemble Q_j des labels associés aux chemins efficaces de Z^0 peuvent être partitionnés en deux sous-ensembles :

- (1) $P_j = \{(T_j^k, C_j^k) \mid \text{le } k^{\text{ème}} \text{ chemin de } s \text{ à } j \text{ } X_{sj}^k \text{ est réalisable pour } Z^1\}$.
- (2) $D_j = \{(T_j^k, C_j^k) \mid \text{le } k^{\text{ème}} \text{ chemin de } s \text{ à } j \text{ } X_{sj}^k \text{ n'est pas réalisable pour } Z^1\}$.

Sachant que les éléments de l'ensemble P_j sont des chemins efficaces pour le problème Z^0 , ils sont aussi efficaces pour Z^1 , on dispose donc d'une partie de la solution du problème Z^1 . Cette solution doit être complétée en remplaçant les éléments de D_j .

Remarques 6

- (a) Les labels de l'ensemble $Q_j = P_j \cup D_j$ définissent une fonction duale représentant un minorant des coûts des labels efficaces au nœud j du problème Z^1 en fonction de la date d'arrivée, car les labels de Q_j dominent leurs éventuels remplaçants.
- (b) D'un autre côté, les labels de l'ensemble P_j définissent une fonction primale représentant un majorant des coûts des labels efficaces au nœud j du problème Z^1 en fonction de la date d'arrivée, car les labels de P_j sont ef-

efficaces pour les deux problèmes Z^0 et Z^1 . La figure (5) montre ces deux fonctions pour un nœud j donné.

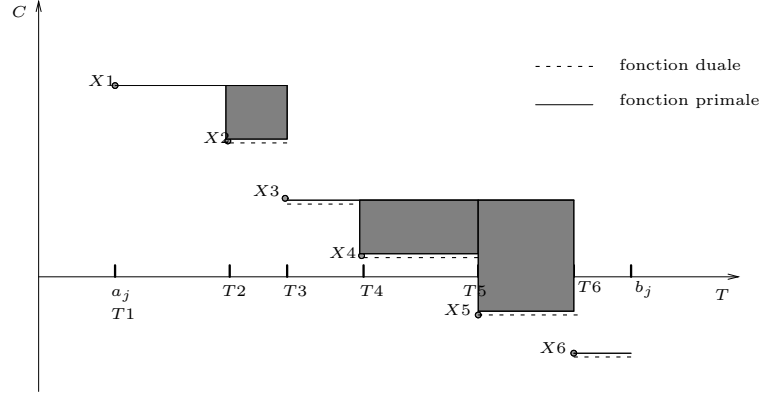


FIG. 5 – Fonctions primale, duale et zones d’incertitude

Les nouvelles solutions efficaces du problème Z^1 appartiennent aux rectangles gris, appelés les zones d’incertitude. L’objectif est donc de réduire ces zones jusqu’à les éliminer. Pour ce faire, il existe deux moyens :

- Augmenter la fonction duale en remplaçant le coût d’un label de D_j par un nouveau minorant. Ce nouveau minorant est calculé en cherchant tous les chemins X_{si} pour lesquels un prolongement par un arc $(i, j) \in A$ produit un chemin X_{sj} dont le label appartient à la zone d’incertitude traitée. L’ensemble EFF des labels efficaces est calculé pour obtenir de nouveaux labels qui définissent le nouveau minorant comme suit :

$$R_j = EFF\{(T_j, C_j) = (\max\{a_j, T_i + d_{ij}\}, C_i + c_{ij}) \mid \forall (i, j) \in A, (T_i, C_i) \in P_i \cup D_i \text{ et } (T_j, C_j) \text{ appartient à la zone d'incertitude traitée}\}$$

- Améliorer la fonction primale en calculant tous les chemins réalisables X_{si} dont le prolongement par un arc $(i, j) \in A$ produit un chemin X_{sj} dont le label appartient à la zone d’incertitude traitée. L’ensemble des solutions efficaces qui définit la nouvelle borne primale est calculé comme suit :

$$RP_j = EFF\{(T_j, C_j) = (\max\{a_j, T_i + d_{ij}\}, C_i + c_{ij}) \mid \forall (i, j) \in A, (T_i, C_i) \in P_i \text{ et } (T_j, C_j) \text{ appartient à la zone d'incertitude traitée}\}$$

Les ensembles P_j et D_j sont alors mis à jour comme suit :

$$P_j = P_j \cup (R_j \cap RP_j), \text{ car les labels appartenants à la fois à } R_j \text{ et } RP_j \text{ détruisent l'incertitude en ce point, et } D_j = (D_j - \{(T_j, C_j)\}) \cap (R_j - (R_j \cap RP_j)).$$

Cette procédure s’arrête lorsqu’il ne reste plus de zones d’incertitude, elle présente l’avantage de générer lors de chaque itération des solutions diversifiées (couvrants le maximum de tâches possible) en un temps réduit dû à la réoptimisation.

4.3.3 Réoptimisation dans la résolution du dual lagrangien du problème du sac-à-dos à deux contraintes

Les travaux [49] et [48] considèrent la réoptimisation dans le cadre de la résolution du dual lagrangien du problème du sac-à-dos à deux contraintes, résolu par l'algorithme de sous-gradient. Ce problème est donné par le programme linéaire (BK) , soit $K(\pi)$ le problème lagrangien obtenu en dualisant la première contrainte :

$$(BK) \begin{cases} \max cx \\ Ax \leq a \\ Bx \leq b \\ x_j \in \{0, 1\}, \quad j \in J \end{cases} \quad (K(\pi)) \begin{cases} \max \pi a + (c - \pi A)x \\ Bx \leq b \\ x_j \in \{0, 1\}, \quad j \in J \end{cases}$$

où $(c_j, A_j, B_j) \in \mathbb{N}^3, \forall j \in J; (a, b) \in \mathbb{N}^2$ et $\pi \in \mathbb{R}_+$ le multiplicateur lagrangien de la contrainte relâchée. La résolution de (BK) par l'algorithme de sous-gradient consiste à résoudre une suite de problèmes de sac-à-dos unidimensionnels $(K(\pi))$, dont chaque résolution est composée de plusieurs phases (ces phases sont détaillées dans [47]). La réoptimisation est intégrée dans la phase du calcul d'un majorant de la valeur optimale de $(K(\pi))$ en résolvant sa relaxation continue que l'on notera $(\bar{K}(\pi))$.

Soit $\bar{x}(\pi)$ la solution optimale de $(\bar{K}(\pi))$, on caractérise la tripartition suivante : $i(\pi) \in J, U(\pi) \subseteq J$ et $L(\pi) = J \setminus (U(\pi) \cup \{i(\pi)\})$, tels que :

$$\bar{x}_j = \begin{cases} 1 & \text{si } j \in U(\pi) \subset \{i \in J \mid \frac{c_i - \pi A_i}{B_i} \geq \frac{c_{i(\pi)} - \pi A_{i(\pi)}}{B_{i(\pi)}} \text{ et } \sum_{i \in U(\pi)} B_i \leq b\} \\ 0 & \text{si } j \in L(\pi) \subset \{i \in J \mid \frac{c_i - \pi A_i}{B_i} < \frac{c_{i(\pi)} - \pi A_{i(\pi)}}{B_{i(\pi)}}\} \\ \frac{b - \sum_{i \in U(\pi)} B_i}{B_j} & \text{tel que } j = i(\pi) \text{ et } \sum_{i \in U(\pi)} B_i + B_{i(\pi)} > b \end{cases} \quad (16)$$

Où $i(\pi)$ est l'indice de la variable fractionnaire que l'on appelle aussi l'indice critique. Cette tripartition est calculée avec l'algorithme NKR [16] qui consiste à sélectionner les objets à mettre dans le sac parmi ceux qui ont les rapports $\frac{c_j - \pi A_j}{B_j}, \forall j \in J$ les plus élevés par rapport à un élément de référence, mis à jour au cours du tri.

Deux techniques ont été élaborées dans ce contexte : la méthode MARIE (Method combining Approximation and Reoptimization for Integer programming Evaluating instances) [49] et la méthode ASIA (Adapted Step size Algorithm for a 0-1 knapsack lagrangean dual) [48].

La méthode MARIE :

Cette méthode est basée sur un algorithme de sous-gradient projeté qui améliore le sous-gradient classique et offre un cadre favorable à la réoptimisation. Rappelons que les multiplicateurs lagrangiens générés par l'algorithme de sous-gradient n'améliorent pas forcément la valeur du lagrangien lors de chaque itération.

Le schéma de sous-gradient projeté consiste en la considération d'un intervalle $[\underline{\pi}, \bar{\pi}]$ contenant un multiplicateur optimal. À chaque itération k de ce schéma, un multiplicateur π^k appartenant à cet intervalle est généré en utilisant une formule particulière pour le calcul du pas de déplacement (voir [49]). De plus, cet intervalle est dynamiquement réduit selon le signe du sous-gradient. Compte tenu de la convexité de la fonction duale, lorsque ce signe est positif, (resp. négatif), $\bar{\pi}$ est alors diminué (resp. $\underline{\pi}$ est alors augmenté).

Cette procédure permet de générer des multiplicateurs proches qui engendrent des problèmes lagrangiens voisins (dont les solutions sont probablement voisines) favorisant ainsi la réoptimisation qui consiste à utiliser la variable fractionnaire à l'itération précédente comme élément de référence à l'itération courante lors du calcul d'une tripartition optimale.

La méthode ASIA :

Cet algorithme est construit pour calculer directement (sans résolution) la valeur de $\bar{K}(\pi)$ lors de chaque itération. Le principe est de caractériser pour un multiplicateur π donné, le multiplicateur voisin $\bar{\pi}$ et la solution optimale de l'instance $\bar{K}(\bar{\pi})$. Pour cela, définissons les droites $C^j, j \in I$ qui représentent l'évolution des coefficients de l'objectif des variables $x_j, \forall j \in J$ en fonction du multiplicateur π .

Pour $\pi = \pi_0 = 0$, l'ensemble $\{C^j, j \in U(\pi_0)\}$ (resp. $\{C^j, j \in L(\pi_0)\}$) représente les droites qui se situent au-dessus (resp. en dessous) de la droite $C^{i(\pi_0)}$ (voir la figure 6). Notons $M^{i,j}$ l'abscisse du point d'intersection entre les droites C^i et C^j et soit $\pi_1 = \min_{j \in U(\pi_0) \cup L(\pi_0)} M^{i(\pi_0),j}$.

Dans l'intervalle $[\pi_0, \pi_1]$ la tripartition ne change pas, donc l'indice critique reste le même, au delà de π_1 la tripartition change, mais l'indice critique peut rester le même. L'idée est de caractériser lors de chaque itération le premier point d'intersection entre la droite correspondant à l'indice critique courant et celles correspondant aux indices restants (voir la figure 6). Cette manière de générer les multiplicateurs duaux permet de déduire la solution courante de $\bar{K}(\cdot)$ du moment où l'on dispose lors de chaque itération de la solution duale et de la variable fractionnaire.

5 Conclusion

Nous avons présenté dans ce rapport quelques méthodes d'accélération de la génération de colonnes issues de la littérature, les premières méthodes apparues sont connues sous le nom des méthodes de stabilisation. Elles visent à stabiliser le comportement des variables duales en guidant la méthode pour générer des solutions proches de la meilleure solution courante. Ces méthodes permettent de diminuer le nombre d'itérations de la génération de colonnes. Une comparaison entre ces techniques appliquées à une série de problèmes a été présentée dans [7].

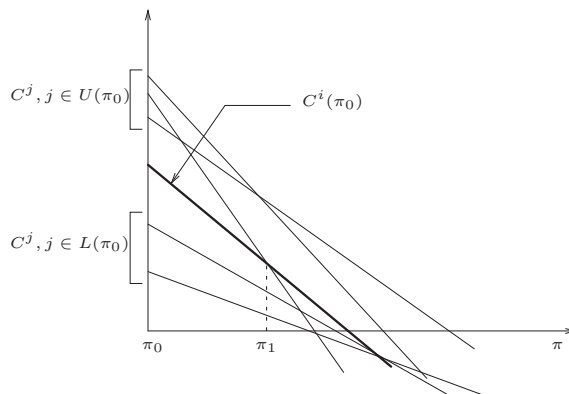


FIG. 6 – Évolution des rapports $\frac{c_i - \pi A_i}{B_i} \forall i \in J$

À côté des méthodes de stabilisation, le schéma de génération de colonnes offre plusieurs autres possibilités d'améliorations, notamment en le combinant avec d'autres méthodes d'optimisation afin de tirer profit de leurs avantages.

Nous nous sommes intéressés dans ce rapport à la possibilité d'amélioration qu'est la réoptimisation. Cette technique a été appliquée efficacement sur divers processus itératifs. Dans le cas de la génération de colonnes, le principe est d'exploiter la résolution du sous-problème lors d'une itération pour résoudre efficacement le sous-problème à l'itération suivante. Notons que la réoptimisation est plus efficace lorsque les instances résolues sont voisines (lors des dernières itérations). À la différence des méthodes de stabilisation qui visent à diminuer le nombre d'itérations, cette technique va tendre à diminuer le temps de résolution des sous-problèmes.

L'efficacité de chacune des méthodes citées dans ce rapport a été validée sur des problèmes variés. Il serait intéressant de les tester sur un ou plusieurs problèmes pour pouvoir établir une comparaison cohérente entre leurs efficacités. Cela permettrait de détecter les points forts et faibles de chaque technique et d'essayer d'y remédier en hybridant les unes avec les autres si possible.

Références

- [1] K. Aardal and S. Van Hoesel. Polyhedral techniques in combinatorial optimization. *Statistica Neerlandica*, 50 :3–26, 1996.
- [2] R. Anbil, J.J. Forrest, and W.R. Pulleyblank. Column generation and the airline crew pairing problem. *Proc. Internat. Congress of Mathematicians Berlin. Extra volume ICM Doc. Math. J. DMV*, pages 677–686, 1998.
- [3] C. Barnhart, E.L. Johnson, G.L. Nemhauser, M.W.P. Savelsbergh, and P. Vance. Branch-and-price : Column generation for solving huge integer programs. *Operations Research*, 46(3) :316–329, 1998.
- [4] H. Ben-Amor and J. Desrosiers. A proximal-like algorithm for column generation stabilization. Technical report, Les Cahiers du GERAD, Montréal, 2003.

- [5] H. Ben-Amor, J. Desrosiers, and A. Frangioni. Stabilization in column generation. Technical Report G-2004-62, Les Cahiers du Gerad, Montréal, 2004.
- [6] D.P. Bertsekas. *Non Linear Programming II*. Athena Scientific, Belmont Massachusetts, 1999.
- [7] O. Briant, C. Lemaréchal, PH. Meurdesoif, S. Michel, N. Perrot, and F. Vanderberck. Comparison of bundle and classical column generation. Technical Report RR-5453, INRIA, 2005.
- [8] M. Peeters D. Huisman, R. Jans and A.P.M. Wagelman. Combining column generation and lagrangean relaxation. Technical report, Erasmus University Rotterdam-Netherlands, EI2003-47, 2003.
- [9] G.B. Dantzig and P. Wolfe. Decomposition principle for linear programs. *Operations Research*, 8 :101–111, 1960.
- [10] G. Desaulniers, J. Desrosiers, and M. M. Solomon. *Column generation*. Springer, New York, 2005.
- [11] M. Desrochers and F. Soumis. A reoptimization algorithm for the shortest path problem with time windows. *European Journal of Operational Research*, 35 :242–254, 1988.
- [12] M. Desrochers and F. Soumis. A column generation approach to urban transit crew scheduling. *Transportation Sci.*, 23 :1–13, 1989.
- [13] G. Desrosiers, F. Soumis, and M. Desrochers. Routing with time windows by column generation. *Networks*, 14 :545–565, 1984.
- [14] R. Dionne. Etude et extension d'un algorithme de murchland. *INFOR*, 16 :132–146, 1978.
- [15] J. Elzinga and T.G. Moore. A central cutting plane algorithm for the convex programming problem. *Mathematical Programming*, 27 :134–145, 1973.
- [16] D. Fayard and G. Plateau. An algorithm for the solution of the 0-1 knapsack problem. *Computing*, 28 :269–287, 1982.
- [17] M.L. Fisher. The lagrangean relaxation method for solving integer programming problems. *Management Science*, 27 :1–18, 1981.
- [18] A. Frangioni. *Dual-Ascent Methods and Multicommodity Flow Problems*. Thesis, University of Pisa-Genova-Udine, 1997.
- [19] A.M. Geoffrion. Lagrangean relaxation for integer programming. *Mathematical Programming Stud.*, 2 :82–114, 1974.
- [20] P.C. Gilmore and R.E. Gomory. A linear programming approach to the cutting stock problem. *Operations Research*, 9 :849–859, 1961.
- [21] J.L. Goffin and J.P. Vial. Cutting planes and column generation techniques with the projective algorithm. *Journal of optimization theory and applications*, 65 :409–429, 1990.
- [22] M. Guignard. *Lagrangean relaxation*. In M.Resende and P.Pardalos, Oxford university press, 2004.
- [23] J.E. Kelley. The cutting-plane method for solving convex programs. *SIAM Journal on Optimization*, 8 :703–712, 1960.

- [24] S. Kim, K.N Chang, and J.Y .Lee. A descent method with linear programming subproblems for nondifferentiable convex optimization. *Mathematical programming*, 71 :17–28, 1995.
- [25] L.S. Lasdon. *Optimization theory for large systems*. MacMillan series in operations research, 1972.
- [26] C. Lemaréchal. An algorithm for minimizing convex functions. *Information processing*, 74 :552–556, 1974.
- [27] C. Lemaréchal, A. Nemirovskii, and Y. Nesterov. New variants of bundle methods. *Mathematical programming*, 69 :111–149, 1995.
- [28] I. Loiseau, J.M. Brito, N. Maculan, and M. Passini. Column generation in integer linear programming. *RAIRO-Operations Research*, 37 :67–83, 2003.
- [29] I. Loiseau, J.M. Brito, N. Maculan, and M. Passini. *Génération de colonnes en programmation linéaire en nombres entiers, Optimisation Combinatoire 1 - Concepts fondamentaux*. Hermes Science Publications, 237-262, 2005.
- [30] L.A.N. Lorena, M.A. Pereira, and S.N.A. Salomão. The lagrangean/surrogate relaxation and the column generation : New bounds and new columns. Technical report, Instituto Nacional de Pesquisas Espaciais - INPE, 2001.
- [31] M E. Lubecke and J. Desrosiers. Selected topics in column generation. Technical Report G-2002-64, Les cahiers du GERAD, 2002.
- [32] R.E. Marsten, W.W. Hogan, and J.W. Blankenship. The boxstep method for large-scale optimization. *Operations Research*, 3 :389–405, 1975.
- [33] A. Mehrotra and M.A. Trick. A column generation approach for graph coloring. *INFORMS J. Comput.*, 8(4) :344–354, 1996.
- [34] O. Du Merle, D. Villeneuve, J. Desrosiers, and P. Hansen. Stabilisation dans le cadre de la génération de colonnes. Technical report, Les cahiers du GERAD, 1997.
- [35] O. Du Merle, D. Villeneuve, J. Desrosiers, and P. Hansen. Stabilized column generation. *Discrete Mathematics*, 194 :229–237, 1999.
- [36] J.D. Murchland. The effect of increasing or decreasing the length of a single arc all shortest distances in a graph. Technical report, Report LBS-TNT-26, London graduate school of buisness studies, 1967.
- [37] A. Nagih and F. Soumis. Nodal aggregation of resource constraints in a shortest path problem. *European Journal of Operational Research*, 172 :500–514, 2006.
- [38] G.L. Nemhauser and W.B. Widhelm. A modified linear program for columnar methods in mathematical programming. *Operations Research*, 19 :1051–1060, 1971.
- [39] G.L. Nemhauser and L.A. Wolsey. Integer and combinatorial optimization. *Wiley, New York*, 1988.
- [40] A. Ouorou. A proximal cutting plane method using Chebychev center for nonsmooth convex optimization. *Technical report*, 2006.
- [41] U. Pape. Changes in networks and ajustement of the length of shortest paths. *Computing*, 12 :357–362, 1974.

- [42] V. Radionov. The parametric problem of shortest distances. *Urdmurt U.S.S.R. Comuta. Math. Math Phys*, 8 :336–342, 1968.
- [43] T.K. Ralphs and M.V. Galati. Decomposition in integer linear programming. Technical Report 04T-019, Lehigh University Industrial and Systems Engineering Department, 2005.
- [44] R. T. Rockafellar. Monotone operators and the proximal point algorithm. *SIAM journal on control and optimization*, 14 :877–898, 1976.
- [45] H. Schramm and J. Zowe. A version of the bundle idea for minimizing a nonsmooth function : Conceptual idea, convergence analysis, numerical results. *SIAM journal on optimization*, 2 :121–152, 1992.
- [46] E.L.F. Senne and L.A.N. Lorena. Stabilizing column generation using lagrangean/surrogate relaxation : an application to p-median location problems. *In Proceedings of the EURO2001 Conference, Erasmus University, Rotterdam*, July 2001.
- [47] B. Thiongane. Réoptimisation dans le dual lagrangien du biknapsack en variables 0-1. *Thesis, University Paris 13*, 2004.
- [48] B. Thiongane, A. Nagih, and G. Plateau. Adapted step size algorithm for a 0-1 biknapsack lagrangean dual. *Annals of Operations Research*, 139(1) :253–373, 2004.
- [49] B. Thiongane, A. Nagih, and G. Plateau. Lagrangean heuristics combined with reoptimization for the biknapsack problem. *Discrete Applied Mathematics*, 154(15) :2200–2211, 2006.
- [50] P. Wenges. Weighted dantzig wolfe decomposition for linear mixed-integer programming. *Int.Trans.Op Res*, 4 :151–162, 1997.
- [51] P. Wolfe. A method of conjugate subgradients for minimizing nondifferentiable convex functions. *Math. Programming Stud.*, 3 :145–173, 1975.