



**HAL**  
open science

# Cônes de matrices et programmation mathématique : quelques applications

Alexandre Laugier

► **To cite this version:**

Alexandre Laugier. Cônes de matrices et programmation mathématique : quelques applications. Autre [cs.OH]. Université Nice Sophia Antipolis, 2002. Français. NNT : . tel-00505441

**HAL Id: tel-00505441**

**<https://theses.hal.science/tel-00505441>**

Submitted on 23 Jul 2010

**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.

École Doctorale STIC  
Département d'Informatique

# THÈSE

Présentée pour obtenir le titre de

**Docteur ès SCIENCES**  
de l'Université de Nice-Sophia Antipolis

Spécialité: **INFORMATIQUE**

par

**Alexandre Laugier**

## **Cônes de matrices et Programmation mathématique : Quelques applications**

Soutenue publiquement le 26 mars 2002 devant le jury composé de:

Fabrice	André	Examineur
Jean-Claude	Bermond	Directeur
Pierre	Bernhard	Président
Michel	Burlet	Co-directeur
Jean	Fonlupt	Rapporteur
Claude	Lemaréchal	Examineur
Philippe	Mahey	Rapporteur

ESSI, Amphi Ouest à 15 heures

*A mes parents, mon frère et Nathalie*

Avant d'adresser des remerciements individuels à quelques personnes envers qui je suis plus particulièrement redevable, je tiens à exprimer ma gratitude à mon environnement professionnel et surtout à celui que j'ai trouvé lors de mon arrivée dans le département ORE. Merci donc à Sophie Maucorps, Jacques Chambon, Erik Chrisment et d'autres pour leur accueil plein de bonne humeur.

Cet accueil au sein du département ORE, et donc le déroulement des onze années qui nous en séparent, n'aurait pas été possible sans Fabrice André. Je tiens à le remercier de la confiance dont il m'a témoigné. Son rire retentissant dans notre bâtiment ou au téléphone m'a encouragé au cours de ces années. J'espère que sa présence dans ce jury n'est qu'une rencontre supplémentaire et surtout pas l'ultime.

Je n'oublie pas l'aide essentielle que Michel Burlet m'a apportée au cours de ces années, récupérant les cours de DEA auprès des étudiants, m'expliquant les concepts fondamentaux de l'optimisation combinatoire durant un repas au restaurant. Je dois à Michel l'intérêt que je porte à l'étude des méthodes polyédrales pour la résolution des problèmes d'optimisation combinatoire. Je peux dire que sans lui ce mémoire ne serait pas ce qu'il est, et peut-être qu'il ne serait pas.

J'associe à ces remerciements Philippe Mahey car j'apprécie la manière dont il m'a encouragé depuis longtemps à poursuivre mes travaux. Je le remercie de la lecture avisée qu'il a faite de ce mémoire, contribuant à son amélioration.

Jean Fonlupt fut mon professeur durant le DEA, j'apprécie sa connaissance du domaine et sa vision profonde, aussi je suis particulièrement heureux qu'il est accepté de rapporter sur ce travail, ses commentaires me serviront sûrement à développer et à donner de l'étoffe à mes travaux.

J'apprécie chez Jean-Claude Bermond son dynamisme communicatif et l'enthousiasme qu'il met à faire toute chose. J'ai aussi très apprécié le soutien qu'il nous a apporté lors d'une période professionnellement difficile que nous eûmes à traverser. A son contact j'ai pu largement améliorer mes connaissances en théorie des graphes.

J'ai rencontré Claude Lemaréchal pour la première fois à Autrans en janvier 99, j'ai été étonné de sa disponibilité. Je suis flatté qu'il ait accepté de faire partie de ce jury.

Je remercie Pierre Bernhard pour le soutien actif et enthousiaste qu'il apporta à un projet que nous eûmes, je le remercie aussi d'avoir accepté de présider ce jury.

Je remercie Michel Goemans pour ses judicieuses observations qui m'ont permis d'éclaircir ma rédaction. En particulier les chapitres 4 et 6 lui doivent beaucoup, il y a apporté la rigueur qui faisait défaut.

Je suis très content de travailler dans l'équipe de Patrick Brown parce qu'il a su créer au

sein de celle-ci une ambiance détendue. Je le remercie de m'avoir permis de développer mes travaux et de les avoir défendu au sein de FTR&D.

Je tiens à remercier mes deux collègues de bureau Sébastien Bertrand et Jérôme Galtier pour la bonne ambiance qui règne entre nous. Je les remercie également d'avoir participé à mon travail. En effet c'est avec Sébastien que nous réalisons nos premiers travaux relatifs à lift-and-project, les travaux sur les flot non-conservatifs. De plus il m'a beaucoup aidé pour résoudre de nombreux problèmes informatiques. Quant à Jérôme sa vision des problèmes et des mathématiques m'a beaucoup aidé à développer mes travaux sur la programmation semi-définie positive, il fut le premier à relire ce mémoire, sa tâche fut rude.

Je remercie Aubin Jarry de m'avoir aidé à résoudre le problème sur les graphes 2-connexes de diamètre donné sur lequel je séchais depuis plusieurs années. J'ai particulièrement apprécié son affabilité. Je pense que nous continuerons à travailler ensemble.

## Résumé

Au cours de ce mémoire nous présentons nos travaux concernant la programmation linéaire en nombres entiers. Ces travaux sont inspirés de ceux développés par L. Lovász et A. Schrijver en 1991 dans [L. Lovász and A. Schrijver : Cones of matrices, set functions and 0-1 optimization, SIAM 1991]. Nous reprenons leurs travaux en les détaillant de sorte à les rendre plus facilement abordables. Nous mettons ainsi clairement en valeur les connexions qui existent entre la programmation linéaire en variables bivalentes et la programmation semi-définie positive. Nous dérivons de la construction de Lovász et Schrijver un algorithme de coupes polyédrales pour résoudre les problèmes de programmation en nombres entiers.

Dans un premier temps nous présentons les applications les plus importantes de la programmation semi-définie positive à la résolution de problèmes d'optimisation combinatoire : le travail de Lovász concernant le problème de stable de poids maximum, [L. Lovász : On the Shannon capacity of a graph, IEEE Transaction on communication 1979], et celui de M. Goemans et D. Williamson relatif au problème de la coupe de poids maximum, [M.X. Goemans and D.P. Williamson : Improved approximations algorithms for maximum cut satisfiability using semi-definite programming] . Puis nous présentons une application de la programmation semi-définie positive au problème du flot de coût minimum avec contrainte de longueur sur les chemins support de flot.

Enfin nous nous intéressons au plongement de treillis dans des cônes convexes en recherchant les bases de Hilbert de ces cônes et plus précisément des cônes définis par Lovász et Schrijver. Nous montrons notamment que ces deux cônes partagent une unique base de Hilbert après avoir vu que les matrices qu'ils contiennent peuvent s'obtenir comme mineurs des matrices semi-définies positives représentant les fonctions fortement décroissantes sur les treillis.

En guise de conclusion nous exposons un travail sur deux cônes de métriques, travail dans lequel nous définissons une relaxation semi-définie positive du problème du graphe partiel  $k$ -arête connexe de poids minimum.

En annexe nous présentons un travail effectué avec A. Jarry dans lequel une conjecture de Bollobas sur le nombre minimum d'arêtes d'un graphe 2-connexe et de diamètre  $d$  est démontrée. Nous présentons aussi un travail mené avec S. Bertrand et P. Mahey relatif au problème de routage de flux non-conservatifs.

---

## CONES OF MATRICES AND MATHEMATICAL PROGRAMMING : SOME APPLICATIONS

### Abstract

All along this dissertation we present our works related to the scope of integer linear programming. This work come from those done by L. Lovász and A. Schrijver in [L. Lovász and A. Schrijver : Cones of matrices, set functions and 0-1 optimization, SIAM 1991]. First we present extensively their work in order to make it more accessible. Thus we show clearly the relations between integer programming and positive semi-definite programming. Then we derive from the Lovász and Schrijver's construction a cutting plane algorithm solving linear integer programs.

Second we present the most famous applications of positive semi-definite devoted to combinatorial optimisation problems : the work of L. Lovász related to maximum independent set and those of M. Goemans and D. Williamson to the maximum cut. Then we explain an application to the minimum cost flow subjected to end to end delay constraint.

Then we look for embeddings of lattices in convex cones and more precisely in the cones defined by L. Lovász and A. Schrijver. We show that the two cones handled in the first part of the dissertation share the same unique Hilbert basis.

At last we show that the latter cones can be viewed as subsets of a cone of metrics, as an application we modelise the problem of the 2-connected subgraph on this cone of metrics and define a positive semi-definite relaxation of this problem. In the annexes we explain a work done together with A. Jarry related to the minimum number of edges of a 2-connected graph with a diameter constraint. A work done together with S. Bertrand and P. Mahey is also explained, it concerns the minimum cost flow in the special behaviour of non conservative flows.

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>5</b>
<b>2</b>	<b>Principaux concepts de la programmation linéaire en variables bivariées</b>	<b>8</b>
2.1	Programmation linéaire . . . . .	9
2.1.1	Inégalités linéaires . . . . .	9
2.1.2	Polarité . . . . .	12
2.1.3	Dualité . . . . .	13
2.1.4	Séparation et optimisation . . . . .	14
2.2	Programmation mathématique en nombres entiers . . . . .	19
2.2.1	Autres formes de polarité : Bloquants et Antibloquants . . . . .	20
2.2.2	Enveloppe convexe de vecteurs entiers . . . . .	24
2.2.3	Treillis et corps convexes . . . . .	28
<b>3</b>	<b>Programmation semi-définie positive</b>	<b>33</b>
3.1	Le cône des matrices semi-définies positives . . . . .	33
3.2	Programme semi-défini positif . . . . .	35
3.3	Programme semi-défini positif et variables 0-1 . . . . .	37
3.4	Le lemme de Farkas étendu . . . . .	39
3.5	La dualité en programmation semi-définie positive . . . . .	40
3.6	Conditions d'écart complémentaires . . . . .	43
3.7	Dualité étendue en programmation semi-définie positive . . . . .	43
3.8	Programmation mathématique et formes hermitiennes . . . . .	45
<b>4</b>	<b>Applications de la programmation semi-définie positive</b>	<b>48</b>
4.1	Le problème du stable de cardinalité maximum . . . . .	49
4.2	Le problème de la coupe de poids maximum . . . . .	50
4.2.1	L'algorithme de Goemans et Williamson . . . . .	51
4.3	Le problème de la $k$ -coupe de poids maximum . . . . .	54
4.4	Formes hermitiennes et problèmes combinatoires . . . . .	55
4.5	Flot et délai . . . . .	57
4.5.1	Flot de délai moyen minimum . . . . .	57
4.5.2	Flot de coût minimum et délai borné . . . . .	58
<b>5</b>	<b>Lift-and-project</b>	<b>68</b>
5.1	La construction de Lovász et Schrijver . . . . .	68
5.1.1	Détermination d'inégalités valides . . . . .	72



---

5.2	Un lift-and-project séquentiel . . . . .	76
5.2.1	Détermination d'inégalités valides . . . . .	79
<b>6</b>	<b>Applications de lift-and-project</b>	<b>82</b>
6.1	Le problème du stable de cardinalité maximum . . . . .	82
6.2	Un algorithme de coupes polyédrales . . . . .	85
6.2.1	Un algorithme de coupes polyédrales basé sur la construction de Lovász et Schrijver . . . . .	85
6.2.2	Ensembles de normalisation . . . . .	91
6.2.3	Utilisation dans un algorithme de type Branch-and-cut . . . . .	92
<b>7</b>	<b>Optimisation dans un treillis</b>	<b>94</b>
7.1	Fonctions fortement décroissantes . . . . .	94
7.2	Projection sur l'espace des singletons et du sous-ensemble vide . . . . .	97
7.3	Totale unimodularité de la matrice $\mathcal{Z}$ . . . . .	99
7.4	Bases de Hilbert des cônes $M(K)$ et $M^+(K)$ . . . . .	100
<b>8</b>	<b>Cônes de matrices et cônes de métriques</b>	<b>104</b>
8.1	Un exemple d'utilisation des cônes $Cut_n$ et $Neg_n$ . . . . .	108
<b>9</b>	<b>Conclusion</b>	<b>113</b>

# Chapitre 1

## Introduction

L'environnement concurrentiel dans lequel s'exercent les activités industrielles a mis en exergue les processus de décisions dont la complexité s'est considérablement accrue au cours des deux dernières décennies. Cette complexité a justifié le développement d'outils d'aide à la décision. Parmi l'ensemble des concepts qui ont permis l'élaboration des outils d'aide à la décision la programmation mathématique en variables binaires s'est avérée être le plus performant. La performance s'entend ici en terme de qualité des solutions fournies et richesse des informations fournies ou prises en compte. La très grande difficulté de résolution des programmes mathématiques, et linéaires en particulier, en variables bivalentes a conduit à la mise au point de méthodes spécialisées pour chaque type de problèmes. Une manière de faire est de rechercher des contraintes spécifiques pour un problème, cette approche s'est fortement développée pour certain problème et tout particulièrement pour le problème du voyageur de commerce. Il nous semble que pendant un temps assez long, disons du milieu des années soixante au début des années quatre vingt dix, la communauté scientifique a renoncé à mettre au point des méthodes permettant de traiter le cas de la programmation linéaire en variables binaires dans toute sa généralité. Si on peut se douter que de telles méthodes ne seront jamais aussi efficaces que des méthodes spécialisées pour résoudre un type de problèmes donné, il demeure qu'elles présentent un intérêt pour la vie économique. En effet la mouvance, voire la volatilité, de l'environnement fait que les contraintes d'un problème changent très rapidement, certaines disparaissent tandis que d'autres surgissent. De ce fait la structure sur laquelle le problème est défini se trouve radicalement transformée, rendant inopérante une méthode spécialisée. C'est parce que nous croyons à l'efficience de la programmation linéaire en nombres entiers que nous avons choisi de nous y intéresser dans notre travail. Ce travail nous conduira de l'approche polyédrale à l'utilisation de cônes de matrices puis de cônes de métriques.

Dans le premier chapitre nous exposons les concepts fondamentaux de la programmation linéaire : en particulier les concepts de séparation et de dualité. Nous y exposons donc les résultats fondamentaux dus à Farkas, Minkowsky, Weyl, Voronoï et Kuhn. Nous présentons le théorème Sep=Opt dû à Grötschel, Lovász et Schrijver. Nous abordons aussi des notions très importantes telles que la notion de polyèdres bloquants, la notion de système TDI et celle qui lui est liée de base de Hilbert. La fin de chapitre est consacrée à l'exposé des principales méthodes de résolution de programmes linéaires en variables binaires : la méthode de coupes de Gomory-Chvátal, la méthode de convexification d'une disjonction de Balas et l'algorithme de réduction de base de Lovász.

Dans le premier chapitre nous avons délibérément occulté une approche particulière des

problèmes d'optimisation combinatoires par la programmation semi-définie positive. Cette approche constitue une part importante de notre travail. Aussi pour que le lecteur trouve toute l'information nécessaire nous rappelons, dans ce deuxième chapitre, les propriétés essentielles des matrices semi-définies positives. Ensuite nous introduisons la notion de programme semi-défini positif, puis à l'instar de ce qui a été fait pour la programmation linéaire nous introduisons le concept de séparation en exposant une version étendue à la programmation semi-définie positive du lemme de Farkas et une théorie de dualité pour la programmation semi-définie positive. Nous mentionnons une théorie exacte de dualité et une extension naturelle de la programmation semi-définie positive aux formes sesquilinéaires. Enfin nous montrons comment la programmation linéaire en variables binaires conduit naturellement à la définition de programmes semi-définis positifs; nous donnons la définition des cônes  $M(K)$  et  $M^+(K)$  dont nous exploitons les propriétés plus loin.

Le troisième chapitre est dédié à la présentation d'applications de la programmation semi-définie positive. Dans un premier temps nous exposons les travaux de Lovász sur le stable de cardinalité maximum, puis ceux de Goemans et Williamson relatifs à la coupe de poids maximum et à la 3-coupe de poids maximum, ce dernier problème étant modélisé par des formes hermitiennes. Le reste de ce chapitre est consacré à l'exposé de travaux que nous avons menés avec J. Galtier. Ces travaux consistent en la modélisation d'une fonction de congestion dans les réseaux par des matrices semi-définies positives. Nous nous intéressons au problème de la minimisation du délai moyen et au problème du flot de coût minimum avec contrainte de délai. Cette approche originale permet d'obvier la difficulté liée à la non différentiabilité des fonctions de congestion lorsque la valeur du flot supporté par un arc atteint la capacité de ce dernier. Pour résoudre ce problème nous avons mis au point une technique de génération de colonnes sur les cônes des matrices semi-définies positives. Le chapitre quatre est consacré à l'exposé de deux méthodes de type lift-and-project permettant de déterminer l'enveloppe convexe de points à coordonnées entières satisfaisant des contraintes linéaires. La première est due à Lovász et Schrijver et utilise les cônes  $M(K)$  et  $M^+(K)$ , tandis que la seconde due à Balas, Ceria et Cornuéjols se fonde sur les travaux de Balas relatifs à la convexification d'une disjonction de polyèdres.

Dans le cinquième chapitre nous présentons l'application au stable de poids maximum faite par Lovász et Schrijver de leur méthode de lift-and-project. Puis, à l'instar de Balas, Ceria et Cornuéjols pour la programmation disjonctive, nous exposons un algorithme de coupes polyédrales que nous avons dérivé de la construction de Lovász et Schrijver pour la résolution des programmes linéaires en variables bivalentes. Cet algorithme se distingue de celui conçu par Balas, Ceria et Cornuéjols par le fait qu'il manipule l'ensemble des variables simultanément. Nous montrons la finitude et l'exactitude de notre algorithme. Puis en utilisant des propriétés de l'intersection de l'opérateur  $N(K)$  et des faces de l'hypercube, propriétés que nous avons montrées au chapitre précédent, nous montrons comment les coupes générées peuvent être manipulées dans le cadre d'un algorithme de type branch-and-cut.

Au cours du sixième chapitre nous exposons une autre manière de construire les cônes  $M(K)$  et  $M^+(K)$ . Cette méthode, toujours due à Lovász et Schrijver, fait intervenir les fonctions d'ensembles fortement décroissantes et la matrice de Möbius d'un treillis binaire. Cette approche permet de mettre clairement en valeur le rôle de l'ensemble vide et des singletons dans les problèmes combinatoires. Après avoir produit une preuve originale du lemme de Gordan nous pouvons démontrer que les cônes  $M(K)$  et  $M^+(K)$  possèdent la même et unique base de Hilbert.

---

Durant un septième et ultime chapitre nous abordons les problèmes d'optimisation combinatoire via la programmation sur des cônes de métriques, typiquement  $Cut_n$  et  $Neg_n$  qui sont respectivement isomorphes au cône des métriques  $l_1$  et à celui des métriques  $l_2$ . Nous montrons les relations qui existent entre ces cônes et les cônes de matrices que nous avons utilisés précédemment, en particulier nous montrons que  $M(K) \subseteq Cor_n$  qui est isomorphe à  $Cut_n$ . Puis nous modélisons le problème du graphe partiel  $k$ -arête-connexe sur le cône  $Cut_n$  et en définissons une relaxation semi-définie positive par plongement dans  $Neg_n$ . Enfin le lecteur trouvera en annexe des travaux qui ne sont pas en relations immédiates avec le corps de cette thèse mais par contre en relation plus immédiate avec notre activité d'ingénieur à France Télécom. La première contribution, réalisée avec A. Jarry, nous a permis de démontrer une conjecture de Bollobas sur le nombre minimum d'arêtes d'un graphe 2-connexe et dont le diamètre est borné. Ce travail trouve une application dans la conception de réseaux sécurisés garantissant un délai d'acheminement borné. La deuxième contribution est tout aussi importante et originale : il s'agit de calculer des routages optimaux en tenant compte des empilages de protocoles. C'est-à-dire que le flot qui circule sur un arc est un multiple de la fraction de la demande qui est routée sur un chemin empruntant cet arc mais de plus le multiplicateur est une fonction du chemin emprunté pour parvenir à l'arc. Une conséquence est que les flots portés par deux chemins, joignant deux mêmes sommets et partageant un arc  $e$ , n'auront pas a priori le même multiplicateur sur cet arc. Ce travail a été mené en collaboration avec S. Bertrand et P. Mahey, il définit un nouveau type de problèmes de réseaux. ce type de problèmes a une grande importance pour les opérateurs de télécommunications historiques car ils sont confrontés à l'hétérogénéité de leurs réseaux.

---

## Chapitre 2

# Principaux concepts de la programmation linéaire en variables bivalentes

Les problèmes d'optimisation sont étudiés depuis plus de deux siècles si on considère que les premiers travaux formalisés sont ceux de G. Monge, [70]. On peut remarquer que la recherche relative aux problèmes d'optimisation s'est développée depuis une soixantaine d'années tant dans un contexte d'économie libérale que dans un contexte d'économie planifiée. Au cours des années quarante et cinquante l'optimisation a connu son essor sous l'impulsion des militaires. C'est ainsi qu'apparut le nom de **Operations Research** c'est à dire la recherche concernant les opérations de guerre. G.B. Dantzig, J. von Neuman et H.W. Kuhn firent nombre de leurs publications dans le cadre de travaux réalisés pour les forces navales américaines. Les efforts de recherche sur le domaine entrepris dans le camp opposé ne sont pas moindres, L.V. Kantorovich qui participa aux études pour la planification de l'économie soviétique, décrit dans [49] les buts recherchés par les deux types d'économie et l'adéquation de la programmation linéaire pour certains problèmes d'organisation et de planification de la production. Puis vinrent N.Z. Shor, A. Karzanov, L. Kachiyan. N'oublions pas la célèbre école hongroise avec P. Erdős et plus tard L. Lovász qui jouera un rôle essentiel dans le développement des mathématiques discrètes.

Toutefois la compétition va se déplacer du terrain militaire vers le terrain économique vers le milieu des années soixante-dix et surtout au cours des années quatre-vingt. Les applications seront alors d'une manière prépondérante industrielles. Deux faits vont promouvoir l'optimisation auprès des industriels :

- l'accentuation de la compétition économique par l'ouverture de zones de libre échange (Traité de Rome en 1956, accord du Gatt) faisant rechercher aux entreprises tout avantage concurrentiel ;
- l'apparition d'ordinateurs puissants fiables et peu onéreux qui permettent la résolution efficace des problèmes.

Que ce soit dans le cadre d'affrontements armés ou de compétition économique la question posée est : que doit-on faire pour satisfaire au mieux un critère sachant qu'un objectif doit être atteint? Le critère peut être représenté par une fonction de plusieurs variables,  $n$ , et l'objectif précisé par des contraintes définissant un domaine auquel les variables instanciées

doivent appartenir. Ainsi pouvons-nous définir la programmation mathématique comme l'étude des méthodes de résolution des problèmes de la forme :

$$\begin{aligned} \text{Min} \quad & f(x) \\ & x \in \mathcal{D} \subseteq \mathbb{R}^n \end{aligned}$$

## 2.1 Programmation linéaire

Une des formes de programmation mathématique les plus étudiées est la programmation linéaire qui correspond au cas où  $f(x)$  est une fonction linéaire de  $n$  variables et où le domaine  $\mathcal{D}$  est déterminé par un ensemble fini d'inégalités linéaires.

### 2.1.1 Inégalités linéaires

Avant de déterminer l'optimum d'une fonction objective sur une région de l'espace il faut répondre à la question : est-ce-que cette région est non vide? Se pose alors la question de la description de la région de l'espace. Les premiers domaines qui furent envisagés sont les polyèdres convexes. On doit à J. Farkas,[28] et [29], et H. Minkowski [69] et H. Weyl,[89] la caractérisation de telles régions en termes d'inégalités linéaires.

**Définition 2.1** *Un cône  $C$  est dit polyédral s'il peut être décrit par un ensemble fini d'inégalités linéaires, c'est-à-dire s'il existe une matrice  $A$  telle que  $C = \{x \mid Ax \geq 0\}$ .*

**Théorème 2.1 (Farkas, Minkowsky, Weyl)**  *$C$  est cône polyédral si et seulement si il existe une famille finie de vecteurs  $\{x_1^*, \dots, x_p^*\}$  tels que  $\forall x \in C, x = \sum_{i=1}^p \lambda_i x_i^*, \lambda_i \geq 0 \forall i \in \{1, \dots, p\}$ .*

Ce théorème peut être vu comme une conséquence d'un théorème appelé généralement théorème fondamental des inégalités linéaires dû à J. Farkas et H. Minkowsky puis repris par H. Weyl et C. Carathéodory, [16].

**Théorème 2.2 (Farkas, Minkowsky, Carathéodory, Weyl) Théorème fondamental des inégalités linéaires**

*Soit  $a_1, \dots, a_m, b$  des vecteurs d'un espace de dimension  $n$ , alors :*

1. *soit  $b$  peut s'écrire comme combinaison linéaire à coefficients positifs ou nuls de vecteurs linéairement indépendants pris parmi  $\{a_1, \dots, a_m\}$  ;*
2. *soit il existe un hyperplan  $H = \{x \mid c^t x = 0\}$  contenant  $t - 1$  vecteurs linéairement indépendants pris parmi  $\{a_1, \dots, a_m\}$  tel que  $c^t b < 0$  et  $c^t a_i \geq 0, \forall i \in \{1, \dots, m\}$  avec  $t = \text{rang}\{a_1, \dots, a_m, b\}$*

La preuve de ce théorème, proche de celle exposée dans [79], va nous permettre d'introduire des notions fondamentales.

Preuve :

Au cours de cette preuve nous supposons que l'espace de dimension  $n$  est  $\mathbb{R}^n$  et que la famille de vecteurs  $\{a_1, \dots, a_m\}$  en est une famille génératrice. Dans un premier temps montrons que les deux propositions du théorème s'excluent mutuellement. Supposons qu'il

existe  $\lambda_1, \dots, \lambda_m$  tous positifs ou nuls tels que  $b = \sum_{i=1}^m \lambda_i a_i$  et supposons que les deux propositions puissent être vérifiées simultanément, alors :

$$0 > c^t b = c^t \sum_{i=1}^m \lambda_i a_i = \sum_{i=1}^m \lambda_i c^t a_i \geq 0$$

ce qui est contradictoire.

Montrons maintenant qu'au moins une des deux propositions est vraie. Choisissons  $n$  vecteurs linéairement indépendants parmi ceux de la famille  $\{a_1, \dots, a_m\}$ , soit  $B = \{a_{i_1}, \dots, a_{i_n}\}$  ces vecteurs. Clairement  $B$  est une base de  $\mathbb{R}^n$  aussi pouvons-nous dire qu'il existe

$\lambda_{i_1}, \dots, \lambda_{i_n}$  tels que  $b = \sum_{j=1}^n \lambda_{i_j} a_{i_j}$ . Si  $\lambda_{i_j} \geq 0 \forall j \in \{1, \dots, n\}$  alors la proposition 1 est vérifiée. La deuxième proposition de l'alternative consiste à dire qu'il existe au moins un indice  $j \in \{1, \dots, n\}$  tel  $\lambda_{i_j} < 0$ , soit  $h$  le plus petit indice tel que  $\lambda_h < 0$ . Posons  $c = \frac{1}{a_h^t a_h} a_h - \sum_{j=1}^n \frac{a_h^t a_{i_j}}{a_h^t a_h a_{i_j}^t a_{i_j}} a_{i_j}$ , on peut alors remarquer que  $H = \{x | c^t x = 0\}$  est

l'hyperplan engendré par  $B - \{a_h\}$ . Aussi nous obtenons  $c^t b = c^t \sum_{j=1}^n \lambda_{i_j} a_{i_j} = \lambda_h < 0$ .

Si  $c^t a_i \geq 0 \forall i \in \{1, \dots, m\}$  alors la proposition 2 est vérifiée, sinon remplaçons  $B$  par  $(B - \{a_h\}) \cup \{a_s\}$  où  $s$  est le plus petit indice tel que  $c^t a_s < 0$ , puis recommencer le traitement décrit dans ce paragraphe.

Afin d'achever la preuve du théorème nous devons montrer que le traitement décrit dans le paragraphe précédent comporte un nombre fini d'étapes. Notons  $B^k$  la base de  $\mathbb{R}^n$  obtenu après  $k$  itérations; si le traitement ne se termine pas alors comme le nombre de bases issues de  $\{a_1, \dots, a_m\}$  est fini, on peut affirmer qu'il existe  $k', k < k'$  tel que  $B^k = B^{k'}$ . Posons  $r$  comme le plus grand indice tel que  $a_r$  ait été retiré au cours d'une des itérations  $k, k+1, \dots, k'-1$ , disons que cet évènement est survenu à l'itération  $p$ . Etant donné que  $B^k = B^{k'}$  on peut dire qu'il existe une itération  $q$  parmi  $k, k+1, \dots, k'-1$  au cours de laquelle  $a_r$  est rentré en base. On a donc  $B^p \cap \{a_{r+1}, \dots, a_m\} = B^q \cap \{a_{r+1}, \dots, a_m\}$ .

Posons  $B^p = \{a_{p_1}, \dots, a_{p_n}\}$ , donc  $b = \sum_{j=1}^n \lambda_{p_j} a_{p_j}$ . Etant donné que le traitement ne s'ar-

rête pas à l'itération  $p$  on peut affirmer qu'il existe  $h_p \in \{p_1, \dots, p_n\}$  tel que  $\lambda_{h_p} < 0$ . Soit  $c_p$  le vecteur déterminant l'hyperplan séparateur calculé à l'itération  $p$ ,  $a_r$  sortant de base durant cette itération on peut dire que le plus petit des indices  $h_p$  est  $r$  et donc  $c_p^t a_i \geq 0 \forall i \in \{p_1, \dots, r-1\}$ . De plus le fait que  $B^p$  et  $B^q$  aient la même intersection avec  $\{a_{r+1}, \dots, a_m\}$  nous permet de dire que  $c_p^t a_i \geq 0 \forall i \in \{r+1, \dots, m\} \cap \{p_1, \dots, p_n\}$ . Par

construction  $c_p^t b < 0$ , ainsi obtient-on :  $0 > c_p^t b = \sum_{j=1}^n \lambda_{p_j} a_{p_j} \geq 0$ , d'où une contradiction. ■

Le théorème précédent et sa démonstration introduisent deux concepts très importants : la **séparation** et l'**élimination** de variables.

Une autre conséquence du théorème fondamental des inégalités linéaires est le lemme

suivant du à Farkas et Minkowsky.

**Lemme 2.1 (Farkas)** *Soit  $A$  une matrice et  $b$  un vecteur, alors le système  $Ax = b, x \geq 0$  a une solution si et seulement si  $y^t b \geq 0$  pour tout vecteur  $y$  tel que  $y^t A \geq 0$ .*

Preuve :

La nécessité de la condition est immédiate car  $y^t b = y^t Ax \geq 0$  pour tout vecteur  $x \geq 0$ , tout vecteur  $y$  tel que  $y^t A \geq 0$  et que par hypothèse  $Ax = b$ . Maintenant supposons qu'il n'existe pas de vecteur  $x \geq 0$  tel que  $Ax = b$  alors  $b$  n'appartient pas au cône généré par les vecteurs colonnes de la matrice  $A$ . Aussi par le théorème 2.2 nous pouvons dire qu'il existe un hyperplan  $H = \{x | y^t x = 0\}$  tel que  $y^t b < 0$  et  $y^t A \geq 0$ . ■

Nous pouvons remarquer au vu de ce qui précède que l'hyperplan  $H$  joue un rôle majeur, ce rôle sera mis en évidence dans la section 2.1.4 où nous poserons la question de savoir si un point appartient à une région de l'espace et sinon qu'est-ce qui l'en sépare? Afin de mieux mettre en évidence la notion de séparation contenue dans le lemme de Farkas nous en donnons une forme alternative.

**Lemme 2.2 (forme alternative du lemme de Farkas)** *Soit  $A$  une matrice et  $b$  un vecteur, alors un seul des deux systèmes d'inégalités suivants a une solution  $Ax = b, x \geq 0$  ou  $y^t b < 0, y^t A \geq 0$ .*

Ici nous retrouvons le fait que soit  $b$  est dans le cône engendré par les colonnes de la matrice  $A$  soit il existe un hyperplan qui le sépare strictement de ce cône.

Maintenant nous allons voir comment savoir si une région de l'espace définie par un système d'inégalités linéaires est vide et sinon trouver au moins un point de cette région. Un des premiers à s'être intéressé aux inégalités linéaires est J.B.J. Fourier qui décrit en 1827, [30], une méthode permettant de trouver une solution, s'il en existe une, à un système d'inégalités linéaires. Ses travaux seront repris par T.S Motzkin en 1936,[71], donnant lieu à la méthode de Fourier-Motzkin que nous exposons brièvement ci-après.

Soit  $A$  une matrice  $m \times n$  et  $b$  un vecteur de dimension  $m$ , considérons alors le système d'inégalités linéaires  $Ax \leq b$ . Multiplions chacune de ces inégalités par un scalaire positif de sorte que nous ayons dans la première colonne de  $A$  des coefficients appartenant à  $\{-1, 0, +1\}$ , ainsi le système peut se réécrire de la manière suivante :

$$\begin{aligned} x_1 + a_{i2}x_2 + \dots + a_{in}x_n &\leq \beta_i & i \in \{1, \dots, m_1\} \\ -x_1 + a_{j2}x_2 + \dots + a_{jn}x_n &\leq \beta_j & j \in \{m_1 + 1, \dots, m_2\} \\ a_{k2}x_2 + \dots + a_{kn}x_n &\leq \beta_k & k \in \{m_2 + 1, \dots, m\} \end{aligned}$$

Partant de là nous pouvons écrire que :

$$\text{Max} \left\{ \sum_{l=2}^n a_{jl}x_l - \beta_j, j \in \{m_1 + 1, \dots, m_2\} \right\} \leq x_1 \leq \text{Min} \left\{ \beta_i - \sum_{l=2}^n a_{il}x_l, i \in \{1, \dots, m_1\} \right\}$$

Ainsi nous pouvons éliminer la variable  $x_1$  et réécrire un système réduit de la forme suivante :

$$\begin{aligned} \sum_{l=2}^n (a_{il} + a_{jl})x_l &\leq \beta_i + \beta_j & i \in \{1, \dots, m_1\}, j \in \{m_1 + 1, \dots, m_2\} \\ \sum_{l=2}^n a_{kl}x_l &\leq \beta_k & k \in \{m_2 + 1, \dots, m\} \end{aligned} \tag{2.1}$$



Le système ci-dessus ne possède plus que  $n - 1$  variables, chacune de ses solutions peut être utilement complétée pour obtenir une solution du système originel en choisissant pour valeur de  $x_1$  une valeur satisfaisant la relation *Min - Max* vue plus haut. Pour obtenir une solution on itère jusqu'à l'obtention d'un système à une inconnue.

**Remarque :**

On peut prouver le lemme de Farkas dans la forme suivante : le système d'inégalités linéaires  $Ax \leq b$  a une solution si et seulement si  $y^t b \geq 0$  pour tout vecteur  $y \geq 0$  tel que  $y^t A = 0$ . Ceci fut fait par H.W. Kuhn, [53].

En effet supposons que le système  $Ax \leq b$  n'ait pas de solution. Notons  $A'x' \leq b'$  le système équivalent mis sous la forme 2.1. Ce dernier n'a pas de solution, aussi par induction pouvons-nous dire que  $(0, \dots, 0, -1)$  est une combinaison linéaire à coefficients positifs ou nuls des lignes de  $[A', b']$ . Maintenant nous pouvons affirmer que chaque ligne de  $[0, A', b']$  est la somme de deux lignes de  $[A, b]$ . Partant de là  $(0, \dots, 0, -1)$  est une combinaison linéaire à coefficients positifs ou nuls de lignes de  $[A, b]$ , ce qui prouve le lemme de Farkas.

### 2.1.2 Polarité

La notion de cône polaire ou domaine corrélatif fut introduite en 1908, [87], par G. Voronoï, il en donna la définition suivante.

**Définition 2.2** *Etant donné un cône  $K \subseteq \mathbb{R}^n$  on appelle cône polaire le cône  $K^* = \{y \in \mathbb{R}^n | y^t x \geq 0 \forall x \in K\}$*

Nous allons voir ce qu'induit la notion de cône polaire du point de vue inégalités et du point de vue séparation. Les résultats suivants ont été prouvés par G. Voronoï dans le contexte plus général des cônes convexes, voir le théorème 3.2. Une matrice est dite de plein rang ligne si ses vecteurs lignes sont des vecteurs indépendants.

**Lemme 2.3** *Soit  $K = \{x \in \mathbb{R}^n | Ax \geq 0\}$  un cône polyédral,  $A$  étant une matrice  $m \times n$  de plein rang lignes. Alors  $K^* = \text{cône}\{A_i, i \in \{1, \dots, m\}\}$ ,  $A_i$  désignant la ligne d'indice  $i$  de la matrice  $A$ .*

Preuve :

Considérons un vecteur  $y = \sum_{i=1}^m \lambda_i A_i$ ,  $\lambda_i \geq 0$ ,  $\forall i \in \{1, \dots, m\}$ . Par définition de  $K$  le produit scalaire d'une ligne de  $A$  avec un vecteur  $x \in K$  est positif ou nul donc  $(\sum_{i=1}^m \lambda_i A_i)^t x \geq 0$ , par conséquent  $y^t x \geq 0$  ainsi  $y \in K^*$ .

Soit  $y \in K^*$  alors  $y^t x \geq 0 \forall x \in K$ , supposons que  $y \notin \text{cône}\{A_i, i \in \{1, \dots, m\}\}$  alors nous pouvons dire qu'il existe un hyperplan  $H = \{x | c^t x = 0\}$  tel que  $c^t y < 0$  et  $c^t A_i \geq 0 \forall i \in \{1, \dots, m\}$ , autrement dit il existe un vecteur de  $K$  dont le produit scalaire avec  $y$  est strictement négatif, une contradiction. ■

Ainsi peut-on voir qu'un polyèdre peut avoir une description par générateurs ou par séparateurs (les contraintes) et que les deux sont liées par la relation de polarité. Nous allons voir que la polarité induit un théorème de séparation et nous y reviendrons à la section 2.1.4.

**Lemme 2.4** *Soit  $K$  un cône polyédral, alors  $x_0 \in K$  si et seulement si  $\text{Min}\{x_0^t y | y \in K^*\} \geq 0$*

Preuve :

Immédiate depuis la définition du cône polaire  $x^t y \geq 0 \forall y \in K^*$ , donc en particulier pour  $y_0 = \text{Argmin}\{x^t y \mid y \in K^*\}$ . ■

**Corollaire 2.1** Soit  $y_0$  un solution optimale de  $\text{Min}\{x_0^t y \mid y \in K^*\} < 0$  alors  $H = \{x \mid y_0^t x = 0\}$  est un hyperplan qui sépare  $x_0$  de  $K$ .

Ce dernier corollaire donne une interprétation géométrique de la séparation déterminée par les rayons extrêmes du cône polaire.

### 2.1.3 Dualité

Nous introduisons la notion de programme dual et décrivons ici les principaux résultats de la dualité en programmation linéaire. Le lecteur pourra se reporter soit au livre de V. Chvátal, [19], soit à celui de A. Schrijver, [79]. Bien que contenu dans le théorème fondamental des inégalités linéaires, théorème 2.2, la notion ne fut réellement explicitée que par von Neuman en 1947, [86], et Kuhn et Tucker en 1951, [54]

Considérons un programme linéaire  $\text{Max}\{c^t x \mid Ax \leq b, x \geq 0\}$  et supposons que nous cherchions à déterminer un majorant de la valeur optimale de ce programme qui soit une fonction des contraintes de ce programme linéaire. Nous pouvons obtenir ce majorant de la manière suivante : considérons un vecteur  $y \geq 0$  dont la dimension est égale au nombre de lignes de la matrice  $A$  que nous supposons de plein rang ; notre problème devient alors de trouver  $y$  tel que  $c^t x \leq y^t Ax \leq y^t b$ . La quantité  $y^t b$ ,  $y$  satisfaisant certaines contraintes, est bien un majorant de la valeur optimale du programme linéaire. Aussi aimerions-nous que ce majorant soit le plus petit possible. On est amené à chercher la valeur optimale du programme linéaire  $\text{Min}\{y^t b \mid y^t A \geq c, y \geq 0\}$ . Ce programme est appelé programme dual. Cette définition de la dualité en programmation linéaire nous permet d'énoncer le théorème suivant.

**Théorème 2.3 (Dualité faible)**

$$\text{Max}\{c^t x \mid Ax \leq b, x \geq 0\} \leq \text{Min}\{y^t b \mid y^t A \geq c, y \geq 0\}.$$

Preuve :

Le théorème est la conséquence immédiate de la construction du programme dual. ■

**Remarques :**

- Par construction le programme dual du dual est le primal.
- Si le programme primal a une solution alors le programme dual a un optimum fini.

La relation liant les deux programmes duaux est plus forte que celle qui vient d'être vue. Elle est donnée par le théorème suivant.

**Théorème 2.4 (Dualité forte)** Soit  $A$  une matrice  $m \times n$ ,  $c \in \mathbb{R}^n$  et  $b \in \mathbb{R}^m$ . Alors :

$$\text{Max}\{c^t x \mid Ax \leq b\} = \text{Min}\{y^t b \mid y^t A = c, y \geq 0\}$$

Le théorème de dualité forte nous permet d'écrire que pour tout  $x$  solution optimale du primal et pour tout  $y$  solution optimale du dual, on a  $c^t x = y^t A x = y^t b$  et donc  $y^t (A x - b) = 0$ ; cette dernière condition appelée condition d'écart complémentaires signifie que si une contrainte du primal est vérifiée de manière lâche la variable duale correspondante est nulle. En réécrivant cette condition dans le sens variables primales contraintes duales, on obtient  $(c - y^t A)x = 0$ , ce qui signifie que si  $x$  est une solution optimale du problème primal alors la fonction objective  $c$  appartient au cône des contraintes serrées qui définissent  $x$ ; ceci nous renvoie directement au théorème 2.2.

### 2.1.4 Séparation et optimisation

Nous avons vu jusqu'à présent comment déterminer si un système d'inégalités linéaires a une solution mais nous avons négligé deux questions : peut-on déterminer si un polyèdre est vide en temps polynomial de la dimension de l'espace? Comment trouver l'optimum d'une fonction objective sur un polyèdre, peut-on le faire en temps polynomial? La question de la complexité des problèmes a été soulevée par J. Edmonds dans [24], article dans lequel il parle de problèmes pour lesquels il existe des algorithmes mieux que finis.

Dès 1910, C. de la Vallée Poussin, [22], s'intéressant au problème d'approximation de Chebyshev qui consiste à trouver le minimum de  $\|Ax - b\|_\infty$ , décrit une méthode qui peut être considérée comme précurseur de l'algorithme du simplexe proposé en 1947 par G.B. Dantzig, [21]. Cet algorithme qui procède par changement de base, pivotage, n'est pas polynomial au moins pour les règles de pivotage connues jusqu'ores; nous ne décrivons pas cet algorithme. Le lecteur peut se reporter à [72] ou à [79]. Jusqu'en 1979 aucun algorithme polynomial n'était connu pour résoudre un programme linéaire qui est un problème de la forme  $Max\{c^t x | Ax \leq b, x \geq 0\}$ . L. Khachiyan, s'inspirant des travaux de N.Z. Shor [82] et [83], proposa le premier algorithme, connu sous le nom de méthode des ellipsoïdes, prouvant que la programmation linéaire est un problème polynomial. C'est cet algorithme que nous allons exposer succinctement ci-dessous, puis nous en verrons une conséquence importante due à M. Grötschel, L. Lovász et A. Schrijver, [45].

#### La méthode de Khachiyan

Nous commençons cette étude en rappelant la définition d'ellipsoïde, en donnant quelques propriétés et en introduisant des notions sur la taille des sommets d'un polyèdre.

**Définition 2.3** Une matrice symétrique  $M$  est dite définie positive si  $x^t M x > 0 \forall x \neq 0$

**Définition 2.4** Une région  $\mathcal{R} \in \mathbb{R}^n$  est un ellipsoïde s'il existe une matrice définie positive  $D$  et un vecteur  $z \in \mathbb{R}^n$  tels que  $\mathcal{R} = \{x \in \mathbb{R}^n | (x - z)^t D (x - z) \leq 1\}$ ,  $z$  est le centre de l'ellipsoïde.

Cette définition montre qu'un ellipsoïde est une transformation affine de la boule unité  $B = \{x \in \mathbb{R}^n | x^t x \leq 1\}$ . Le théorème suivant précise de quelle manière on peut déterminer un ellipsoïde contenant un demi-ellipsoïde.

**Théorème 2.5** Soit  $E$  un ellipsoïde de paramètres  $(z, D)$  et soit  $a \neq 0$  un vecteur de  $\mathbb{R}^n$ . Soit  $E'$  un ellipsoïde contenant  $E \cap \{x \mid a^t x \leq a^t z\}$  et tel que le volume de  $E'$  soit le plus petit possible. Alors  $E'$  est unique, il a pour paramètres  $(z', D')$  avec :

$$\begin{aligned} z' &= z - \frac{1}{n-1} \frac{Da}{\sqrt{a^t D a}} \\ D' &= \frac{n-1}{n^2-1} \left[ D - \frac{2}{n-1} \frac{Da a^t D}{a^t D a} \right] \end{aligned}$$

de plus  $\frac{\text{vol}(E')}{\text{vol}(E)} < e^{-\frac{1}{2n+2}}$ .

Etant donné un polyèdre  $P \subseteq \mathbb{R}^n$  on définit la facette-complexité de  $P$ ,  $\psi$ , comme étant le plus petit entier  $\psi \geq n$  tel qu'il existe un système d'inégalités linéaires à coefficients rationnels  $Ax \leq b$  décrivant  $P$  et dont chacune des lignes a une taille bornée par  $\psi$ . La taille d'une ligne est la longueur du plus grand numérateur ou dénominateur des coefficients qui la composent. On définit un paramètre semblable pour les sommets: la complexité sommets  $\nu$  comme étant le plus petit entier  $\nu \geq n$  tel qu'il existe des vecteurs à composantes rationnelles  $x_1, \dots, x_k, y_1, \dots, y_l$  chacun de taille au plus  $\nu$  et tels que  $P = \text{conv}(\{x_1, \dots, x_k\}) + \text{cône}\{y_1, \dots, y_l\}$ . De plus on a  $\nu \leq 4n^2\psi$  et  $\psi \leq 4n^2\nu$ .

La méthode des ellipsoïdes travaille en plusieurs temps: tout d'abord on s'assure qu'un polyèdre est non vide et si tel est le cas elle donne un point intérieur de ce polyèdre. Ensuite on calcule un point  $x_{<}$  optimal à  $\epsilon$  près. Enfin on transforme le point intérieur  $x_{<}$  en un sommet  $x^*$  optimal. Pour la clarté de l'exposé nous supposons dans ce qui suit que le polyèdre  $P$  est borné et de pleine dimension et que nous pouvons effectuer tout calcul aussi précisément que requis. Dans la première étape qui consiste à déterminer un point intérieur du polyèdre la méthode va construire une suite d'ellipsoïdes de volume décroissant jusqu'à ce qu'un d'entre eux ait son centre dans le polyèdre. Pour initialiser la méthode le premier ellipsoïde est la boule de rayon  $2^\nu$ ,  $P \subseteq \{x \mid \|x\| \leq 2^\nu\}$ . Donc le centre de ce premier ellipsoïde est  $z_0 = 0$  et la matrice associée est  $D_0 = 2^{2\nu} Id$ ,  $Id$  désignant la matrice identité. Supposons maintenant que nous soyons rendus à l'étape  $i$ , alors  $P \subseteq E_i$  qui est un ellipsoïde de paramètres  $(z_i, D_i)$ . Si  $z_i \in P$  alors nous avons trouvé un point de  $P$ , sinon supposons que  $z_i$  viole l'inégalité  $a_{j_0} x \leq b_{j_0}$  appartenant au système qui définit  $P$ . Si  $E_{i+1}$  est le plus petit ellipsoïde contenant  $E_i \cap \{x \mid a_{j_0}^t x \leq a_{j_0}^t z_i\}$ . Alors étant donné que  $P$  est contenu tant dans  $E_i$  que dans le demi-espace  $\{x \mid a_{j_0}^t x \leq a_{j_0}^t z_i\}$  on peut dire que  $P \subseteq E_{i+1}$ . Nous avons vu au théorème 2.5 comment déterminer  $(z_{i+1}, D_{i+1})$  les paramètres de  $E_{i+1}$ . Nous avons vu que les ellipsoïdes  $E_i$  et  $E_{i+1}$  ont des volumes dont le rapport est inférieur à  $e^{-\frac{1}{2n+2}}$ , par ailleurs nous savons que le volume de l'ellipsoïde initial est inférieur à  $4^{n\nu}$  aussi par récurrence nous pouvons dire que  $\text{vol}(E_{i+1}) \leq 4^{n\nu} e^{-\frac{i+1}{2n+2}}$ . Etant donné que nous avons supposé que  $P$  est borné et qu'il est de pleine dimension nous pouvons dire qu'il existe  $n+1$  vecteurs affinement indépendant  $x_0, \dots, x_n$  tels que  $P = \text{conv}(\{x_0, \dots, x_n\})$  aussi  $\text{vol}(P) \geq \text{vol}(\text{conv}(\{x_0, \dots, x_n\})) \geq 2^{-2n\nu}$ . Supposons que nous ayons exécuté un nombre  $I$  d'itérations de la méthode, on a alors :

$$2^{-2n\nu} \leq \text{vol}(P) \leq \text{vol}(E_I) < 4^{n\nu} e^{-\frac{I}{2n+2}}$$

Nous voyons bien que la proposition ci-dessus ne tient plus si on prend un nombre d'itérations suffisamment grand,  $I \geq 16^{n^2\nu}$  par exemple. Aussi nous pouvons affirmer que nous aurons trouver un point de  $P$  avant d'atteindre l'itération  $I$ , celle-ci est totalement déterminée par un polynôme de la taille des descripteurs de  $P$ .

La méthode de Khachiyan nous permet donc de trouver en un temps polynomial un point

d'un polyèdre. Elle permet aussi de déterminer de manière approchée l'optimum d'une fonction objective  $c$ . Considérons  $x_0$  un point appartenant à l'intérieur du polyèdre  $P$  et deux sphères centrées en  $x_0$ , une de rayon  $r$  contenue dans  $P$  et l'autre de rayon  $R$  contenant  $P$ . L'algorithme démarre donc avec une matrice définie positive  $D_0 = R^2 Id$ . Considérons à l'itération  $j$  le point courant  $x_j$ . On note  $\delta_j = c^t x_j$  la valeur de la fonction objective au point  $x_j$ . Si  $x_j$  appartient à l'intérieur de  $P$  on calcule alors un ellipsoïde  $E_{j+1}$  de paramètre  $x_{j+1}, D_{j+1}$  contenant  $P \cap \{x \mid -c^t x < -\delta_j\}$  comme vu précédemment. Si  $x_j$  n'appartient pas à l'intérieur de  $P$  alors il existe une inégalité  $a_{i_j}^t x \leq b_{i_j}$  parmi celles définissant  $P$  telle que  $a_{i_j}^t x_j \geq b_{i_j}$ , on calcule un ellipsoïde  $E_{j+1}$  contenant  $E_j \cap \{a_{i_j}^t x \leq a_{i_j}^t x_j\}$  comme vu lors de la recherche de points admissibles. Grâce à un argument sur le rapport de volume des ellipsoïdes on peut montrer qu'une solution approchée à  $\epsilon$  près est obtenue après un nombre d'itérations égal à  $2n(n+1) \lceil \log \frac{2R^2 \|c\|}{r\epsilon} \rceil$ .

A ce stade nous possédons un algorithme polynomial pour résoudre de manière approchée un programme linéaire, nous allons voir que via une transformation polynomiale du programme linéaire, l'algorithme en fournit une solution optimale. Pour cela modifions la fonction objective  $c$  du programme linéaire comme suit. On pose  $\xi = 2\nu^{2n}$  et on considère la fonction objective  $c_0 = c\xi + (1, \xi, \xi^2, \dots, \xi^{n-1})^t$ . Sans perte de généralité on peut supposer que  $c$  est un vecteur dont les coordonnées sont des entiers.

**Proposition 2.1** *Soit  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  un polyèdre rationnel. Le programme linéaire  $Max\{c_0^t x, x \in P\}$  a un optimum unique  $x^*$  qui est une solution optimale du programme linéaire  $Max\{c^t x, x \in P\}$ .*

Preuve :

Considérons  $\bar{x}$  un sommet de  $P$  distinct de  $x^*$ , ces deux vecteurs étant supposés à coordonnées rationnelles on peut poser  $\bar{x} = (\frac{\bar{p}_1}{\bar{q}_1}, \dots, \frac{\bar{p}_n}{\bar{q}_n})^t$  et  $x^* = (\frac{p_1^*}{q_1^*}, \dots, \frac{p_n^*}{q_n^*})^t$ . La différence de ces deux vecteurs est alors donnée par le terme générique :

$$x_i^* - \bar{x}_i = (p_i^* \bar{q}_i - q_i^* \bar{p}_i) \left( \prod_{j \neq i} q_j^* \bar{q}_j \right) \left( \prod_{j=1}^n q_j^* \bar{q}_j \right)^{-1}$$

Ceci permet de dire qu'il existe  $\alpha < \frac{1}{2}\xi$  et un vecteur  $u$  entier dont chaque composante d'indice  $i$  vérifie  $|u_i| < \xi$  tel que  $x^* - \bar{x} = \frac{u}{\alpha}$ . Par hypothèse  $x^*$  est l'optimum unique du programme  $Max\{c_0^t x, x \in P\}$  aussi pouvons-nous écrire :

$$c_0^t (x^* - \bar{x}) = c_0^t \frac{u}{\alpha} = \frac{1}{\alpha} (c^t \xi^n u + \sum_{i=1}^n \xi^{i-1} u_i) \geq 0$$

Etant donné que  $|U_i| < \xi$  on a :  $\left| \sum_{i=1}^n \xi^{i-1} u_i \right| \leq \left| (\xi - 1) \sum_{i=1}^n \xi^{i-1} \right| = |\xi^n - 1|$ . Comme

$\xi^n c^t u \geq \sum_{i=1}^n \xi^{i-1} u_i$  et que  $c^t u$  est un entier relatif il vient que  $c^t u \geq 0$ . Aussi a-t-on

établi que quelque soit le sommet  $\bar{x}$  on a  $c^t x^* \geq c^t \bar{x}$ , ce qui montre que  $x^*$  est une solution optimale du programme  $Max\{c^t x, x \in P\}$ . Par ailleurs on peut remarquer que  $c_0^t (x^* - \bar{x}) \neq 0$ , ce qui montre que  $x^*$  est l'optimum unique du programme  $Max\{c_0^t x, x \in P\}$ . ■

On peut donc dire que nous disposons d'un algorithme polynomial pour résoudre un

programme linéaire. Remarquons que dans l'étape de recherche d'un point de l'espace appartenant au polyèdre  $P$  nous avons supposé que celui-ci était donné explicitement par le système d'inégalités linéaires qui le définit. L'algorithme que nous venons de voir est polynomial mais le nombre de contraintes  $m$  intervient dans ce polynôme. Que se passe-t-il si le nombre de contraintes est grand, de taille exponentielle par rapport au nombre de variables, ou si le polyèdre nous est donné de manière implicite?

### Le problème de séparation

Considérons un polyèdre  $P$  défini par un ensemble d'inégalités linéaires dont la taille maximale est  $\psi$ . De plus supposons que nous ne connaissons pas explicitement cet ensemble d'inégalités, mais que nous soyons capables de résoudre, en un temps polynomial de  $n$  et de  $\psi$ , le problème suivant :

**Séparation** : étant donné un vecteur à composantes rationnelles  $y$ , appartient-il à  $P$ ? Sinon retourner un hyperplan  $H = \{x \mid c^t x = b\}$  tel que  $c^t y < b$  et  $c^t u \geq b \forall u \in P$ .

Remarquons que cette définition du problème de séparation nous renvoie au théorème 2.2 de Farkas, Minkowsky et Weyl. On peut remarquer que si le polyèdre  $P$  est de pleine dimension la méthode des ellipsoïdes permet de trouver un vecteur de  $P$  en un temps polynomial de  $n$  et  $\psi$ , pour peu qu'on puisse traiter le problème de séparation sur le polyèdre considéré en temps polynomial. Ce fait fut remarqué par M. Grötschel, L. Lovász et A. Schrijver en 1981, [45], ainsi que par R. Karp et C. Papadimitriou, [50]. En effet la résolution en temps polynomial du problème de séparation nous permet de tester en temps polynomial à chacune des étapes si le centre de l'ellipsoïde appartient au polyèdre  $P$  et si ce n'est pas le cas de déterminer une contrainte qui est violée.

Le théorème suivant, dont le lecteur pourra trouver la preuve dans [79], précise ce qui vient d'être dit sur la relation existant entre réalisabilité et séparation.

**Théorème 2.6** *Il existe un algorithme  $\mathcal{R}$  permettant de tester la réalisabilité d'un système d'inégalités linéaires, de taille maximum  $\psi$ , définissant un polyèdre  $P \in \mathbb{R}^n$  et un polynôme  $\chi(\psi, n)$  de sorte que si les entrées de l'algorithme de réalisabilité sont  $\psi, n$  et un algorithme de séparation sur  $P, \mathcal{S}$ , alors  $\mathcal{R}$  calcule un vecteur  $x \in P$  ou conclut que ce polyèdre est vide en un temps borné par  $\alpha\chi(\psi, n)$ ,  $\alpha$  étant le temps maximum nécessaire à  $\mathcal{S}$  pour s'exécuter sur des entrées de taille maximum  $\chi(\psi, n)$ .*

Ce théorème a des conséquences très importantes, notamment en ce qui concerne l'optimisation. Nous allons voir maintenant comment cela conduit à l'équivalence des problèmes de séparation et d'optimisation. On définit le problème d'optimisation sur un polyèdre rationnel  $P$  par :

**Optimisation** : étant donné un vecteur à composantes rationnelles  $c$ , fournir une des réponses suivantes :

1.  $x_0 = \text{Argmin}\{c^t x \mid x \in P\}$
2.  $y_0 \in \text{recône}(P)$  tel que  $c^t y > 0$ ,  $\text{recône}(P) = \{y \in \mathbb{R}^n \mid x + y \in P, \forall x \in P\}$  est appelé cône de récession ou cône caractéristique de  $P$ . Ceci correspond au cas où l'optimum sur  $P$  de la fonction objective  $c$  n'est pas borné.
3. montrer que  $P$  est vide.

**Corollaire 2.2** *Il existe un algorithme  $\mathcal{O}$  tel que si les entrées de  $\mathcal{O}$  sont  $(n, \psi, \mathcal{S})$ ;  $n$  et  $\psi$  étant des entiers naturels,  $\mathcal{S}$  étant un algorithme de séparation sur le polyèdre rationnel  $P$  défini par des inégalités linéaires de taille bornée par  $\psi$ , alors  $\mathcal{O}$  résout pour la fonction objective  $c$  à composantes rationnelles le problème d'optimisation sur  $P$  en un temps borné par un polynôme de  $n, \psi$ , la taille de  $c$  et le temps d'exécution de l'algorithme  $\mathcal{S}$ .*

Nous ne donnons ici que des éléments de preuve pour ce corollaire ainsi que pour le suivant concernant le cas où l'optimum de la fonction objective  $c$  sur le polyèdre  $P$  est fini, nous invitons le lecteur à se reporter à l'article original [45], à [79] page 178, à [68] pages 201 à 208, ou bien encore à [72] pages 162 à 163 pour une preuve exhaustive.

Squelette de preuve :

Supposons que les paramètres  $(n, \psi, \mathcal{S})$  et  $c$  satisfont les hypothèses du corollaire et que  $\xi$  soit la taille du vecteur  $c$ . Etant donné que l'optimum,  $\delta$ , du programme  $\text{Max}\{c^t x \mid x \in P\}$  est borné on peut affirmer qu'il est atteint par un vecteur de taille au plus  $4n^2\psi$ . Aussi pouvons-nous dire que la taille de  $\delta$  est inférieure ou égale à  $2\xi + 8n^2\psi$ . Posons  $\beta = 3\xi + 12n^2\psi$  et prenons  $m_0 = -2^\beta$  et  $M_0 = 2^\beta$ . Procédons alors par dichotomie en testant grâce à l'algorithme  $\mathcal{R}$  si le polyèdre  $P \cap \{x \mid c^t x \geq \frac{m_i + M_i}{2}\}$  est vide ou non. S'il est vide on met à jour le majorant :  $M_{i+1} = \frac{m_i + M_i}{2}$ , sinon on met à jour le minorant :  $m_{i+1} = \frac{m_i + M_i}{2}$ . Le nombre d'étapes nécessaires à ce parcours dichotomique de l'intervalle  $[m_0, M_0]$  nous assure alors d'avoir obtenu l'optimum du programme  $\text{Max}\{c^t x \mid x \in P\}$  en un temps borné par un polynôme de  $n, \psi$  et  $\xi$ . ■

Nous allons voir maintenant la forme polaire du corollaire 2.2 qui montre que le problème de séparation se réduit polynomialement au problème d'optimisation.

**Corollaire 2.3** *Il existe un algorithme  $\mathcal{O}^*$  tel que si les entrées de  $\mathcal{O}^*$  sont  $(n, \psi, \mathcal{O})$ ;  $n$  et  $\psi$  étant des entiers naturels,  $\mathcal{O}$  étant un algorithme d'optimisation sur le polyèdre rationnel  $P$  défini par des inégalités linéaires de taille bornée par  $\psi$ , alors  $\mathcal{O}^*$  résout pour tout vecteur  $y$  à composantes rationnelles le problème d'optimisation sur  $P$  en un temps borné par un polynôme de  $n, \psi$ , la taille de  $y$  et le temps d'exécution de l'algorithme  $\mathcal{O}$ .*

De même que pour le corollaire précédent nous ne donnons qu'un canevas de la preuve que le lecteur pourra trouver dans les références précédemment indiquées.

Squelette de preuve :

Considérons  $P^*$  le polaire de  $P = \{x \in \mathbb{R}^n \mid Ax \geq 0\}$ , alors par définition  $P^* = \{y \in \mathbb{R}^n \mid y^t A \geq 0\}$ ; ainsi peut-on dire que si  $P$  est décrit par un ensemble d'inégalités de taille maximum  $\psi$  il en va de même pour  $P^*$ . Par ailleurs nous avons vu au lemme 2.4 que  $x_0 \in P$  si et seulement si  $\text{Min}\{x_0^t y \mid y \in P^*\} \geq 0$  et réciproquement  $y_0 \in P^*$  si et seulement si  $\text{Min}\{y_0^t x \mid x \in P\} \geq 0$ . Ainsi peut-on voir que si on sait optimiser en temps polynomial sur le polyèdre  $P$  alors on sait séparer en temps polynomial sur son polaire  $P^*$  et réciproquement. Par hypothèse on possède un algorithme d'optimisation sur  $P$  donc de séparation sur  $P^*$ . Alors par le corollaire 2.2 on peut dire qu'il existe un algorithme s'exécutant en un temps polynomial de  $n, \psi$  et le temps d'exécution de l'algorithme  $\mathcal{O}$  permettant de résoudre le problème d'optimisation sur  $P^*$ ; l'existence de l'algorithme  $\mathcal{O}^*$  suit alors. ■

Les corollaires 2.2 et 2.3 nous permettent d'énoncer le résultat suivant qui établit l'équi-

valence entre les problèmes de séparation et d'optimisation.

**Corollaire 2.4 (Grötschel, Lovász, Schrijver)** *Pour tout polyèdre rationnel  $P$  le problème de séparation peut être résolu en temps polynomial si et seulement si le problème d'optimisation peut être résolu en temps polynomial.*

**Corollaire 2.5** *Si  $P$  est un polyèdre rationnel et  $P^*$  son polaire, alors les quatre propositions suivantes sont équivalentes :*

1. *Il existe un algorithme polynomial pour résoudre le problème d'optimisation sur  $P$ ;*
2. *il existe un algorithme polynomial pour résoudre le problème d'optimisation sur  $P^*$ ;*
3. *il existe un algorithme polynomial pour résoudre le problème de séparation sur  $P$ ;*
4. *il existe un algorithme polynomial pour résoudre le problème de séparation sur  $P^*$ .*

## 2.2 Programmation mathématique en nombres entiers

Si la nature offre à bien des égards l'aspect d'un continuum il existe des situations naturelles qui présentent plus que des discontinuités ; elles sont formés de quanta, introduisant un aspect discret. Ces situations qui recourent aux nombres entiers surviennent dès qu'on est appelé à manipuler des entités non seccables : un mouton demeure indivisible du moins en tant qu'élément du troupeau. Aujourd'hui les problèmes en nombres entiers et plus particulièrement les problèmes en variables binaires ont une grande importance dans l'activité humaine car ils permettent la modélisation des problèmes de décision : on associera naturellement une variable binaire à une action, si on détermine que la valeur de la variable est 1 l'action sera menée à terme, par contre si la valeur de la variable est 0 l'action ne sera pas entreprise. De plus la numérisation de l'information tant pour son traitement que pour son transport et son entreposage accroît l'intérêt de grands secteurs économiques pour la résolution des problèmes de programmation mathématique en variables binaires. D'autant que ces derniers sont aussi généraux que les problèmes définis avec des entiers quelconques. En effet considérons une variable  $x$  qui est bornée par  $b$ , soit  $p$  un entier tel que  $2^p$  est la plus petite puissance de 2 qui soit supérieure à  $b$ , alors la variable  $x$  peut être remplacée par  $p$  variables binaires associées aux puissances de 2 de 0 à

$p - 1$  et  $x \leq b$  devient  $\sum_{i=0}^{p-1} 2^i x_i \leq b$ . Remarquons que cette transformation est polynomiale

car elle utilise  $\lceil \text{Log}_2 b \rceil$  variables. Nous appellerons programme linéaire en nombres entiers un programme mathématique d'une des deux formes :

$$\left\{ \begin{array}{l} \text{Max} \quad c^t x \\ \text{s.c.} \quad Ax \leq b \\ \quad \quad x \in \mathbf{Z}^n \end{array} \right. \quad \text{ou} \quad \left\{ \begin{array}{l} \text{Max} \quad c^t x \\ \text{s.c.} \quad Ax \leq b \\ \quad \quad x \in \{0, 1\}^n \end{array} \right.$$

Nous avons vu précédemment qu'un programme linéaire peut être résolu en un temps polynomial de sa taille, qu'en est-il si on rajoute des contraintes d'intégrité sur les coordonnées des vecteurs solution ?

**Théorème 2.7** *Le problème d'existence d'une solution à un programme linéaire en variables binaires est  $\mathcal{NP}$ -complet.*



Preuve :

Considérons une proposition logique de  $n$  variables booléennes. Le problème de satisfaisabilité, qui consiste à déterminer une affectation des variables de sorte que la proposition soit vraie, est connu pour être  $\mathcal{NP}$ -complet, [35] page 161. Supposons que la proposition logique soit en forme normale conjonctive formée de  $m$  disjonctions dont  $I_1, \dots, I_m$  sont les ensembles d'indices des variables présentes dans chacune des disjonctions. La proposition sera satisfaite si et seulement si chacune des disjonctions est vraie pour l'instanciation des variables considérée. Une disjonction  $j \in \{1, \dots, m\}$  est vraie si et seulement si  $\sum_{i \in I_j^+} x_i + \sum_{i \in I_j^-} (1 - x_i) \geq 1$ ,  $x_i \in \{0, 1\}$ , où  $I_j^+$  est l'ensemble des indices pour lesquels la variable  $x_i$  est présente et  $I_j^-$  celui pour lesquels  $\bar{x}_i$  est présente. Finalement on peut dire que la proposition logique pourra être satisfaite si et seulement si il existe une solution au système de contraintes suivant :  $\sum_{i \in I_j^+} x_i + \sum_{i \in I_j^-} (1 - x_i) \geq 1, \forall j \in \{1, \dots, m\} \forall x \in \{0, 1\}^n$ .

Ainsi la satisfaisabilité d'une proposition logique se réduit polynomialement à la recherche d'une solution à coordonnées 0-1 d'un ensemble de contraintes linéaires. ■

La programmation en variables 0 – 1 est, dans le cas général, un problème  $\mathcal{NP}$ -difficile, toutefois il existe des cas de programmes linéaires en variables binaires qui peuvent être résolus en temps polynomial, notamment si la matrice  $A$  est totalement unimodulaire et si le vecteur  $b$  est entier ou si le système d'inégalités est total dual intègre, nous reviendrons sur ce dernier point plus tard.

### 2.2.1 Autres formes de polarité : Bloquants et Antibloquants

Considérons un ensemble  $E$  et  $F$  une famille de sous-ensembles de  $E$  ; à un sous-ensemble  $f \in F$  on associe un vecteur  $x_f$  de dimension  $|E|$  dont la composante relative à  $e \in E$  vaut 1 si  $e \in f$  et 0 sinon,  $x_f$  est le vecteur caractéristique de  $f$  relativement à  $E$ . On peut naturellement se poser la question de savoir quelle est la famille,  $G$ , de sous-ensemble de  $E$  telle que  $\forall g \in G$  l'ensemble  $g \cap f$  est non vide  $\forall f \in F$ . En termes d'inégalités linéaires cette question se résume à trouver les solutions du système :  $x_f^t y \geq 1 \forall f \in F, y \in \{0, 1\}^{|E|}$ . La théorie des polyèdres bloquants introduite par D.R. Fulkerson au début des années soixante-dix permet de répondre partiellement à cette question. Ses travaux fondateurs se trouvent dans [32], [33] et [34]. De plus nous allons voir qu'elle permet de définir un concept de séparation plus naturel pour les problèmes d'optimisation combinatoire que la polarité définie par Voronoï. Mais avant cela voyons ce que sont les polyèdres bloquants et quelques unes de leurs propriétés.

**Théorème 2.8** *Les trois définitions ci-dessous sont équivalentes :*

1.  $P$  est un polyèdre de  $\mathbb{R}^{n,+}$  et  $\forall y \in \mathbb{R}^n, \forall x \in P y \geq x \implies y \in P$ .
2. Il existe une famille de  $m$  vecteurs  $\{c_1, \dots, c_m\} \subseteq \mathbb{R}^{n,+}$  telle que :  $P = \text{conv}(\{c_1, \dots, c_m\}) + \mathbb{R}^{n,+} = \{c_1, \dots, c_m\}^\uparrow$ .  $P$  est appelé le dominant des vecteurs  $c_1, \dots, c_m$ .
3. Il existe une famille de  $t$  vecteurs  $\{d_1, \dots, d_t\} \subseteq \mathbb{R}^{n,+}$  telle que :

$$P = \{x \in \mathbb{R}^{n,+} \mid d_j^t x \geq 1, \forall j \in \{1, \dots, t\}\}.$$

On dira alors que  $P$  est un polyèdre de type bloquant.

Cette troisième définition, proche de celle de cône (stabilité par addition de vecteurs de  $\mathbb{R}^{n,+}$ ), rejoint notre préoccupation originelle : supposons que la famille  $\{d_1, \dots, d_t\}$  nous soit donnée, un vecteur  $x \in \mathbb{R}^{n,+}$  intersecte-t-il les sous-ensembles représentés par les vecteurs  $\{d_1, \dots, d_t\}$ ? La réponse est oui si  $d_j^t x \geq 1 \forall j \in \{1, \dots, t\}$  et non s'il existe  $j \in \{1, \dots, t\}$  tel que  $d_j^t x < 1$  dans ce dernier cas l'hyperplan d'équation  $d_j^t x = 1$  sépare strictement  $x$  de  $P$ .

**Définition 2.5** On appelle polyèdre bloquant d'un polyèdre  $P$  le polyèdre  $B(P) = \{y \in \mathbb{R}^{n,+} \mid y^t x \geq 1, \forall x \in P\}$ .

Poursuivons notre parallèle entre la notion de bloquant et la notion de polarité avec le théorème suivant qui établit pour la notion de polyèdre bloquant d'un polyèdre de type bloquant des propriétés semblables à celles établies par G. Voronoï pour le cône polaire d'un cône convexe.

**Théorème 2.9** (Fulkerson 1970) Soit  $P \subseteq \mathbb{R}^{n,+}$  un polyèdre de type bloquant, alors :

1.  $B(P)$  est aussi un polyèdre de type bloquant.
2.  $B(B(P)) = P$ .
3.  $x \in P \iff x \geq 0$  et  $y^t x \geq 1 \forall y \in B(P)$ .
4.  $P = \{c_1, \dots, c_m\}^\dagger$  si et seulement si  $B(P) = \{y \in \mathbb{R}^{n,+} \mid y^t c_i \geq 1, \forall i \in \{1, \dots, m\}\}$

Preuve :

– Montrons la deuxième proposition du théorème.

1.  $P \subseteq B(B(P))$ .

Si  $x \in P$  alors par définition du bloquant d'un polyèdre  $y^t x \geq 1, \forall y \in B(P)$  et donc  $x \in B(B(P))$ .

2.  $B(B(P)) \subseteq P$

$P$  est de type bloquant donc il existe une famille de vecteurs  $\{a_1, \dots, a_m\} \in \mathbb{R}^{n,+}$  telle que  $P = \{x \in \mathbb{R}^{n,+} \mid a_i^t x \geq 1, \forall i \in \{1, \dots, m\}\}$ . Considérons  $y \notin P$  alors il existe  $i_0 \in \{1, \dots, m\}$  tel que  $a_{i_0}^t y < 1$ . Or par définition de  $B(P)$  on a  $a_{i_0} \in B(P)$  et nécessairement  $y \notin B(B(P))$ . Aussi peut-on conclure que  $B(B(P)) \subseteq P$ .

– Montrons la quatrième proposition du théorème.

1.  $P = \{c_1, \dots, c_m\}^\dagger \implies B(P) = \{y \in \mathbb{R}^{n,+} \mid y^t c_i \geq 1, \forall i \in \{1, \dots, m\}\}$ .  
 $P = \{c_1, \dots, c_m\}^\dagger$ , alors  $x \in P$  implique que :

$$x = \sum_{i=1}^m \alpha_i c_i + y ; 0 \leq \alpha_i \leq 1 ; \sum_{i=1}^m \alpha_i = 1 ; y \in \mathbb{R}^{n,+}.$$

Par définition du polyèdre bloquant de  $P$ ,  $u \in B(P)$  signifie que pour tout  $x \in P$  :

$$u^t x = u^t \left( \sum_{i=1}^m \alpha_i c_i + y \right) = u^t \left( \sum_{i=1}^m \alpha_i c_i \right) + u^t y \geq 1.$$

Etant donné que  $\sum_{i=1}^m \alpha_i c_i \in P$  on a  $u^t \left( \sum_{i=1}^m \alpha_i c_i \right) \geq 1$  ce qui permet d'écrire que  $u^t y \geq 0$ , ce qui est toujours vérifié car  $u$  et  $y \in \mathbb{R}^{n,+}$ . On peut donc conclure que  $B(P) = \{y \in \mathbb{R}^{n,+} \mid y^t c_i \geq 1, \forall i \in \{1, \dots, m\}\}$ . Ce qui, au passage, prouve le premier point du théorème.

2.  $B(P) = \{y \in \mathbb{R}^{n,+} \mid y^t c_i \geq 1, \forall i \in \{1, \dots, m\}\} \implies P = \{c_1, \dots, c_m\}^\dagger$ .  
Soit  $P' = \{c_1, \dots, c_m\}^\dagger$  alors par le sens direct de la proposition on peut dire que :

$$B(P') = \{y \in \mathbb{R}^{n,+} \mid y^t c_i \geq 1, \forall i \in \{1, \dots, m\}\}$$

et par hypothèse :

$$\{y \in \mathbb{R}^{n,+} \mid y^t c_i \geq 1, \forall i \in \{1, \dots, m\}\} = B(P)$$

donc  $B(P) = B(P')$ . Maintenant par la deuxième proposition du théorème on a  $B(B(P)) = P = B(B(P')) = P'$ , ce qui achève la démonstration.

- Le point 3 du théorème est une conséquence immédiate de la définition de bloquant d'un polyèdre bloquant. En effet :

$$\begin{array}{lll} x \in P \implies y^t x \geq 1 & \forall y \in B(P) & \text{par définition de } B(P) \\ y^t x \geq 1 \forall y \in B(P) & \implies x \in B(B(P)) & \text{par définition de } B(P) \\ & x \in P = B(B(P)) & \text{par le point 2 du théorème} \end{array}$$

- Le point 1 a déjà été prouvé dans le point 4. ■

De la définition de polyèdre bloquant d'un polyèdre de type bloquant et par le point 2 du théorème de Fulkerson nous pouvons affirmer que  $x_0 \in P$ , un polyèdre de type bloquant, si et seulement si  $\text{Min}\{x_0^t y, y \in B(P)\} \geq 1$  et que si  $y^*$  est une solution optimale de  $\text{Min}\{x_0^t y, y \in B(P)\}$  telle que  $x_0^t y^* < 1$  alors l'hyperplan  $H = \{x \mid x^t y^* = 1\}$  sépare strictement  $x_0$  de  $P$ . Ce qui permet de dire que le problème de séparation sur  $P$  est équivalent au problème d'optimisation sur  $B(P)$ . Ce résultat est dû à M. Grötschel, L. Lovász et A. Schrijver et s'énonce de la manière suivante :

**Théorème 2.10 (Grötschel, Lovász, Schrijver)** *Soit  $P \subseteq \mathbb{R}^{n,+}$  un polyèdre bloquant et  $B(P)$  son bloquant alors il existe un algorithme polynomial pour résoudre le problème d'optimisation sur  $P$  si et seulement si il existe un algorithme polynomial pour résoudre le problème d'optimisation sur  $B(P)$ .*

Preuve :

Soit  $l \in \mathbb{R}^n$  et  $\alpha$  une solution optimale de  $\text{Min}\{l^t x, x \in P\}$ , alors par définition du bloquant d'un polyèdre on peut dire que  $l \in B(P)$  si et seulement si  $l^t \alpha \geq 1$ . Aussi pouvons nous dire que si  $l \notin B(P)$  l'hyperplan d'équation  $\alpha^t y = 1$  sépare strictement  $l$  de  $B(P)$ . Ceci

montre que si on sait résoudre en temps polynomial le problème d'optimisation sur  $P$  on sait résoudre en temps polynomial le problème de séparation sur  $B(P)$ . De plus nous savons par le corollaire 2.5 que s'il existe un algorithme de séparation s'exécutant en temps polynomial sur un polyèdre alors il existe un algorithme d'optimisation s'exécutant en temps polynomial sur ce polyèdre ; ainsi pouvons-nous affirmer qu'il existe un algorithme polynomial d'optimisation sur  $B(P)$ . ■

Avant de conclure ce paragraphe consacré aux polyèdres bloquant citons deux exemples.

- Considérons un graphe orienté  $G = (V, E)$  et deux de ses sommets  $s$  et  $t$ . Soit  $C$  l'ensemble des  $s-t$  coupes de  $G$ , les éléments de  $C$  sont minimaux pour l'inclusion. Il est connu que le polyèdre  $\{x \in \mathbb{R}^{|E|, +} \mid y_c x^t \geq 1, \forall c \in C\}$  a ses sommets en bijection avec les  $s-t$  chemins de  $G$ , que son bloquant est le polyèdre des coupes et qu'on peut optimiser en temps polynomial sur chacun d'eux.
- Soit  $E$  un ensemble et  $S_1, \dots, S_m$   $m$  sous-ensembles de  $E$ , le problème qui consiste à trouver un sous-ensemble de  $E$  qui intersecte chacun des sous-ensemble  $S_i, \forall i \in \{1, \dots, m\}$  est connu pour être  $\mathcal{NP}$ -complet (Set covering problem, voir [35] page 221) et peut s'écrire : trouver  $x \in \mathbb{R}^m$  tel que  $\sum_{e \in S_i} x_e \geq 1 \forall i \in \{1, \dots, m\}$ ,  $x \in \{0, 1\}^m$ , ce qui détermine un polyèdre bloquant.

**Théorème 2.11** *Soit  $F = \{f_1, \dots, f_n\}$  une famille fine de sous-ensembles propres d'un ensemble  $E$ , soit  $G = \{g \subset E \mid g \cap f \neq \emptyset \forall f \in F\}$ ,  $g$  minimum pour l'inclusion. Alors  $\{x_f, f \in F\}^\dagger = \{x \mid x \geq 0, y_g^t x \geq 1 \forall g \in G\}$  si et seulement si pour toute fonction objective à coefficients rationnels positifs ou nuls on a  $\text{Min}\{w(f), f \in F\} = \text{Max}\{\mathbf{1}^t y \mid y \geq 0, y(G \ni e) \leq w_e\}$ .*

Preuve :

1. Nécessité de la condition.

Par la dualité en programmation linéaire :

$$\text{Max}\{\mathbf{1}^t z \mid y \geq 0, y_g^t z \leq w_e \forall g \in G, g \ni e\} = \text{Min}\{w^t x \mid x \geq 0, y_g^t x \geq 1, \forall g \in G\}$$

Par hypothèse  $\{x_f, f \in F\}^\dagger = \{x \mid x \geq 0, y_g^t x \geq 1 \forall g \in G\}$ , ce qui permet d'écrire que :

$$\text{Min}\{w^t x \mid x \geq 0, y_g^t x \geq 1, \forall g \in G\} = \text{Min}\{w^t x \mid x \in \{F_1, \dots, F_n\}^\dagger\}$$

De plus  $w \in \mathbb{Q}^{n, +}$  et donc :

$$\text{Min}\{w^t x \mid x \in \{F_1, \dots, F_n\}^\dagger\} = \text{Min}\{w(f), f \in F\}$$

2. Suffisance de la condition.

Se montre avec des arguments similaires à la nécessité. ■

Le théorème précédent établit une relation *Min* – *Max* et donne partant de là une condition pour que le problème d’optimisation défini sur les vecteurs caractéristiques de sous-ensembles puisse être résolu en temps polynomial. Si  $F$  est l’ensemble des  $s - t$  coupes le théorème dit que la coupe de poids minimum vaut le packing (un packing de sous-ensembles d’un ensemble est une collection des ces sous-ensembles disjoints deux à deux) de  $s - t$  chemins de sorte qu’un arc de poids  $w_e$  n’est pas emprunté par plus de  $w_e$  chemins, autrement dit “flot-max=coupe-min”. Des ensembles  $F$  et  $G$  qui vérifient le théorème sont dits  $\mathbb{Q}^+$  max-flow-min-cut.

### 2.2.2 Enveloppe convexe de vecteurs entiers

Considérons une famille  $F$  de sous-ensembles d’un ensemble  $E$  et  $x_f$  le vecteur caractéristique de  $f \in F$ . On sait que  $\text{Min}\{w_f, f \in F\} = \text{Min}\{w^t x, x \in \text{conv}(\{x_f, f \in F\})\}$ , où  $w$  est une pondération des éléments de  $E$  et  $\text{conv}(\{x_f, f \in F\})$  est l’enveloppe convexe des vecteurs  $x_f$ . On peut donc dire que si on sait déterminer en temps polynomial l’enveloppe convexe d’un ensemble de vecteurs on sait optimiser en temps polynomial sur cet ensemble de vecteurs. C’est dire l’importance en optimisation combinatoire de la génération de l’enveloppe convexe de vecteurs à coordonnées  $0 - 1$ .

Dans ce qui suit nous décrivons succinctement deux méthodes de génération de l’enveloppe convexe de vecteurs à coordonnées entières. La première due à R. Gomory, [42] et [43], et V. Chvátal, [18], consiste à générer des hyperplans séparant les solutions fractionnaires de programme linéaires de leurs solutions entières. Une variante a été proposée par A. Caprara et M. Fischetti, [15]. La deuxième due à E. Balas, [4] et [5], consiste à convexifier une disjonction de polyèdres. Cette disjonction survient naturellement en programmation en variables binaires, en effet  $\{x \mid Ax \leq b, x_i \in \{0, 1\}\}$  peut s’exprimer par  $\{x \mid Ax \leq b, x_i = 0\} \cup \{x \mid Ax \leq b, x_i = 1\}$ . Ces deux méthodes ont des relations très fortes avec ce qui est exposé au chapitres 5 et 6.2.

### Totale duale intégrité

La notion de systèmes d’inégalités vérifiant la propriété de totale duale intégrité a été introduite en 1977 par J. Edmonds et Giles, [25]. Dans ce qui suit nous n’employerons plus que l’abréviation TDI.

Par le théorème de dualité forte en programmation linéaire nous savons que :

$$\text{Max}\{c^t x \mid Ax \leq b\} = \text{Min}\{y^t b \mid y^t A = c, y \geq 0\},$$

les coefficients de  $A$ ,  $b$  et  $c$  étant supposés être rationnels. Aussi peut-on dire que si l’optimum est entier pour tout vecteur  $c$  dont les coordonnées sont entières alors il est atteint par une solution entière pour tout vecteur  $c$  pour lequel l’optimum est fini. Le système d’inégalités  $Ax \leq b$  est dit TDI si le minimum de la relation de dualité a une solution entière pour tout vecteur  $c$  à coordonnées entières pour lequel l’optimum est borné.

La notion de système TDI est centrale pour la programmation linéaire en nombres entiers et mériterait que nous lui consacrons davantage d’attention. En effet cette notion établit des relations de dualité entre polyèdres dont les points extrêmes sont à composantes entières. L’importance de la notion de système TDI est illustrée par les deux résultats ci-dessous.

**Théorème 2.12** *Un polyèdre rationnel  $P$  a ses points extrêmes à coordonnées entières si et seulement si chacun des ses hyperplans supports contient des vecteurs à coordonnées*

entières.

Squelette de preuve :

L'interprétation géométrique est assez claire, en effet si  $P$  est un polyèdre entier alors chacune de ses faces contient au moins un vecteur entier et comme l'intersection de  $P$  avec un de ses hyperplans supports détermine une face de  $P$  il en résulte que cet hyperplan contient des vecteurs entiers. Ceci prouve la nécessité de la condition. La suffisance de la condition peut être prouvée en développant l'argument suivant : considérons une face minimale de  $P$  qui ne contient pas de vecteur à coordonnées entières, si  $P$  n'est pas entier une telle face existe, alors on peut montrer en utilisant la définition de TDI que l'hyperplan support de  $P$  qui détermine cette face ne contient pas de vecteur entier, ce qui prouve le théorème par contraposée. ■

**Corollaire 2.6** *Si le système d'inégalités  $Ax \leq b$  est TDI et si  $b$  est un vecteur dont les composantes sont des entiers alors les points extrêmes du polyèdre  $P = \{x \mid Ax \leq b\}$  sont des vecteurs dont les composantes sont des entiers.*

Squelette de preuve :

Le système  $Ax \leq b$  étant TDI, le minimum est atteint par un vecteur dont les composantes sont entières pour tout vecteur  $c$  entier pour lequel l'optimum est fini. Le vecteur  $b$  étant entier la valeur de cet optimum est entière aussi pouvons-nous dire que l'optimum du maximum est atteint par une solution à coordonnées entières pour toute fonction objective rationnelle. ■

La notion de système TDI fut complétée par R. Giles et W. Pulleyblank qui introduirent la notion de base de Hilbert, [36], il s'agit des descripteurs entiers des polyèdres. Nous reviendrons plus tard sur cette notion, notamment au chapitre 7.

### La méthode de Gomory-Chvátal

Considérons un polyèdre  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  et supposons que nous cherchions l'enveloppe convexe des points à coordonnées entières de ce polyèdre,  $P_0$ . Soit  $c^t x \leq \delta$  une inégalité valide pour  $P$ , alors il est clair que  $\lfloor c \rfloor^t x \leq \lfloor \delta \rfloor$  où  $\lfloor c \rfloor$  est le vecteur obtenu depuis  $c$  en prenant la partie entière inférieure de chacune de ses coordonnées, est une inégalité valide pour  $P_0$ . Si on prend  $c^t x \leq \delta$  comme combinaison linéaire à coefficients positifs ou nuls des lignes de  $Ax \leq b$  et qu'on itère le processus un nombre fini de fois on génère  $P_0$ . C'est ce que nous allons voir maintenant.

Considérons un polyèdre rationnel  $P$ , soit  $H = \{x \mid c^t x \leq \delta\}$  un demi-espace tel que  $P \subseteq H$ , nous supposons que les composantes de  $c$  sont des entiers premiers entre eux. Alors on peut dire que l'enveloppe convexe des vecteurs à coordonnées entières de  $H$  est  $H_0 = \{x \mid c^t x \leq \lfloor \delta \rfloor\}$ . On définit  $P^{(1)} = \bigcap_{H \supseteq P} H_0$ . Le fait que  $P \subseteq H$  implique  $P_0 \subseteq H_0$

permet de définir une hiérarchie de relaxation de la manière suivante :

$$P_0 \subseteq \dots \subseteq P^{(2)} \subseteq P^{(1)} \subseteq P \quad (2.2)$$

La méthode de Gomory consiste à résoudre des programmes linéaires sur  $P$ ,  $P^{(1)}$ ,  $P^{(2)}$  etc... jusqu'à ce qu'une solution entière soit obtenue. Nous allons voir maintenant que le

nombre d'itérations de cette méthode est fini. Une première étape dans cette démonstration consiste à montrer que chacun des  $P^{(1)}, P^{(2)}, \dots$ , est généré de manière finie par un système d'inégalités linéaires.

**Théorème 2.13** *Soit  $P \in \mathbb{R}^n$  un polyèdre rationnel,  $P = \{x \mid Ax \leq b\}$ , tel que le système  $Ax \leq b$  est TDI et la matrice  $A$  est à coefficients entiers. Alors  $P^{(1)} = \{x \mid Ax \leq \lfloor b \rfloor\}$ .*

Preuve :

Considérons un polyèdre  $P$  satisfaisant les hypothèses du théorème. Soit  $a_i^t x \leq b_i$  une inégalité du système  $Ax \leq b$ , cette inégalité détermine un demi-espace  $H^i$  contenant  $P$ . Soit  $H_0^i$  l'enveloppe convexe des vecteurs à coordonnées entières de  $H^i$ , on a alors  $H_0^i \subseteq \{x \mid a_i^t x \leq \lfloor b_i \rfloor\}$ , ce qui implique  $P^{(1)} \subseteq \{x \mid Ax \leq \lfloor b \rfloor\}$ .

Considérons un demi-espace  $H = \{x \mid c^t x \leq \delta\}$  tel que  $P \subseteq H$ , supposons que les coordonnées de  $c$  sont des entiers premiers entre eux, donc  $H_0 = \{x \mid c^t x \leq \lfloor \delta \rfloor\}$ . Etant donné que  $H$  contient  $P$  par le théorème de dualité forte en programmation linéaire on peut écrire que :

$$\delta \geq \text{Max}\{c^t x \mid Ax \leq b\} = \text{Min}\{y^t b \mid y \geq 0, y^t A = c\}$$

Par hypothèse le système  $Ax \leq b$  est TDI, aussi le minimum est-il atteint par un vecteur à coordonnées entières,  $\bar{y}$ , et on a :

$$\left. \begin{array}{l} Ax \leq \lfloor b \rfloor \\ c^t x = \bar{y}^t Ax \end{array} \right\} \implies c^t x \leq \bar{y}^t \lfloor b \rfloor \leq \lfloor \bar{y}^t b \rfloor \leq \lfloor \delta \rfloor$$

Ainsi peut-on dire que  $\{x \mid Ax \leq \lfloor b \rfloor\} \subseteq H_0$ , ceci étant vrai pour chaque demi-espace contenant  $P$  on peut conclure que  $\{x \mid Ax \leq \lfloor b \rfloor\} \subseteq P^{(1)}$ . ■

Si on considère une face  $F$  d'un polyèdre rationnel  $P$  on peut voir aisément que  $F^{(1)} = P^{(1)} \cap F$  et par récurrence que  $F^{(j)} = P^{(j)} \cap F$ .

Poursuivons notre démonstration en montrant que le nombre d'éléments de la hiérarchie de relaxation est fini.

**Théorème 2.14** *Pour tout polyèdre rationnel  $P$ ,  $P_0$  peut être atteint en un nombre fini d'étapes,  $n$ . C'est-à-dire que  $P_0 = P^{(n)}$ .*

Preuve :

Considérons un polyèdre rationnel  $P$  que nous supposons de pleine dimension et contenant des vecteurs à composantes entières. Alors il existe une matrice  $A$  dont les coefficients sont des entiers et deux vecteurs  $b \in \mathbb{Q}^m$  et  $b' \in \mathbb{N}^m$ . tels que  $P = \{x \mid Ax \leq b\}$  et  $P_0 = \{x \mid Ax \leq b'\}$ . Soit  $a^t x \leq \beta$  une inégalité prise parmi  $Ax \leq b$  et  $a^t x \leq \beta'$  l'inégalité correspondante dans  $Ax \leq b'$ . Soit  $H = \{x \mid a^t x \leq \beta'\}$ , supposons que  $\forall j P^{(j)} \not\subseteq H$ . Par définition on peut dire que  $P^{(1)} \subseteq \{x \mid a^t x \leq \lfloor \beta \rfloor\}$ , aussi pouvons-nous affirmer qu'il existe deux entiers  $\beta''$  et  $k$  satisfaisant :

$$\left. \begin{array}{l} \beta' < \beta'', P^{(j)} \subseteq \{x \mid a^t x \leq \beta''\} \forall j \geq k \\ P^{(j)} \not\subseteq \{x \mid a^t x \leq \beta'' - 1\} \forall j \geq k \end{array} \right\}. \quad (2.3)$$

Soit  $F$  la face de  $P$  induite par l'hyperplan  $\{x \mid a^t x = \beta''\}$ . L'hypothèse  $P^{(j)} \not\subseteq H \forall j$  nous conduit à dire que  $F$  ne contient pas de vecteurs dont les coordonnées sont des entiers. Aussi, par induction, nous pouvons dire qu'il existe une itération de rang  $q$  telle que  $F^{(q)} = \emptyset$ . Etant donné que  $F^{(i)} = P^{(i)} \cap F$  on a :

$$\emptyset = P^{k+q} \cap F = P^{k+q} \cap \{x \mid a^t x = \beta''\}.$$

Ce qui conduit à dire que  $P^{k+q} \subseteq \{x \mid a^t x \leq \beta''\}$  implique  $P^{k+q} \subseteq \{x \mid a^t x \leq \beta'' - 1\}$ . Ce qui contredit 2.3. Ainsi vient-on de montrer qu'il existe un entier  $j$  tel que  $P^{(j)} \subseteq H$ , ceci étant vrai pour chacune des inégalités de  $Ax \leq b'$ . On peut conclure à la finitude de la méthode de Gomory dans ce cas.

Si  $P$  ne contient pas de vecteurs dont les coordonnées sont des entiers alors il existe un hyperplan d'équation  $c^t x = \delta$  contenant  $P$ , avec  $c$  un vecteurs à coordonnées entières et  $\delta$  non entier. Alors  $P \subseteq \{x \mid c^t x \leq \delta\}$  et  $P \subseteq \{x \mid c^t x \geq \delta\}$  et donc  $P^{(1)} \subseteq \{x \mid c^t x \leq \lfloor \delta \rfloor\} \cap \{x \mid c^t x \geq \lceil \delta \rceil\} = \emptyset$ . ■

La preuve de la méthode des plans coupants de Gomory que nous venons de donner n'est pas originale, toutefois elle justifie sa place ici lorsqu'elle est mise en regard de la preuve de la méthode de L. Lovász et A. Schrijver et surtout de la preuve de la finitude de l'algorithme de coupes polyédrales que nous présentons à la section 6.2.

### Disjonction et convexification

Comme nous l'avons déjà vu un programme linéaire en variables binaires, plus généralement en variables entières, peut être vu comme une disjonction de polyèdres. Malheureusement l'union de polyèdres n'est en général pas convexe ce qui obère la résolution des problèmes d'optimisation. Toutefois si on sait générer l'enveloppe convexe d'une disjonction de polyèdres on sait résoudre les problèmes d'optimisation sur un tel domaine. E. Balas a donné dans ses travaux préalablement cités les conditions de convexification d'une disjonction. Ces conditions sont précisées par le théorème suivant.

**Théorème 2.15** *Soit  $\{P_i = \{x \in \mathbb{R}^n \mid A_i x \geq b_i\}$ ,  $i \in I\}$  un ensemble de polyèdres non vides. Alors  $\text{conv}(\bigcup_{i \in I} P_i)$  est l'ensemble des vecteurs  $x \in \mathbb{R}^n$  pour lesquels il existe des vecteurs  $(y_i, y_0^i) \in \mathbb{R}^n \times \mathbb{R}^+$  tels que :*

$$\left. \begin{aligned} x &= \sum_{i \in I} y_i \\ A_i y_i &\geq b_i y_0^i \quad \forall i \in I \\ y_0^i &\geq 0, \quad \sum_{i \in I} y_0^i = 1 \end{aligned} \right\} \quad (2.4)$$

Preuve :

– Montrons que  $x \in \text{conv}(\bigcup_{i \in I} P_i) \implies x$  satisfait 2.4.

Dire que  $x \in \text{conv}(\bigcup_{i \in I} P_i)$  c'est dire que  $x$  peut s'écrire comme combinaison linéaire

convexe d'éléments de  $\bigcup_{i \in I} P_i$ . Posons  $x = \sum_{i \in I} \alpha_i z_i$ ,  $\alpha_i \geq 0$ ,  $\sum_{i \in I} \alpha_i = 1$ ,  $z_i \in P_i \quad \forall i \in I$ .

Dire que  $z_i \in P_i$  signifie que  $A_i z_i \geq b_i$  soit en multipliant par  $\alpha_i \geq 0$  cette inégalité  $\alpha_i A_i z_i = A_i \alpha_i z_i \geq \alpha_i b_i$ . Si on pose  $y_i = \alpha_i z_i$  et  $y_0^i = \alpha_i$  on a déterminé des vecteurs  $(y_i, y_0^i)$ ,  $\forall i \in I$  tels que  $(y_i, y_0^i)$  et  $x$  satisfont 2.4.



– Montrons que si  $x$  et  $(y_i, y_0^i)$ ,  $\forall i \in I$  satisfont 2.4 alors  $x \in \text{conv}(\bigcup_{i \in I} P_i)$ .

Par hypothèse:  $A_i y_i \geq y_0^i b_i$  donc si  $y_0^i \neq 0$  on a  $\frac{1}{y_0^i} A_i y_i \geq b_i$ ,  $y_0^i \geq 0$ ,  $\sum_{i \in I} y_0^i = 1$  et

$x = \sum_{i \in I} y_i$ , posons  $z_i = \frac{1}{y_0^i} y_i$ . Clairement  $z_i \in P_i \forall i \in I$ . De plus nous pouvons écrire

que  $x = \sum_{i \in I} y_0^i z_i$ ,  $x$  est combinaison linéaire convexe des  $z_i$ .

■

Ces travaux concernant la programmation disjonctive ont été repris par E. Balas, S. Ceria et G. Cornuéjols pour la résolution des problèmes en variables mixtes, ils sont largement présentés au chapitre 5.

### 2.2.3 Treillis et corps convexes

Les méthodes de résolution des programmes linéaires en nombres entiers que nous avons vues jusqu'à présent sont des méthodes qui peuvent être qualifiées de duales. Duales en ce sens que le point courant n'est pas une solution du problème, les contraintes d'intégrité étant en général violées, mais une solution du problème relaxé. Lorsqu'un point à coordonnées entières est atteint il est solution optimale du problème en nombres entiers. C'est notamment ce que nous avons vu avec la méthode de Gomory. Toutefois on peut imaginer une autre approche de la programmation linéaire en nombre entiers, cette approche prenant plus en compte la nature discrète du problème à résoudre. Ce serait une méthode primale en ce sens qu'à chaque étape de l'algorithme la solution courante vérifierait l'ensemble des contraintes, y compris celles d'intégrité. Cette approche de la programmation linéaire en nombres entiers a fait l'objet de nombreux travaux et permis d'établir certains résultats essentiels tels celui de H. Lenstra, [58], montrant que la programmation linéaire en variables entières peut être résolue en temps polynomial lorsque le nombre de variables est fixé. Citons encore celui de W. Cook, J. Fonlupt et A. Schrijver, [20], où il est montré que toute solution d'un programme en variables entières peut s'exprimer comme combinaison linéaire à coefficients entiers strictement positifs d'un nombre polynomial,  $2n - 1$ , de vecteurs pris parmi les générateurs du polyèdre sur lequel est défini le programme. Un ensemble de vecteurs à coordonnées entières tel que tout vecteur à coordonnées entières peut s'écrire comme combinaison linéaire des vecteurs de cet ensemble dont les coefficients sont des entiers naturels est appelé base de Hilbert. Cette notion fut introduite par R. Giles et W. Pulleyblank, [36]. Cette notion est importante pour l'approche primale que nous voulons présenter mais aussi pour l'algorithmique combinatoire. Les algorithmes d'élimination de cycles de coûts moyens négatifs sont parmi les plus connus pour résoudre les problèmes de flots; il existe un ensemble de cycles indépendants qui forment une base de Hilbert pour les flots. K. Aardal, R. Weismantel et L. Wolsey ont dressé un état de l'art concernant ces approches dans [1]. Par ailleurs R. Weismantel a développé cette approche dans [88].

Considérons un programme linéaire en variables entières :

$$Plne = \text{Min}\{c^t x \mid Ax = b, x \text{ entier}\}.$$

Etant donné  $x_0$  un point particulier de  $\{Ax = b, x \text{ entier}\}$  et  $x$  un point quelconque de cet ensemble, considérons le vecteur  $y = x_0 - x \in L = \{x \mid Ax = 0, x \text{ entier}\}$ . il est connu

qu'un tel ensemble forme un treillis et peut se réécrire de la manière suivante :

$$L = \{x \mid x = B\lambda, \lambda \text{ entier}\}$$

où  $B$  est la matrice dont les colonnes sont les vecteurs à composantes entières générant  $L$ . Si  $B$  est un ensemble de colonnes minimum il appelé base de Hilbert de  $L$ . On peut toujours supposer que  $c \geq 0$  et qu'au moins une de ses composantes est non nulle. La matrice  $D$  de terme générique  $D_{ii} = c_i, D_{ij} = 0, i \neq j$  est définie positive et induit une norme :  $\|x\|_D = \sqrt{x^t D x}$ . On peut alors s'intéresser au problème qui consiste à trouver le vecteur  $x \in L$  qui minimise  $\|x\|_D$ , si  $x \in \{0, 1\}^n$  alors  $c^t x = x^t D x$ . Il a été conjecturé que ce problème, connu sous le nom de plus "court" vecteur d'un treillis, est  $\mathcal{NP}$ -complet. Nous allons voir comment on peut malgré tout approcher ce problème en prenant le plus petit, au sens de la norme, vecteur parmi ceux qui composent une base réduite du treillis.

### Réduction de base d'un treillis

On définit  $\det(L)$  comme le déterminant d'une matrice dont les vecteurs colonnes forment une base du treillis  $L$ , on a alors  $\det(L) \leq \prod_{i=1}^n \|b_i\|$  quelque soit la base  $\{b_1, \dots, b_n\}$ . Par ailleurs Ch. Hermite, [48], montra que pour tout treillis il existe une base  $\{b_1, \dots, b_n\}$  telle que  $\prod_{i=1}^n \|b_i\|_D \leq c \times \det(L)$  où  $c$  ne dépend que de la dimension du treillis. Nous allons voir l'algorithme de L. Lovász qui permet de déterminer une base qui satisfait cette deuxième inégalité pour une valeur de  $c$  qui n'est pas la meilleure connue mais pour  $c' = 2^{n(n-1)\frac{1}{4}}$ , [57]. Soit  $\{b_1, \dots, b_n\}$  une base du treillis  $L$ . Dans une première étape l'algorithme de L. Lovász calcule une orthogonalisation de Gram-Schmidt des vecteurs  $b_1, \dots, b_n$  :

$$b_1^* = b_1, b_j^* = b_j - \sum_{k=1}^{j-1} \mu_{jk} b_k^*, \mu_{jk} = \frac{b_j^t D b_k^*}{\|b_k^*\|^2}$$

**Remarque :** Le vecteur  $b_j^*$  est orthogonal, au sens de la matrice  $D$ , au sous-espace engendré par les vecteurs  $b_1^*, \dots, b_{j-1}^*$  et donc les vecteurs de l'orthogonalisation de Gram-Schmidt sont orthogonaux deux à deux. Un multiplicateur  $\mu_{jk}$  est nul si et seulement si les vecteurs  $b_k^*$  et  $b_j$  sont orthogonaux.

Une base sera dite réduite si les deux conditions suivantes sont satisfaites :

$$\begin{aligned} |\mu_{jk}| &\leq \frac{1}{2} \forall k, \forall j \in \{1, \dots, n\}, k < j & 2.5.1 \\ \|b_j^* - \mu_{j,j-1} b_{j-1}^*\|^2 &\geq \frac{3}{4} \|b_{j-1}^*\|^2 \forall j \in \{2, \dots, n\} & 2.5.2 \end{aligned} \quad (2.5)$$

La condition 2.5.1 est satisfaite si  $b_j$  et  $b_k^*$  sont quasi orthogonaux ou si la composante de  $b_j$  suivant  $b_k^*$  est petite. Si cette condition est violée on remplace  $b_j$  par  $b_j - [\mu_{jk} - \frac{1}{2}] b_k^*$ . Supposons que la condition 2.5.1 soit satisfaite, il se peut alors que nous ayons un premier vecteur qui soit assez grand puis un deuxième petit relativement au premier et ainsi de suite. Afin d'éviter ce fait la condition 2.5.2 est imposée. Si cette dernière est violée on échange  $b_j$  et  $b_{j-1}$ , puis on recalcule une orthogonalisation de Gram-Schmidt. On itère le processus jusqu'à ce que les deux conditions soient vérifiées.

**Théorème 2.16 (Lovász)** Soit  $\{b_1, \dots, b_n\}$  une base réduite du treillis  $L \subset \mathbb{R}^n$ . Alors :

$$\det(L) \leq \prod_{j=1}^n \|b_j\| \leq c' \times \det(L), \text{ avec } c' = 2^{n(n-1)\frac{1}{4}}.$$

**Corollaire 2.7** Soit  $\{b_1, \dots, b_n\}$  une base réduite du treillis  $L \subset \mathbb{R}^n$  que nous supposons ordonnée de sorte que  $\|b_n\| = \text{Max}\{\|b_j\| \mid j \in \{1, \dots, n\}\}$ . Soit  $H$  le sous-espace engendré par les vecteurs  $b_1, \dots, b_{n-1}$  et  $h$  la distance de  $b_n$  à  $H$ . Alors :

$$\frac{1}{c'} \|b_n\| \leq h \times \|b_n\|$$

L'algorithme de L. Lovász fut généralisé à tout type de distance en 1992 par L. Lovász et H.E. Scarf, [63].

### Réduction de base et programmation entière

Nous allons voir ici l'algorithme de H.W. Lenstra, [58], pour la programmation linéaire en nombres entiers. On démontre ainsi que la programmation linéaire en nombres entiers peut être résolue en temps polynomial si le nombre de variables est fixé.

L'idée centrale des algorithmes de programmation linéaire en variables entières utilisant l'algorithme de réduction de base est qu'un treillis est contenu dans des hyperplans parallèles et dénombrables. Pour déterminer si un polyèdre contient un vecteur dont les coordonnées sont entières on peut énumérer ces hyperplans. Afin d'éviter un arbre d'énumération trop important on calcule une représentation des hyperplans du treillis telle que la distance entre eux ne soit pas trop petite. Pour ce faire on emploie l'algorithme de réduction de base.

Soit  $P = \{x \mid Ax \leq b\}$ , la question qui nous intéresse est de savoir si  $P$  contient un vecteur à composantes entières. H.W. Lenstra observa que si le polyèdre est "fin" le cas est défavorable car il peut être "coincé" entre un grand nombre d'hyperplans. Aussi suggéra-t-il d'effectuer une transformation linéaire qui a pour effet de l'enfler. Soit  $B(p, z) = \{x \mid \|x - p\| \leq z\}$  la boule de centre  $p$  et de rayon  $z$ . La transformation,  $\tau$ , appliquée au polyèdre  $P$  est construite de sorte que  $B(p, r) \subset \tau(P) \subset B(p, R)$  avec  $p \in \tau(P)$  et  $\frac{R}{r} \leq c''$ . On doit aussi appliquer la transformation  $\tau$  au treillis  $L$ , la question devient alors  $\tau(L) \cap \tau(P)$  est-il vide? Supposons  $L$  isomorphe à  $\mathbb{Z}^n$ , on peut alors remarquer que  $\{\tau(e_1), \dots, \tau(e_n)\}$  est une base de  $L$  mais que les vecteurs de cette base ne sont pas orthogonaux et qu'ils peuvent être grands. C'est ici qu'intervient la réduction de base.

**Lemme 2.5 (Lenstra)** Soit  $\{b_1, \dots, b_n\}$  une base du treillis  $L$ . Alors pour tout  $x \in \mathbb{R}^n$  il existe  $y \in L$  tel que :

$$\|x - y\|^2 \leq \frac{1}{4} \sum_{j=1}^n \|b_j\|^2$$

Posons  $L' = \tau(L)$  et considérons  $\{b_1, \dots, b_n\}$  une base réduite de  $L'$  telle que :

$$\det(L') \leq \prod_{j=1}^n \|b_j\| \leq c' \times \det(L')$$

De plus supposons les vecteurs de la base indexés de sorte que  $\|b_n\| = \text{Max}\{\|b_j\| \mid j \in \{1, \dots, n\}\}$ . Le lemme étant vrai quelque soit  $x \in \mathbb{R}^n$  il est vrai en particulier pour  $x = p$ , on peut donc trouver  $y \in L'$  tel que :

$$\|p - y\|^2 \leq \frac{1}{4} \sum_{j=1}^n \|b_j\|^2 \leq \frac{1}{4} n \|b_n\|^2$$

Si  $y \in \tau(P)$  alors nous avons une réponse à notre question, si  $y \notin P$  nous savons que  $y \notin B(p, r)$  car cette boule est totalement contenue dans  $\tau(P)$ , partant de là on peut dire que la distance à  $p$  est plus grande que le rayon de la boule, aussi :

$$r < \|p - y\| \leq \frac{1}{2}\sqrt{n} \|b_n\|$$

Maintenant on crée  $t$  sous-problèmes en considérant les intersections du polyèdre  $\tau(P)$  avec  $t$  hyperplans contenant le treillis  $L$ . Chaque sous-problème est représenté par un nœud dans l'arbre d'énumération, l'intégralité du processus est répété en chaque nœud : transformation, réduction de base et branchement. La profondeur de l'arbre d'énumération créé par ce processus récursif est au plus  $n$  car la dimension des sous-problèmes décroît de 1 à chaque niveau. Nous allons voir que le nombre de nœuds présent à un niveau donné est polynomialement borné par une grandeur qui ne dépend que de la dimension. Soit  $H$  l'hyperplan engendré par les vecteurs  $b_1, \dots, b_{n-1}$  et  $h$  la distance de cet hyperplan à  $b_n$ ,  $L''$  le treillis dont une base est  $\{b_1, \dots, b_{n-1}\}$ , ainsi peut-on écrire le treillis  $L$  de la manière suivante :

$$L = L'' + \mathbf{Z}b_n \subset H + \mathbf{Z}b_n = \bigcup_{k \in \mathbf{Z}} (H + kb_n)$$

Le treillis  $L$  est donc contenu dans un ensemble dénombrable d'hyperplans, la distance entre deux hyperplans consécutifs est  $h$  que nous savons borné par le corollaire 2.7. Afin de déterminer quel est le nombre maximum d'hyperplans qui intersectent  $P$  on lui substitue la boule  $B(p, R)$ , si  $t$  est le nombre d'hyperplans qui intersectent cette boule on a  $t - 1 \leq \frac{2R}{h}$  par ailleurs nous savons que  $2R \leq 2rc' < c'\sqrt{n} \|b_n\|$  et  $h \geq \frac{1}{c} \|b_n\|$  il vient alors que  $t - 1 \leq \frac{2R}{h} < cc'\sqrt{n}$ .

### Base de Hilbert et systèmes TDI

Pour finir ce rapide tour d'horizon sur le rôle des bases de Hilbert en programmation entière nous analysons succinctement la connexion entre ce concept et celui de système d'inégalités TDI.

La relation de dualité forte :

$$\text{Max}\{c^t x \mid Ax \leq b\} = \text{Min}\{y^t b \mid y \geq 0, y^t A = c\}$$

et la définition de systèmes TDI que nous avons vu précédemment nous permettent d'énoncer le théorème suivant.

**Théorème 2.17** *Le système d'inégalités  $Ax \leq b$  est TDI si et seulement si pour toute face  $F$  du polyèdre  $P = \{x \mid Ax \leq b\}$  les vecteurs lignes de  $A$  qui correspondent aux contraintes serrées définissant  $F$  forment une base de Hilbert.*

Preuve :

#### 1. Nécessité de la condition.

Considérons une face  $F$  de  $P = \{x \mid Ax \leq b\}$ , soit  $a_1, \dots, a_k$  les lignes de  $A$  qui correspondent aux contraintes serrées définissant  $F$ . Soit  $c \in \text{cône}(\{a_1, \dots, a_k\}) \cap \mathbf{Z}^n$ . Par hypothèse  $Ax \leq b$  est TDI aussi le  $\text{Min}$  de la relation de dualité a une solution optimale entière, ce qui montre que  $c$  peut s'écrire comme combinaison linéaire à coefficients dans  $\mathbf{N}$  des vecteurs  $a_1, \dots, a_k$ .

## 2. Suffisance de la condition.

Soit  $c$  tel que  $Max\{c^t x \mid Ax \leq b\} = Min\{y^t b \mid y \geq 0, y^t A = c\} = \delta$  soit fini. Soit  $F$  une face minimale de  $P$  telle que  $\forall x \in F$  on atteint l'optimum du  $Max$ , Alors si  $a_1, \dots, a_k$  sont les lignes de  $A$  correspondant aux contraintes qui définissent  $F$ ,  $c \in c\hat{o}ne(\{a_1, \dots, a_k\})$ . Par hypothèse  $\{a_1, \dots, a_k\}$  est une base de Hilbert donc  $c = \sum_{i=1}^k \lambda_i a_i$ . En complétant le vecteur  $(\lambda_1, \dots, \lambda_k)^t$  par des 0 on obtient un vecteur  $y$  tel que :

$$c^t x = y^t A x = y^t b \quad \forall x \in F$$

$y$  atteint l'optimum du  $Min$ , ceci étant vrai quelque soit  $c$  le système  $Ax \leq b$  est TDI. ■

Nous achevons ainsi cette revue des résultats concernant la programmation linéaire et la programmation linéaire en nombres entiers qui nous ont paru les plus importants et qui, nous le pensons, justifient les choix des problèmes que nous avons étudiés.

Après avoir introduit la programmation semi-définie positive nous présentons les applications les plus marquantes qui en ont été faite en optimisation combinatoire et nous exposons un travail que nous fîmes sur les flots avec des contraintes de longueur des chemins, ces longueurs étant exprimées par des fonction SDP-représentables.

Dans un deuxième temps nous exposons la construction de L. Lovász et A. Schrijver, [64], permettant de générer l'enveloppe convexe de points à coordonnées 0-1. Puis nous montrons comment nous avons pu dériver de cette construction un algorithme de coupes polyédrales.

La construction de L. Lovász et A. Schrijver repose sur deux cônes de matrices,  $M(K)$  et  $M^+(K)$ , dont nous étudions les bases de Hilbert.

## Chapitre 3

# Programmation semi-définie positive

Ce chapitre consacré à la programmation semi-définie positive peut être vu comme le prolongement de l'introduction ; nous y retrouvons appliqués à la programmation semi-définie positive les concepts essentiels que nous avons exposés dans le cadre de la programmation linéaire : séparation et dualité. Dans un premier temps nous rappelons certaines propriétés des matrices semi-définies positives et du cône qu'elles forment. Ensuite nous donnons une définition de programme semi-défini positif ; nous montrons comment un programme linéaire en variables 0-1 conduit naturellement à la définition d'un programme semi-défini positif et pourquoi celui-ci consiste en un renforcement de la relaxation linéaire. M. Goemans et L. Tunçel, [39] ont donné des conditions pour que ce renforcement soit strict. Nous avons vu au chapitre précédent l'importance de la dualité en programmation linéaire, il en va de même en programmation semi-définie positive avec toutefois de légères différences. Nous présentons tout d'abord une dualité lagrangienne classique appliquée à la programmation semi-définie positive et nous montrons qu'à contrario de la programmation linéaire l'existence d'une solution du programme dual ne suffit pas à assurer que le programme primal possède une solution bornée atteinte, mais qu'une condition plus forte est nécessaire (condition de Slater) pour assurer la compacité du domaine. Nous insistons sur cette forme de dualité en programmation semi-définie positive car elle nous servira pour les applications présentées au chapitre suivant. Nous présentons toutefois, brièvement, une forme étendue de la dualité en programmation semi-définie positive qui permet de s'affranchir des conditions de Slater et fournit une théorie exacte de dualité dans ce contexte. Cette forme est étendue dans le sens où le programme semi-défini positif est plongé dans un espace de plus grande dimension que celui des matrices semi-définies positives de sorte à introduire des matrices qui résorbent le saut de dualité éventuel. Enfin nous présentons une extension de la programmation semi-définie positive sur l'espace des formes sesquilinéaires hermitiennes.

### 3.1 Le cône des matrices semi-définies positives

On note  $S_n$  l'espace vectoriel des matrices carrées d'ordre  $n$  symétriques :  $A \in S_n \iff a_{ij} = a_{ji}, \forall i, \forall j \in \{1, \dots, n\}$ .

**Définition 3.1** Une matrice  $A \in S_n$  est dite semi-définie positive si  $x^t A x \geq 0 \quad \forall x \in \mathbb{R}^n$ .

Soit une matrice  $A \in S_n$ , alors les trois propositions suivantes sont équivalentes :

- $A$  est une matrice semi-définie positive.
- Toutes les valeurs propres de la matrice  $A$  sont positives ou nulles.
- Il existe une matrice  $B$  carrée d'ordre  $n$  telle que  $A = B^t B$ .

La troisième des propositions ci-dessus a pour conséquence que toute matrice semi-définie positive peut s'écrire comme matrice de Gram, c'est à dire qu'il existe  $n$  vecteurs  $x_1, \dots, x_n$  tels que  $a_{ij} = x_i^t x_j$ . Par conséquent les coefficients de la diagonale d'une matrice semi-définie positive sont positifs ou nuls.

**Théorème 3.1**  $U_{psd}$ , l'ensemble des matrices semi-définies positives, est un cône convexe.

Preuve :

- Soit  $A \in U_{psd}$  et  $\lambda$  un scalaire positif ou nul, alors :

$$x^t(\lambda A)x = \lambda(x^t A x) \geq 0 \iff (\lambda A) \in U_{psd}$$

- Soit  $A$  et  $B \in U_{psd}$  alors :

$$\forall x \ x^t(A+B)x = x^t A x + x^t B x \geq 0 \text{ et donc } (A+B) \in U_{psd}$$

■

**Remarque :** Le cône  $U_{psd}$  n'est pas un cône polyédral, c'est à dire qu'il n'est pas généré de manière finie. Ceci peut se déduire directement de la définition des matrices semi-définies positives.

On définit le produit scalaire des matrices  $A$  et  $B$ , toutes deux carrées d'ordre  $n$ , de la manière suivante :

$$A \cdot B = \sum_{i=1}^n \sum_{j=1}^n a_{ij} b_{ij} = \text{tr}(A^t B)$$

où  $\text{tr}(M) = \sum_{i=1}^n m_{ii}$  est la trace de la matrice  $M$ . Nous avons vu en 2.1.2 ce qu'est le cône polaire d'un cône, G. Voronoï démontra les propriétés suivantes :

**Théorème 3.2 (Voronoi)** Soit  $K \subseteq \mathbb{R}^n$  un cône fermé, alors :

- si  $K$  est un cône polyédral alors  $K^*$  est aussi un cône polyédral
- Si  $K$  est un cône convexe alors  $K^*$  l'est aussi et  $(K^*)^* = K$
- si  $K_1$  et  $K_2$  sont deux cônes convexes alors :  $K_1 \subset K_2 \iff K_2^* \subset K_1^*$

**Théorème 3.3** Le cône  $U_{psd}$  est autopolaire,  $U_{psd}^* = U_{psd}$ .

Preuve :

Soit  $A$  et  $B \in U_{psd}$  alors il existe deux matrices carrées d'ordre  $n$ ,  $X$  et  $Y$ , telles que :

$$A = X^t X \text{ et } B = Y^t Y$$

Le produit scalaire des matrices  $A$  et  $B$  peut alors s'écrire de la manière suivante :

$$A \cdot B = tr(A^t B) = tr(AB^t) = tr([X^t X] \cdot [Y^t Y]) = tr([XY^t] \cdot [YX^t]) = tr(M^t M)$$

$$\text{en posant } M = [YX^t]$$

Les matrices  $X$  et  $Y$  étant carrées, la matrice  $M$  est carrée, aussi la matrice  $\tilde{M} = M^t M$  est une matrice semi-définie positive. Par définition on a  $x^t \tilde{M} x \geq 0 \forall x \in \mathbb{R}^n$ , en particulier si  $x = e_i$ ,  $e_i$  étant le vecteur dont toutes les composantes sont nulles sauf celle d'indice  $i$  qui vaut 1, ce qui permet d'écrire :

$$e_i^t \tilde{M} e_i = \tilde{m}_{ii} \geq 0$$

On peut donc affirmer que  $A \cdot B = tr(\tilde{M}) \geq 0 \forall A$  et  $\forall B \in U_{psd}$ , ce qui permet de conclure que  $U_{psd} \subset U_{psd}^*$ .

Considérons une matrice symétrique  $A$  et supposons que  $A \notin U_{psd}$ , alors elle possède au moins une valeur propre  $\lambda < 0$ . Etant donné que  $A \in S_n$  elle peut s'écrire comme suit :

$$A = P \text{Diag}[\lambda, \lambda_2, \dots, \lambda_n] P^t$$

où  $P$  est une matrice dont les vecteurs colonnes sont deux à deux orthogonaux et  $\text{Diag}[\lambda, \lambda_2, \dots, \lambda_n]$  est la matrice diagonale formée avec les valeurs propres de la matrice  $A$ . Considérons une matrice  $B = P \text{Diag}[1, 0, \dots, 0] P^t$ , il est clair que  $B \in U_{psd}$ . Considérons alors le produit scalaire  $A \cdot B$  :

$$A \cdot B = tr(A^t B) = tr(P \text{Diag}[\lambda, 0, \dots, 0] P^t) = \lambda < 0$$

Ce qui montre que toute matrice ayant une valeur propre strictement négative n'appartient pas à  $U_{psd}^*$  et donc  $U_{psd}^* \subset U_{psd}$ . ■

On peut remarquer que le caractère semi-défini positif des matrices induit une relation d'ordre partiel sur  $U_{psd}$  :  $A \succeq B$  si et seulement si  $A - B$  est semi-définie positive ce qu'on notera dorénavant  $A - B \succeq 0$ .

### 3.2 Programme semi-défini positif

On appelle programme semi-défini positif un programme linéaire défini sur le cône des matrices semi-définies positives, c'est un programme de la forme :

$$Sdp = \begin{cases} \text{Max} & C \cdot X \\ & A_i \cdot X \leq b_i \quad \forall i \in \{1, \dots, m\} \\ & X \in U_{psd} \end{cases}$$



Si les matrices  $A_i$ ,  $i \in \{1, \dots, m\}$  et  $C$  sont des matrices diagonales, le programme  $Sdp$  est un programme linéaire générique et la condition  $X \in U_{psd}$  peut être remplacée par les contraintes  $x_{jj} \geq 0 \quad \forall j \in \{1, \dots, n\}$ . Nous verrons plus loin comment les programmes semi-définis positifs permettent de définir des relaxations de problèmes en nombres entiers plus efficaces que leurs relaxations linéaires. Un programme semi-défini positif peut être résolu de manière approchée en temps polynomial avec la précision choisie par un algorithme de points intérieurs, voir par exemple [2], [76].

Si un programme linéaire peut soit :

- avoir un optimum borné,
- avoir un optimum non borné,
- ne pas avoir de solution.

Un programme semi-défini positif peut aussi avoir un optimum borné qui n'est pas atteint. Le programme ci-dessous en est un exemple :

$$\begin{array}{l} \text{Max} \quad -x_1 \\ \left( \begin{array}{ccc} x_1 & 1 & 0 \\ 1 & x_2 & 0 \\ 0 & 0 & x_1 \end{array} \right) \succeq 0 \end{array}$$

On voit clairement que les solutions possibles sont  $x_1 > 0$ , aussi l'optimum de la fonction objectif est 0 et celui-ci ne peut pas être atteint.

Considérons le problème d'optimisation d'une fonction convexe suivant :

$$\begin{array}{l} \text{Min} \quad \frac{(c^t x)^2}{d^t x} \\ \text{s.c.} \quad Ax + b \geq 0 \end{array}$$

On suppose que  $d^t x > 0$  pour tous les  $x$  tels que  $Ax + b \geq 0$ . Comme il est montré dans [85] ce programme peut se transformer en programme semi-défini positif de la manière suivante. Introduisons tout d'abord une variable  $y$  de façon à obtenir une fonction objectif linéaire, on obtient alors :

$$\begin{array}{l} \text{Min} \quad y \\ \text{s.c.} \quad Ax + b \geq 0 \\ \frac{(c^t x)^2}{d^t x} \leq y \end{array} \quad (3.1)$$

Dire que  $\frac{(c^t x)^2}{d^t x} \leq y$  est équivalent à dire que  $d^t x y - (c^t x)^2 \geq 0$ , ce qui revient à dire que le déterminant de la matrice  $M$  ci-dessous est positif ou nul, celle-ci est donc semi-définie positive.

$$M = \begin{pmatrix} y & c^t x \\ c^t x & d^t x \end{pmatrix}$$

Le programme 3.1 se réécrit alors comme suit :

$$\begin{array}{l} \text{Min} \quad y \\ \text{s.c.} \quad \left( \begin{array}{ccc} \text{Diag}[Ax + b] & 0 & 0 \\ 0 & y & c^t x \\ 0 & c^t x & d^t x \end{array} \right) \succeq 0 \end{array}$$

### 3.3 Programme semi-défini positif et variables 0-1

Nous allons voir maintenant comment construire une relaxation semi-définie positive d'un programme linéaire en variables 0 – 1. Considérons le problème qui consiste à déterminer l'optimum d'une fonction objective  $c$  sur un ensemble de vecteurs à coordonnées en 0-1 satisfaisant un ensemble de contraintes,  $Ax \geq b$  par exemple. Soit  $K$  le cône défini par  $K = \{x \in \mathbb{R}^{n+1} | Ax \geq 0\}$  avec  $A = [-b, \tilde{A}]$ . La région de l'espace des vecteurs satisfaisant les contraintes originelles est définie par l'intersection de  $K$  et de l'hyperplan d'équation  $\{x \in \mathbb{R}^{n+1} | x_0 = 1\}$ . De plus on impose aux vecteurs d'être à coordonnées en 0-1. Aussi le problème à résoudre s'écrit-il de la manière suivante :

$$Plne = \begin{cases} Min & c^t x \\ s.t. & x \in S = K \cap \{x \in \mathbb{R}^{n+1} | x_0 = 1\} \cap \{0, 1\}^{n+1} \end{cases} \quad (3.2)$$

Soit  $Q$  le cône engendré par les vecteurs de  $\{x \in \{0, 1\}^{n+1} \cap \{x | x_0 = 1\}\}$ . On note  $K^*$  le cône polaire de  $K$  qui est donc le cône engendré par les lignes de la matrice  $A = [-b, \tilde{A}]$ . Dire que  $Ax \geq 0$  c'est dire que :

$$u^t x \geq 0 \quad \forall u \in K^* \quad (3.3)$$

Soit  $Q^*$  le cône polaire du cône  $Q$ . Dire que  $x_i \in \{0, 1\}, \forall i \in \{1, \dots, n\}, x_0 = 1$  implique que :

$$v^t x \geq 0 \quad \forall v \in Q^* \quad (3.4)$$

Les équations 3.3 et 3.4 permettent d'écrire :

$$\begin{aligned} (u^t x) \cdot (v^t x) &\geq 0 \\ (u^t x) \cdot (x^t v) &\geq 0 \\ u^t [x x^t] v &\geq 0 \end{aligned}$$

$[x x^t]$  est une matrice carrée  $Y$  d'ordre  $n + 1$  qui satisfait par construction les propriétés suivantes :

- $y_{ij} = x_i x_j = x_j x_i = y_{ji} \iff Y = Y^t$
- $y_{ii} = x_i^2 \geq 0$
- $u^t Y v \geq 0 \quad \forall u \in K^*, \forall v \in Q^*$

Par ailleurs étant donné qu'on veut  $x_i \in \{0, 1\}, i = 1, \dots, n$  et que  $x_0 = 1$  on peut poser que  $x_i^2 = x_i = y_{ii}$  et dire que la matrice  $Y$  vérifie  $y_{00} = 1$  et  $y_{0i} = y_{i0} = y_{ii}$ . De plus dire que  $Y = x x^t$ , c'est dire que  $Y$  peut s'écrire comme matrice de Gram et donc  $Y$  est une matrice semi-définie positive.

**Définition 3.2** On appellera  $M(K)$  l'ensemble des matrices  $Y$  qui satisfont les trois premières des propriétés ci-dessous et  $M^+(K) = M(K) \cap U_{psd}$ . On peut aisément vérifier que  $M(K)$  et  $M^+(K)$  sont des cônes convexes.

- $Y = Y^t$
- $diag(Y) = Y e_0$ , où  $diag(Y)$  est le vecteur formé par les termes diagonaux de  $Y$
- $u^t Y v \geq 0 \quad \forall u \in K^*, \forall v \in Q^*$

$$- Y \in U_{psd}$$

**Lemme 3.1** Si  $e_i$  désigne le  $i^{eme}$  vecteur unité et  $f_i = e_0 - e_i$ , alors  $Q^*$  est le cône engendré par les vecteurs  $e_i$  et  $f_i$ ,  $i \in \{0, \dots, n\}$ .

Preuve :

Par construction  $Q = \text{cône}(\{x \in \{0, 1\}^{n+1} \cap \{x | x_0 = 1\}\})$ , c'est-à-dire que si  $x \in Q$  on a :

$$x = e_0 + \sum_{i=1}^n \lambda_i e_i, \quad \lambda_i \geq 0 \quad \forall i \in \{1, \dots, n\}$$

D'autre part dire que  $u \in Q^*$  c'est dire que  $u^t x \geq 0, \forall x \in Q$ , soit encore :

$$\begin{aligned} u^t(e_0 + \sum_{i=1}^n \lambda_i e_i) &= u^t e_0 + u^t(\sum_{i=1}^n \lambda_i e_i) \\ &= u_0 + \sum_{i=1}^n \lambda_i u_i \\ &= u_0 + \sum_{i|u_i < 0} \lambda_i u_i + \sum_{i|u_i \geq 0} \lambda_i u_i \geq 0 \end{aligned} \quad (3.5)$$

Si  $u \in \text{cône}\{e_i, f_i\}$  alors il peut s'écrire  $u = \sum_{i=1}^n \phi_i f_i + \sum_{i=1}^n \eta_i e_i$  avec  $\phi_i \geq 0$  et  $\eta_i \geq 0$ , on a alors :

$$\begin{aligned} u^t x &= (\sum_{i=1}^n \phi_i f_i + \sum_{i=1}^n \eta_i e_i)^t (e_0 + \sum_{i=1}^n \lambda_i e_i) \\ &= \sum_{i=1}^n \phi_i f_i^t e_0 + \sum_{i=1}^n \eta_i e_i^t e_0 + \sum_{i=1}^n \sum_{j=1}^n \phi_i f_i^t \lambda_j e_j + \sum_{i=1}^n \sum_{j=1}^n \eta_i e_i^t \lambda_j e_j \end{aligned}$$

Par définition des  $f_i$  on a  $f_i^t e_0 = 1, e_i^t e_0 = 0, \forall i \neq 0, f_i^t e_j = 0$  si  $i \neq j, e_i^t e_j = 0$  si  $i \neq j$  et  $e_i^t e_i = 1$ , aussi peut-on réécrire l'équation ci-dessus sous la forme suivante :

$$u^t x = \sum_{i=1}^n \phi_i - \sum_{i=1}^n \phi_i \lambda_i + \sum_{i=1}^n \eta_i \lambda_i \quad (3.6)$$

En identifiant terme a terme les deux expressions de  $u^t x$  des equations 3.5 et 3.6, on peut alors poser :

- $\phi_i = -u_i$  si  $u_i < 0$  et 0 sinon
- $\eta_i = u_i$  si  $u_i \geq 0$  et 0 sinon
- $u_0 = \sum_{i=1}^n \phi_i$

Ceci montre que si  $u \in Q^*$  alors  $u$  peut s'écrire comme combinaison linéaire positive des vecteurs  $e_i$  et  $f_i$  et donc  $Q^* \subset \text{cône}\{e_i, f_i\}$ . Montrer que  $\text{cône}\{e_i, f_i\} \subset Q^*$  est immédiat. ■

Afin de renforcer notre définition de la relaxation semi-définie positive d'un programme linéaire en nombres entiers nous allons énoncer quelques propriétés supplémentaires des coefficients des matrices de  $M(K)$  et  $M^+(K)$ . De plus nous supposons toujours que le cône  $K$  est inclus dans le cône  $Q$ , cette hypothèse se fait sans perte de généralité car nous cherchons à résoudre un problème d'optimisation sur des vecteurs à composantes 0-1, donc si  $K \not\subseteq Q$  on peut toujours remplacer  $K$  par  $K \cap Q$ .

**Lemme 3.2** *Si  $K \subseteq Q$ , la matrice  $Y \in M(K)$  telle que définie en 3.2 vérifie les propriétés suivantes :*

1.  $y_{ij} \geq 0 \forall i, \forall j$
2.  $y_{ij} \leq y_{ii} = y_{0i} \leq y_{00}$
3.  $y_{ij} \geq y_{ii} + y_{jj} - y_{00}$

Preuve :

Par hypothèse  $K \subseteq Q$  et donc  $Q^* \subset K^*$ , d'autre part on sait que  $Q^* = \text{cône}\{e_i, f_i\}$  et que  $Y$  vérifie  $u^t Y v \geq 0 \forall u \in K^*, \forall v \in Q^*$  donc en particulier pour les vecteurs de  $K^*$  qui sont dans  $Q^*$ , et donc pour les rayons extrêmes de ce cône, on peut alors écrire :

$$e_i^t Y e_j = y_{ij} \geq 0$$

Les points 2 et 3 du lemme se démontrent de façon similaire. ■

On peut alors définir une relaxation semi-définie positive de (3.2) de la manière suivante :

$$Sdne = \left\{ \begin{array}{l} \text{Min } c^t \text{diag}(Y) \\ \text{Diag}(A_i) \cdot Y \geq b_i \\ y_{00} = 1 \\ y_{ij} \geq 0 \\ y_{ij} \leq y_{ii} \\ y_{ij} \geq y_{ii} + y_{jj} - y_{00} \\ \text{diag}(Y) = Y e_0 \\ Y \succeq 0 \end{array} \right.$$

où  $\text{Diag}(A_i)$  est la matrice diagonale formée avec les coefficients de la ligne  $i$  de la matrice  $A$ . Ces résultats définissant une relaxation semi-définie positive d'un programme linéaire en nombres entiers sont exposés d'une manière différente dans [47].

### 3.4 Le lemme de Farkas étendu

Le lemme de Farkas donne dans le cas linéaire les conditions d'existence d'une solution à un système d'inégalités. Nous allons voir comment il peut être étendu à la programmation semi-définie positive. Le lemme de Farkas ne peut pas être adapté au cas semi-défini positif sans adjoindre de conditions supplémentaires. Nous devons nous assurer que le cône  $K = \{A_i \cdot X; X \succeq 0\}$  est un fermé. Il est montré dans [2] que  $K$  est fermé s' il existe  $y \in \mathbb{R}^m$  tel que  $\sum_{i=1}^m y_i A_i \succ 0$ . Ce fait est à mettre en relation avec le théorème de

Weierstrass et est illustré par l'exemple où l'optimum du programme est borné mais non atteint.

**Lemme 3.3 (Farkas)** *Considérons  $m$  matrices  $A_i$  et un vecteur  $b \in \mathbb{R}^m$ . Soit  $y \in \mathbb{R}^m$  un vecteur tel que  $\sum_{i=1}^m y_i A_i \succ 0$ . Alors il existe une matrice  $X \in U_{psd}$  telle que  $A_i \cdot X = b_i$ ,  $\forall i \in \{1, \dots, m\}$  si et seulement si  $y^t b \geq 0$  pour tout vecteur  $y$  tel que  $\sum_{i=1}^m y_i A_i \succeq 0$ .*

Preuve :

1.  $\exists X \in U_{psd}$  telle que  $A_i \cdot X = b_i$ ,  $\forall i \in \{1, \dots, m\} \implies y^t b \geq 0$ ,  $\forall y$  tel que  $\sum_{i=1}^m y_i A_i \succeq 0$ . Par hypothèse  $\sum_{i=1}^m y_i A_i \succeq 0$  et  $X \in U_{psd}$ , le cône  $U_{psd}$  étant autopolaire on peut écrire :

$$\sum_{i=1}^m y_i A_i \cdot X = y^t b \geq 0$$

2.  $\exists X \in U_{psd}$  telle que  $A_i \cdot X = b_i$ ,  $\forall i \in \{1, \dots, m\} \iff y^t b \geq 0$ ,  $\forall y$  tel que  $\sum_{i=1}^m y_i A_i \succeq 0$ . Supposons que le système  $A_i \cdot X = b_i$ ,  $X \in U_{psd}$ ,  $\forall i \in \{1, \dots, m\}$  n'ait pas de solution. Alors il existe un hyperplan qui sépare strictement le vecteur  $b$  du cône  $\{c \mid c_i = A_i \cdot X; X \succeq 0\}$ ; c'est-à-dire qu'il existe un vecteur  $y$  tel que :

$$y^t b < 0 \text{ et } y^t c = \sum_{i=1}^m y_i A_i \cdot X \geq 0 \quad \forall X \succeq 0$$

ce qui implique que la matrice  $\sum_{i=1}^m y_i A_i$  est semi-définie positive, d'où une contradiction. ■

### 3.5 La dualité en programmation semi-définie positive

Un programme semi-défini positif pouvant être assimilé à un programme linéaire sur le cône  $U_{psd}$  il est naturel de rechercher, comme en programmation linéaire, des relations de dualité. Le lecteur pourra trouver un exposé complet sur la dualité en programmation semi-définie positive dans [77].

Considérons le couple de programmes semi-définis positifs suivant :

$$Sdp = \begin{cases} \text{Max} & C \cdot X \\ \text{s.c.} & A_i \cdot X = b_i \quad \forall i \in \{1, \dots, m\} \\ & X \in U_{psd} \end{cases} \quad Dsdp = \begin{cases} \text{Min} & y^t b \\ \text{s.c.} & \sum_{i=1}^m y_i A_i - C \succeq 0 \end{cases}$$

Avant d'aborder la dualité nous allons voir une condition de régularité, condition de Slater, qui lorsqu'elle est satisfaite nous permettra d'établir un théorème de dualité forte. Cette condition a déjà été entrevue lors de l'extension à la programmation semi-définie positive du lemme de Farkas.

**Lemme 3.4 (Slater)** *Si il existe  $y \in \mathbb{R}^m$  tel que  $\sum_{i=1}^m y_i A_i \succ 0$ , alors le programme  $Sdp$  a un optimum borné.*

Preuve :

Supposons qu'il existe une solution strictement réalisable du programme dual  $Dsdp$  :  $S = \sum_{i=1}^m y_i A_i - C \succ 0$ . Considérons une solution  $X$  de  $Sdp$  telle que  $C \cdot X \geq v$ , on a alors :

$$X \cdot S = y^t b - C \cdot X \leq y^t b - v$$

Soit  $\lambda_{min}$  la plus petite des valeurs propres de  $S$ , étant donné que  $S \succ 0$  on peut dire que  $\lambda_{min} > 0$  et que  $S \succeq \lambda_{min} Id$ . Aussi peut-on écrire que :

$$\lambda_{min} X \cdot Id \leq X \cdot S \leq y^t b - v \iff X \cdot Id \leq \frac{y^t b - v}{\lambda_{min}}$$

La somme des termes diagonaux de  $X$  est donc bornée, aussi peut-on affirmer que toutes les solutions de  $Sdp$  sont bornées. ■

Nous allons maintenant établir dans ce contexte les relations de dualité qui lient les programmes  $Sdp$  et  $Dsdp$ .

**Théorème 3.4 (Dualité faible)** *Supposons que les deux programmes semi-définis positifs ci-dessus aient des solutions réalisables, alors :*

$$C \cdot X \leq y^t b$$

Preuve :

Par le lemme de Farkas on sait que le programme  $Sdp$  admet une solution si et seulement si  $y^t b \geq 0$  pour tout vecteur  $y$  tel que  $\sum_{i=1}^m y_i A_i \succeq 0$ . Par hypothèse  $Sdp$  admet une solution alors :

$$y^t b = \left( \sum_{i=1}^m y_i A_i \right) \cdot X \geq 0$$

Par ailleurs dire que  $\sum_{i=1}^m y_i A_i - C \succeq 0$  c'est dire que  $\left( \sum_{i=1}^m y_i A_i - C \right) \cdot X \geq 0, \forall X \in U_{psd}$  et donc en particulier pour les  $X$  solution de  $Sdp$ , on peut alors écrire :

$$\left. \begin{array}{l} y^t b = \left( \sum_{i=1}^m y_i A_i \right) \cdot X \geq 0 \\ \left( \sum_{i=1}^m y_i A_i - C \right) \cdot X \geq 0 \end{array} \right\} \implies y^t b \geq C \cdot X$$

Nous allons voir que sous l'hypothèse que le primal et le dual soient tous deux définis sur un cône fermé, il existe un théorème de dualité forte en programmation semi-définie positive. ■

**Théorème 3.5 (Dualité forte)** Soit  $z_{max}$  et  $z_{min}$  les optima respectifs de  $Sdp$  et  $Dsdp$ . Supposons qu'il existe un vecteur  $y \in \mathbb{R}^m$  tel que  $\sum_{i=1}^m y_i A_i \succ 0$ , alors  $z_{max} = z_{min}$ .

Preuve :

Supposons que  $z_{max} < z_{min}$ , alors le système :

$$\begin{aligned} \text{Max } C \cdot X &= z_{min} \\ A_i \cdot X &= b_i \quad \forall i \in \{1, \dots, m\} \\ X &\in U_{psd} \end{aligned}$$

n'a pas de solution. Aussi le lemme de Farkas étendu à la programmation semi-définie positive nous permet de dire qu'il existe un vecteur  $(y_0, y) \in \mathbb{R}^{m+1}$  tel que :

$$y_0 C + \sum_{i=1}^m y_i A_i \succeq 0 \quad \text{et} \quad y_0 z_{min} + y^t b < 0 \quad (3.7)$$

Si  $y_0 = 0$ , alors 3.7 est équivalent à  $\sum_{i=1}^m y_i A_i \succeq 0$  et  $y^t b < 0$ . Ce qui signifie que  $Sdp$  n'a pas de solution, ce qui est contraire aux hypothèses.

Si  $y_0 > 0$  on a  $C + \frac{1}{y_0} \sum_{i=1}^m y_i A_i \succeq 0$  et  $z_{min} + \frac{1}{y_0} (y^t b) < 0$ , ce qui signifie que  $z_{min}$  n'est pas solution optimale de  $Dsdp$ .

Si  $y_0 < 0$  alors en divisant 3.7 par  $-y_0$  on obtient :

$$-C - \frac{1}{y_0} \sum_{i=1}^m y_i A_i \succeq 0 \quad \text{et} \quad -z_{min} - \frac{1}{y_0} (y^t b) < 0$$

L'inégalité stricte nous permet d'affirmer qu'il existe  $\epsilon > 0$  tel que :

$$-C - \frac{1}{y_0} \sum_{i=1}^m y_i A_i \succeq 0 \quad \text{et} \quad -z_{min} - \frac{1}{y_0} (y^t b) < -\epsilon \quad (3.8)$$

Par hypothèse  $z_{min}$  est la valeur optimale de  $Dsdp$  aussi il existe  $y^{min}$  tel que :

$$C - \sum_{i=1}^m y_i^{min} A_i \succeq 0 \quad \text{et} \quad z_{min} - b^t y^{min} < \epsilon \quad (3.9)$$

En additionnant 3.8 et 3.9 on obtient :

$$-\sum_{i=1}^m \left( \frac{y_i}{y_0} + y_i^{min} \right) A_i \succeq 0 \quad \text{et} \quad -\left( \frac{y}{y_0} + y^{min} \right)^t b < 0$$

ce qui, avec le lemme de Farkas, permet de dire que  $Sdp$  n'a pas de solution. L'hypothèse  $z_{max} < z_{min}$  conduit à une contradiction et comme par le théorème de dualité faible  $z_{max} \leq z_{min}$  on peut conclure que  $z_{max} = z_{min}$ . ■

### 3.6 Conditions d'écart complémentaires

Le théorème de dualité forte vu précédemment nous permet d'écrire que si  $X$  et  $y$  sont des optima respectivement de  $Sdp$  et  $Dsdp$  :

$$C \cdot X = \sum_{i=1}^m y_i A_i \cdot X = y^t b \quad (3.10)$$

Les conditions d'écart complémentaires peuvent prendre deux formes :

1.  $\sum_{i=1}^m y_i (A_i \cdot X - b_i) = 0$ , nous retrouvons ainsi une condition d'écart complémentaires sous forme classique, une variable strictement positive de  $Dsdp$  correspond à une contrainte serrée de  $Sdp$ .
2.  $(\sum_{i=1}^m y_i A_i - C) \cdot X = 0$ .

### 3.7 Dualité étendue en programmation semi-définie positive

Nous avons vu précédemment que l'obtention d'un théorème de dualité forte suppose l'observation de condition de régularité, condition de Slater, ceci est induit par la définition du programme dual. L'exemple suivant montre qu'avec cette définition de la dualité il peut exister un saut de dualité bien que les optima respectifs du primal et du dual soient bornés et atteints. Considérons le programme semi-défini positif suivant :

$$\begin{array}{l} \text{Max} \\ \left( \begin{array}{ccc} x_2 & & \\ \alpha - x_2 & 0 & 0 \\ 0 & -x_1 & -x_2 \\ 0 & -x_2 & 0 \end{array} \right) \succeq 0 \end{array}$$

En utilisant la propriété de diagonale dominante des matrices semi-définies positives on peut voir que la solution optimale de ce programme est 0. Son programme dual s'énonce comme suit :

$$\begin{array}{l} \text{Min} \quad \alpha u_1 \\ \text{s.c} \quad u_{2,2} = 0 \\ \quad \quad u_{11} + 2u_{23} = 1 \\ \quad \quad U \succeq 0 \end{array}$$

Etant donné que  $u_{22} = 0$  par la diagonale dominante des matrices semi-définies positives on a  $u_{23} = 0$  et donc  $u_{11} = 1$ , aussi la valeur optimale du programme dual est  $\alpha$ . On voit donc que si la condition de Slater n'est pas vérifiée on peut avoir un saut de dualité non nul bien que le primal et le dual aient des optima respectifs bornés et atteints. Pour obvier cet inconvénient V. Ramana, [77], a défini un programme dual étendu de la manière suivante. Considérons un programme primal :

$$\begin{array}{l} \text{Max} \quad c^t x \\ \text{s.c} \quad \sum_{i=1}^m x_i A_i \preceq A_0 \end{array}$$



Son programme dual est alors défini par :

$$\begin{aligned}
\text{Min} \quad & A_0 \cdot (U + W_m) \\
\text{s.c.} \quad & A_i \cdot (U + W_m) = c_i, \forall i \in \{1, \dots, m\} \\
& A_i \cdot (U_j + W_{j-1}) = 0, \forall i \in \{0, \dots, m\}, \forall j \in \{1, \dots, m\} \\
& U_j \succeq W_j W_j^t, \forall j \in \{1, \dots, m\} \\
& U \succeq 0 \\
& W_0 = 0
\end{aligned}$$

L'auteur montre que si 0 est réalisable pour le programme primal on a  $A_0 \cdot U_j = 0$  et  $A_0 \cdot W_j = 0, \forall j \in \{1, \dots, m\}$ , et  $(A_0 - \sum_{i=1}^m x_i A_i) \cdot U_j = 0$  et  $(A_0 - \sum_{i=1}^m x_i A_i) \cdot W_j = 0$ . Aussi la contribution de la matrice  $W_m$  à la fonction objective est nulle, elle va jouer le rôle de matrice d'écart, absorbant le saut de dualité. Si nous écrivons le programme dual étendu dans le cas de l'exemple précédent nous obtenons :

$$\begin{aligned}
\text{Min} \quad & A_0 \cdot (U + W_2) \\
\text{s.c.} \quad & A_1 \cdot (U + W_2) = 0 \\
& A_2 \cdot (U + W_2) = 1 \\
& A_0 \cdot (U_2 + W_1) = 0 \\
& A_1 \cdot (U_2 + W_1) = 0 \\
& A_2 \cdot (U_2 + W_1) = 0 \\
& A_0 \cdot U_1 = 0 \\
& A_1 \cdot U_1 = 0 \\
& A_2 \cdot U_1 = 0 \\
& U_1 \succeq W_1 W_1^t \\
& U_2 \succeq W_2 W_2^t \\
& U \succeq 0
\end{aligned}$$

Avec :

$$A_0 = \begin{pmatrix} \alpha & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, A_1 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \text{ et } A_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

On peut voir que  $U = U_2 = W_1 = 0, U_2 = e_3 e_3^t$  et  $W_2 = e_3 e_2^t$  est une solution du programme dual, la valeur de la fonction objective est alors nulle, aussi par le théorème de dualité faible nous pouvons affirmer que cette solution est optimale. Le théorème de dualité obtenu par V. Ramana en définissant ainsi le programme dual est le suivant :

### **Théorème 3.6 (Ramana)**

1. Si  $x$  est primal réalisable et si  $(U, W)$  est dual réalisable alors :

$$c^t x \leq A_0 \cdot (U + W).$$

2. Si le problème primal admet une solution réalisable alors sa valeur optimale est bornée si et seulement si le problème dual admet une solution réalisable.
3. Si le problème primal et le problème dual admettent tous deux une solution réalisable alors leurs valeurs optimales sont égales.

4. Si les valeurs optimales tant du problème primal que du problème dual sont finies alors ce dernier atteint sa valeur optimale.

Cette définition de la dualité forte ainsi que les limites de la dualité lagrangienne sont aussi exposées dans [78]. Ainsi possède-t-on d'une théorie de dualité en programmation semi-définie positive tout à fait semblable à celle que nous connaissons en programmation linéaire.

### 3.8 Programmation mathématique et formes hermitiennes

Nous allons voir comment la programmation semi-définie positive s'étend sur les espaces vectoriels dont le corps de référence est  $\mathbb{C}$  le corps des nombres complexes. M. Goemans et D. Williamson ont montré dans [41] comment utiliser le fait que l'unité possède  $n$  racines  $n^{\text{ième}}$  distinctes dans  $\mathbb{C}$  pour modéliser des problèmes d'optimisation combinatoire dont la solution peut présenter  $n$  choix, par exemple la  $n$ -coupe de poids maximum d'un graphe. Cette application sera développée au chapitre suivant.

**Définition 3.3** Soit  $E$  un espace vectoriel dont le corps de référence est  $\mathbb{C}$  et  $h$  une forme bilinéaire,  $h$  est dite sesquilinéaire si elle vérifie :

- $\forall x \in E \ h(x, \lambda_1 y_1 + \lambda_2 y_2) = \lambda_1 h(x, y_1) + \lambda_2 h(x, y_2), \forall y_1, y_2 \in E, \forall \lambda_1, \lambda_2 \in \mathbb{C}$  ;
- $\forall x_1, x_2 \in E, \forall \lambda_1, \lambda_2 \in \mathbb{C} \ h(\lambda_1 x_1 + \lambda_2 x_2, y) = \bar{\lambda}_1 h(x_1, y) + \bar{\lambda}_2 h(x_2, y)$ .

Une forme sesquilinéaire  $h$  est dite hermitienne si  $\forall (x, y) \in E \times E$  on a  $h(x, y) = \bar{h}(y, x)$ .

Associés à une forme hermitienne  $h$  définie sur  $E$  un espace vectoriel de dimension  $n$  une matrice  $n \times n$   $A = (a_{kl})$ , de la définition de forme hermitienne on déduit aisément que :

- $A = \bar{A}^t = (\bar{a}_{kl})^t$  ;
- $A = Re(A) + Im(A)$  où  $Re(A) = (Re \ a_{kl})$  est une matrice symétrique et  $Im(A) = (Im \ a_{kl})$  est une matrice antisymétrique.

Une matrice  $A$  associée à une forme hermitienne sera dite matrice hermitienne, on notera  $H_n$  l'espace des matrices hermitiennes. Une forme hermitienne  $h$  est dite définie si  $h(x, \bar{x}) = 0 \implies x = 0$ , elle est dite positive si  $\forall x \in E \ h(x, \bar{x}) \geq 0$ .

**Lemme 3.5** Soit  $A \in H_n$  la matrice associée à la forme hermitienne  $h$  alors la forme quadratique  $h(x, \bar{x}) = \bar{x}Ax$  a ses valeurs dans  $\mathbb{R}$ . De plus toutes les valeurs propres de la matrice  $A$  sont réelles.

Preuve :

Etant donné que  $A \in H_n$  on peut écrire  $A = Re(A) + Im(A)$ ,  $Re(A)$  étant symétrique et  $Im(A)$  étant antisymétrique. La forme quadratique peut se réécrire de la manière suivante :

$$\bar{x}Ax = \bar{x}Re(A)x + \bar{x}Im(A)x \in \mathbb{R}$$

En écrivant  $A$  dans la base de ses vecteurs propres on obtient  $A = P^{-1}Diag(\Lambda)P$ ,  $P$  étant une matrice orthogonale et  $\Lambda$  le vecteurs des valeurs propres de  $A$ . Aussi la forme quadratique associée à  $h$  peut s'écrire  $\bar{x}Ax = \bar{x}P^{-1}Diag(\Lambda)Px$ , ce qui permet de conclure que toutes les valeurs propres de  $A$  sont réelles.

■

**Corollaire 3.1** *Soit  $h$  une forme hermitienne définie positive et  $A$  la matrice qui lui est associée, alors toutes les valeurs propres de  $A$  sont strictement positives.  $A$  est alors dite définie positive.*

Les formes hermitiennes induisent un produit scalaire défini par  $u \cdot v = h(u, \bar{v}) = \bar{v}Au$ , en particulier si  $A = Id$  on obtient  $u \cdot v = \bar{v}^t u$ . Par ailleurs on peut voir que si  $A \in H_n$  les quatre propositions suivantes sont équivalentes :

- $\bar{x}^t Ax \geq 0, \forall x \in \mathbb{C}$  ;
- toutes les valeurs propres de  $A$  sont réelles et positives ou nulles ;
- il existe une matrice  $B \in H_n$  telle que  $A = B\bar{B}^t$  ;
- il existe  $n$  vecteurs  $v_k \in \mathbb{C}^n$  tels que  $a_{kl} = v_k \cdot v_l$ .

Une telle matrice  $A \in H_n$  est dite semi-définie positive. Ainsi retrouve-t-on des propriétés semblables à celles vues pour les matrices semi-définies positives à coefficients réels. On peut maintenant définir un programme semi-défini positif sur l'espace des matrices hermitiennes. Soit  $C \in H_n$  et  $m$  matrices  $A_j$  et un vecteur  $b \in \mathbb{C}$ , un programme semi-défini positif sur l'espace  $H_n$  est de la forme :

$$Csdp = \begin{cases} \text{Max} & C \cdot X \\ \text{s.t.} & A_j \cdot X \leq b_j \\ & X \in H_n \\ & X \succeq 0 \end{cases}$$

Nous allons voir maintenant qu'un tel programme se réduit polynomialement à un programme semi-défini positif défini sur les matrices symétriques à coefficient réels et donc qu'il peut se résoudre avec un facteur d'erreur  $\epsilon$  en temps polynomial de la dimension de l'espace sur lequel il est défini.

Considérons la transformation  $\tau$  définie comme suit :

$$\begin{aligned} \tau : H_n &\longrightarrow S_{2n} \\ X &\longrightarrow \tau(X) = \begin{pmatrix} \text{Re}(X) & -\text{Im}(X) \\ \text{Im}(X) & \text{Re}(X) \end{pmatrix} \end{aligned}$$

où  $S_{2n}$  désigne l'espace des matrices symétriques d'ordre  $2n$ . On peut remarquer que :

- $\tau(X)$  est une matrice symétrique car  $\text{Re}(X)$  est une matrice symétrique et  $\text{Im}(X)$  est une matrice antisymétrique ;
- la transformation  $\tau$  est linéaire.

**Lemme 3.6**  $\tau(A) \cdot \tau(B) = 2A \cdot B$

Preuve :

$$\begin{aligned}
\tau(A) \cdot \tau(B) &= \text{tr}(\tau(B)^t \tau(A)) \\
&= \text{tr} \left( \begin{pmatrix} \text{Re}(B) & \text{Im}(B) \\ -\text{Im}(B) & \text{Re}(B) \end{pmatrix} \begin{pmatrix} \text{Re}(A) & -\text{Im}(A) \\ \text{Im}(A) & \text{Re}(A) \end{pmatrix} \right) \\
&= 2\text{tr}(\text{Re}(B)\text{Re}(A) + \text{Im}(B)\text{Im}(A)) \\
&= 2\text{tr}(B^t A) \\
&= 2A \cdot B
\end{aligned}$$

■

Considérons maintenant le programme défini sur  $S_{2n}$  de la manière suivante :

$$Sdp = \begin{cases} \text{Max} & \tau(C) \cdot Y \\ \text{s.t.} & \tau(A_j) \cdot Y \leq 2b_j \quad \forall j \in \{1, \dots, m\} \\ & \begin{pmatrix} E_{kl} & 0 \\ 0 & -E_{kl} \end{pmatrix} \cdot Y = 0 \quad \forall k, l \in \{1, \dots, n\} \\ & \begin{pmatrix} 0 & E_{kl} \\ E_{kl} & 0 \end{pmatrix} \cdot Y = 0 \quad \forall k, l \in \{1, \dots, n\} \\ & Y \in S_{2n} \\ & Y \succeq 0 \end{cases}$$

**Théorème 3.7** Soit  $X$  une solution du programme  $Csdp$ , alors  $\tau(X)$  est une solution du programme  $Sdp$ , de plus la valeur de la fonction objective de  $Sdp$  est alors exactement le double de valeur de la fonction objective de  $Csdp$ .

Réciproquement si  $Y$  est une solution de  $Sdp$ , alors  $\tau^{-1}(Y)$  est une solution de  $Csdp$ , de plus la valeur de la fonction objective de  $Csdp$  est alors exactement la moitié de la valeur de la fonction objective de  $Sdp$ .

Preuve :

- $\tau(A_j) \cdot \tau(X) = 2A_j \cdot X \leq 2b_j$  ;
- $\tau(C) \cdot \tau(X) = 2C \cdot X$ .
- $X \in H_n \implies X = \text{Re}(X) + i\text{Im}(X)$ , ainsi étant donné que  $\text{Re}(X) = \text{Re}(X)^t$ ,  $\text{Im}(Y) = -\text{Im}(Y)^t$  et que  $\tau$  est linéaire les 2<sup>ieme</sup> et 3<sup>ieme</sup> contraintes sont satisfaites ;
- par construction  $\tau(X) \in S_{2n}$  ;
- soit  $x_\lambda$  le vecteur propre de  $X$  associé à la valeur propre  $\lambda$ , alors  $y = (\text{Re}(x), \text{Im}(x))^t$  est le vecteur propre de  $Y = \tau(X)$  associé à la valeur propre  $\lambda$ . Par conséquent si les valeurs propres de  $X$  sont positives ou nulles, celles de  $Y$  le sont aussi.

■

Ainsi nous avons montré que  $Csdp$  se réduit polynomialement à  $Sdp$ .

Ce paragraphe dédié à la programmation mathématique sur le cône des formes hermitiennes semi-définies positives conclut ce tour d'horizon relatif à la programmation semi-définie positive, dont nous verrons quelques applications au chapitre suivant.

## Chapitre 4

# Applications de la programmation semi-définie positive

Dans ce chapitre nous présentons quatre applications de la programmation semi-définie positive à des problèmes d'optimisation combinatoire. Ces quatre applications donnent un tour d'horizon assez complet des possibilités qu'offre la programmation semi-définie positive pour aborder des problèmes combinatoires. La première application présentée est celle qui fut proposée par L. Lovász en 1979 et consiste en la définition d'une relaxation semi-définie positive du problème du stable de poids maximum d'un graphe. Ce travail est un travail fondateur car il est le premier où est montré ce que la programmation semi-définie positive peut apporter : renforcements naturels des relaxations des problèmes pour lesquels la relaxation linéaire donne de mauvais résultats et associations d'espaces métriques aux problèmes d'optimisation combinatoire, par étiquetage des sommets du graphe dans le cas du stable. La deuxième application est due à M. Goemans et D. Williamson, elle reprend la technique d'étiquetage des sommets du graphe grâce à la factorisation comme matrice de Gram d'une matrice semi-définie positive, ceci permettant d'associer un espace métrique au problème de la coupe de poids maximum. Après avoir plongé ce dernier espace dans un espace probabilisable M. Goemans et D. Williamson obtiennent un algorithme d'approximation, le meilleur connu à ce jour, pour le problème de la coupe de poids maximum. Les problèmes de définition d'espace de petite dimension pour la factorisation des matrices semi-définies positives comme matrice de Gram sont peu abordés dans ce document, nous y reviendrons dans le chapitre de conclusion car ils nous semblent procurer des voies de prospection intéressantes. Nous présentons aussi une application de la programmation semi-définie positive sur les espaces complexes, celle-ci est aussi due à M. Goemans et D. Williamson.

Enfin nous présentons une quatrième application de la programmation semi-définie qui est le fruit d'un travail que nous réalisons avec J. Galtier. Cette application est de nature assez différentes des trois premières car elle ne consiste ni en la définition de relaxation ni en la recherche d'étiquetage. Une pondération des arêtes d'un graphe étant donnée par une fonction convexe on remplace cette fonction par une matrice semi-définie positive. Cette substitution, qui peut paraître contre nature, présente un avantage lorsque la fonction de pondération présente une asymptote verticale. De plus ayant appliqué cette représentation pour un problème de flot avec contraintes sur la somme des poids des arcs des chemins support du flot, nous avons mis au point une technique de génération de colonnes

sur le cône des matrices semi-définies positives, généralisant la technique bien connue en programmation linéaire.

## 4.1 Le problème du stable de cardinalité maximum

Le problème du stable de cardinalité maximum d'un graphe est à l'origine des applications de la programmation semi-définie positive à l'optimisation combinatoire. Soit  $G = (X, E)$  un graphe non orienté où  $X$  désigne l'ensemble des sommets et  $E$  celui des arêtes. On appelle ensemble stable du graphe  $G$  tout sous-ensemble de sommets  $S$  tel que  $\forall (s_1, s_2) \in S \times S$  l'arête  $(s_1 s_2)$  n'appartient pas à  $E$ . Le complémentaire d'un stable est donc une clique. On voit donc que tous les sommets d'un stable peuvent être coloriés avec une même couleur, c'est à dire que si les sommets du graphe figurent les cellules d'un réseau toutes celles correspondant à un stable peuvent recevoir la même allocation de fréquence.

Le problème du stable de cardinalité maximum, qui est connu pour appartenir à la classe des problèmes  $\mathcal{NP}$ difficiles, peut se formuler comme un programme linéaire en nombres entiers de la manière suivante :

$$Stable_{max} = \begin{cases} Max & \mathbf{1}^t x \\ & x_i + x_j \leq 1 \quad \forall (ij) \in E \\ & x_i \in \{0, 1\} \quad \forall i \in X \end{cases}$$

où  $\mathbf{1}$  désigne le vecteur de dimension  $n = |X|$  dont toutes les composantes sont égales à 1. La relaxation linéaire obtenue à partir de cette formulation du problème est connue comme étant assez mauvaise. On note  $\alpha(G)$  la valeur d'une solution optimale de  $Stable_{max}$ . Dans [60] Lovász a défini un majorant,  $\theta(G)$ , de  $\alpha(G)$ . Ce majorant peut se calculer en résolvant un programme semi-défini positif.

Soit  $P$  l'ensemble des matrices symétriques  $A$  telles que  $a_{ij} = 1$  si  $(ij) \notin E$  ou  $i = j$ . Notons  $J_k$  la matrice carrée d'ordre  $k$  dont tous les coefficients valent 1,  $J_n = J$ . Soit  $E_{ij}$  la matrice carrée symétrique d'ordre  $n$  dont tous les coefficients sont nuls sauf  $e_{ij} = e_{ji}$ . On remarque alors que :

$$A \in P \iff A = J - \sum_{(ij) \in E} x_{ij} E_{ij}$$

On appelle mineur d'une matrice  $A$  la sous matrice obtenue en supprimant des lignes de  $A$  et les colonnes de même indice. Nous rappelons sans le démontrer le résultat suivant sur l'entrelacement des valeurs propres d'une matrice et les valeurs propres de ses mineurs.

**Théorème 4.1** *Soit  $A$  une matrice symétrique d'ordre  $n$  dont les valeurs propres vérifient  $\lambda_1 \leq \dots \leq \lambda_n$ . Soit  $B$  un mineur symétrique d'ordre  $k$  de  $A$  dont les valeurs propres vérifient  $\mu_1 \leq \dots \leq \mu_k$ , alors :  $\lambda_i \leq \mu_i \leq \lambda_{i+n-k}$ .*

**Corollaire 4.1 (Lovász)**

*Si le graphe  $G = (X, E)$  admet un stable de taille  $k$  alors :*

$$\lambda_{max}(A) \geq \lambda_{max}(J_k) = k$$

où  $\lambda_{max}(M)$  désigne la plus grande des valeurs propres de la matrice  $M$ .

Preuve :

Si le graphe  $G$  contient un stable de taille  $k$  alors toute matrice  $A \in P$  contient  $J_k$  comme mineur, aussi d'après le théorème précédent  $\lambda_{max}(A) \geq \lambda_{max}(J_k)$ ,  $\forall A \in P$ . D'autre part  $J_k$  étant la matrice carrée d'ordre  $k$  dont tous les coefficients valent 1, on a  $\lambda_{max}(J_k) = k$  ■

On peut donc affirmer que  $Min\{\lambda_{max}(A) \mid A \in P\} \geq \alpha(G)$ . Ceci est une formulation du paramètre  $\theta(G)$ , nous allons voir comment elle conduit à un programme semi-défini positif. Soit  $\{\lambda_i, i = 1, \dots, n\}$  les valeurs propres d'une matrice  $A$ . Chercher  $\lambda_{max}(A) = Max\{\lambda_i, i = 1, \dots, n\}$  est équivalent à chercher le plus petit nombre  $t$  tel que :

$$t \geq \lambda_i \iff t - \lambda_i \geq 0, \quad \forall i \in \{1, \dots, n\}$$

Or  $t - \lambda_i, i \in \{1, \dots, n\}$  sont les valeurs propres de la matrice  $tId - A$  si  $Id$  désigne la matrice identité, et dire que toutes les valeurs propres d'une matrice sont positives ou nulles est équivalent à dire que cette matrice est semi-définie positive. On peut donc écrire que :

$$\lambda_{max}(A) = Min\{t \mid tId - A \succeq 0\}$$

On a vu précédemment que  $A \in P \iff A = J - \sum_{(ij) \in E} x_{ij} E_{ij}$ , on peut donc formuler  $\theta(G)$  comme le programme semi-défini positif suivant :

$$\theta(G) = Min \quad t$$

$$tId + \sum_{(ij) \in E} x_{ij} E_{ij} - J \succeq 0$$

En appliquant le théorème de dualité forte vu au chapitre précédent on peut écrire :

$$\theta(G) = Max \quad J \cdot Y$$

$$y_{ij} = 0 \quad \forall (ij) \in E$$

$$tr(Y) = 1$$

$$Y \succeq 0$$

La matrice  $Y$  recherchée étant semi-définie positive elle peut se décomposer comme matrice de Gram. Il existe donc  $n$  vecteurs  $v_i$  tels que  $y_{ij} = v_i^t v_j$ . De plus on veut  $y_{ij} = v_i^t v_j = 0, \forall (ij) \in E$ . Si maintenant on définit les labels  $v_i$  non plus sur  $G$  mais sur son complémentaire  $\bar{G} = (X, \bar{E})$  on est amené à chercher des labels orthogonaux  $v_i$  et  $v_j, \forall (ij) \notin \bar{E}$ . Ainsi on trouve la deuxième définition de  $\theta(G)$  donnée par Lovász, pour plus de détails voir [52].

## 4.2 Le problème de la coupe de poids maximum

Le problème de la coupe de poids maximum intervient notamment dans la génération de contraintes pour la résolution du programme maître associé à une décomposition de Benders d'un problème multiflot, [8]. Nous allons voir un algorithme d'approximation basé sur une relaxation semi-définie positive pour ce problème, cet algorithme est dû à M. Goemans et D. Williamson,[40].

Soit  $G = (X, E)$  un graphe non orienté et  $w$  une pondération des arêtes de ce graphe que nous supposons positive dans un premier temps. Le problème de la coupe de poids

maximum est défini par :  $Max\{ \sum_{(ij) \in \delta(S)} w_{ij}; S \subset X, S \neq V\}$ , où  $\delta(S)$  est le cocycle de

l'ensemble de sommets  $S$ . Ce problème est  $\mathcal{NP}$ difficile. Le problème de la coupe de poids maximum peut se reformuler comme un programme quadratique en nombres entiers de la manière suivante :

$$Qcoupe = \begin{cases} Max & \frac{1}{2} \sum_{(ij) \in E} w_{ij}(1 - y_i y_j) \\ s.c. & y_i \in \{-1, +1\}, \forall i \in X \end{cases} \quad (4.1)$$

On définit alors le support d'une coupe  $S = \{i \in X \mid y_i = 1\}$ . Considérons la matrice  $Y = [y_{ij}] = [y_i y_j]$ , cette matrice est semi-définie positive. Les termes diagonaux de  $Y$  sont égaux à 1 et son rang est 1. En effet considérons une ligne  $y_i$ ,  $i \geq 2$  de  $Y$ , alors si le sommet  $i$  appartient au même support de coupe que le sommet 1 la ligne  $y_i$  est égale à la ligne  $y_1$ ; par contre si le sommet  $i$  appartient au complémentaire dans  $X$  du support de coupe contenant le sommet 1 on a  $y_i = -y_1$ . En relâchant la contrainte sur le rang de la matrice  $Y$  on peut définir un programme semi-défini positif fournissant un majorant pour le problème de la coupe de poids maximum. Si on note  $W$  la matrice des poids des arêtes,  $W = [w_{ij}]$ , on peut écrire ce programme comme suit :

$$SDcoupe = \begin{cases} Max & \frac{1}{4}W \cdot (J - Y) \\ s.c. & \begin{aligned} diag(Y) &= \mathbf{1} \\ Y &\succeq 0 \end{aligned} \end{cases} \quad (4.2)$$

Etant donné que la matrice  $Y$  est semi-définie positive elle peut se décomposer comme une matrice de Gram, c'est à dire qu'il existe  $n$  vecteurs  $v_i \in \mathbb{R}^n$ ,  $i \in \{1, \dots, n\}$  tels que  $y_{ij} = v_i^t v_j$ , de plus les termes diagonaux de  $Y$  sont égaux à 1 ce qui implique que  $\|v_i\| = 1, \forall i \in \{1, \dots, n\}$ . Si  $S_p^n$  est la sphère unité de  $\mathbb{R}^n$  la relaxation semi-définie positive du problème de la coupe de poids maximum peut se reformuler de manière suivante :

$$P = \begin{cases} Max & \frac{1}{2} \sum_{(ij) \in E} w_{ij}(1 - v_i^t v_j) \\ s.c. & v_i \in S_p^n \end{cases} \quad (4.3)$$

#### 4.2.1 L'algorithme de Goemans et Williamson

L'algorithme mis au point par Goemans et Williamson se déroule en trois étapes :

1. Résolution du programme semi-défini positif 4.3 de manière à obtenir un ensemble de vecteurs optimaux  $v_i$ .
2. Choix d'un vecteur  $r$  uniformément distribué sur la sphère  $S_p^n$ .
3. Le support de coupe  $S$  est alors défini par  $S = \{i \in X \mid v_i^t r \geq 0\}$

**Remarque :** Pour résoudre le programme 4.3 on va en fait procéder en deux étapes :

1. Tout d'abord résoudre d'une manière approchée le programme 4.2 en utilisant, par exemple, un algorithme de points intérieurs pour la programmation semi-définie positive. Ceci nous assure de trouver une valeur supérieure à  $Z_p^* - \epsilon$ ,  $Z_p^*$  étant la valeur optimale des programmes 4.3 et 4.2, en un temps polynomial de la taille de l'instance et de  $\log \frac{1}{\epsilon}$ .



2. Une fois qu'une solution quasi optimale de 4.2 a été trouvée on peut obtenir les vecteurs  $v_i$ ,  $i \in \{1, \dots, n\}$  en calculant une décomposition de Cholesky de la matrice  $\bar{Y}$  quasi optimale pour 4.2 de manière à ce que  $\frac{1}{2} \sum_{(ij) \in E} w_{ij}(1 - v_i^t v_j) \geq Z_p^* - \epsilon$ .

Nous allons maintenant analyser les performances de l'algorithme aléatoire et plus précisément déterminer quelle est l'espérance mathématique de la valeur des coupes obtenues ainsi.

Soit  $\{v_1, \dots, v_n\}$  des vecteurs de la sphère unité  $S_p^n$ . On note  $E[W]$  l'espérance mathématique de la valeur des coupes  $\delta(S)$  produites par l'algorithme aléatoire. On peut noter que  $E[W]$  est égale à la somme des poids des arêtes multipliés par la probabilité que les deux extrémités d'une arête ne soient pas du même côté de la coupe. C'est à dire par définition de  $S$  la probabilité que  $v_i^t r$  et  $v_j^t r$  ne soient pas de même signe.

**Lemme 4.1 (Goemans et Williamson)**

La probabilité que  $v_i^t r$  et  $v_j^t r$  ne soient pas de même signe est égale à :  $\frac{1}{\pi} \arccos(v_i^t v_j)$

Preuve :

Par symétrie la probabilité que  $v_i^t r$  et  $v_j^t r$  ne soient pas de même signe vaut deux fois  $P(v_i^t r \geq 0 \text{ et } v_j^t r < 0)$ . Si on pose  $\theta = \arccos(v_i^t v_j)$  on peut voir que l'intersection de l'ensemble  $\{r \mid v_i^t r \geq 0 \text{ et } v_j^t r < 0\}$  et de la sphère  $S_p^n$  est un digone d'angle  $\theta$ . Aussi la portion de sphère ainsi déterminée est  $\frac{\theta}{2\pi}$ , donc  $P(v_i^t r \geq 0 \text{ et } v_j^t r < 0) = \frac{\theta}{2\pi}$ , d'où le lemme. ■

Le lemme précédent et la définition de l'espérance mathématique nous permettent d'énoncer le corollaire suivant.

**Corollaire 4.2 (Goemans et Williamson)**

$$E[W] = \frac{1}{\pi} \sum_{(ij) \in E} w_{ij} \arccos(v_i^t v_j).$$

On définit  $\alpha = \text{Min}\left\{\frac{2}{\pi} \frac{\theta}{1 - \cos(\theta)}\right\} > 0.87856$ . C'est cette constante  $\alpha$  qui va donner une borne pour l'algorithme.

**Lemme 4.2**  $\forall y \in [-1, +1]$  on a :  $\frac{1}{\pi} \arccos(y) \geq \frac{\alpha}{2}(1 - y)$ .

Preuve :

Immédiate si on pose  $y = \cos(\theta)$ . ■

Le théorème suivant est le résultat important qui est exposé dans [40].

**Théorème 4.2 (Goemans et Williamson)**

$$E[W] \geq \frac{\alpha}{2} \sum_{(ij) \in E} w_{ij}(1 - v_i^t v_j).$$

Preuve :

$$E[W] = \sum_{(ij) \in E} \frac{w_{ij} \arccos(v_i^t v_j)}{\pi} \geq \frac{\alpha}{2} \sum_{(ij) \in E} w_{ij}(1 - v_i^t v_j).$$

Nous allons maintenant voir ce que devient la borne du théorème précédent si les poids

des arêtes sont quelconques. Remarquons tout d'abord un fait similaire à celui du lemme 4.2 : pour tout  $z \in [-1, +1]$  on a  $1 - \frac{1}{\pi} \arccos(z) \geq \frac{\alpha}{2}(1+z)$  ce qui se déduit en remplaçant dans le lemme 4.2  $y$  par  $-z$  et en notant que  $\cos(\pi - \theta) = -\cos(\theta)$ .

**Théorème 4.3 (Goemans et Williamson)**

Soit  $W^- = \sum_{(ij) \in E} w_{ij}^-$  où  $w_{ij}^- = \text{Min}\{0, w_{ij}\}$ . Alors :

$$E[W] - W^- \geq \frac{\alpha}{2} \left( \sum_{(ij) \in E} w_{ij}(1 - v_i^t v_j) - W^- \right)$$

Preuve :

La quantité  $E[W] - W^-$  peut se réécrire de la manière suivante :

$$\sum_{\substack{(ij) \in E \\ w_{ij} > 0}} w_{ij} \frac{\arccos(v_i^t v_j)}{\pi} + \sum_{\substack{(ij) \in E \\ w_{ij} < 0}} |w_{ij}| \left(1 - \frac{\arccos(v_i^t v_j)}{\pi}\right)$$

D'une manière similaire  $\frac{1}{2} \left( \sum_{(ij) \in E} w_{ij}(1 - v_i^t v_j) - W^- \right)$  peut se réécrire :

$$\sum_{\substack{(ij) \in E \\ w_{ij} > 0}} w_{ij} \frac{1 - v_i^t v_j}{2} + \sum_{\substack{(ij) \in E \\ w_{ij} < 0}} |w_{ij}| \frac{1 + v_i^t v_j}{2}$$

Le résultat du théorème se déduit alors en identifiant membre à membre les termes de l'inégalité et en appliquant la remarque faite avant son énoncé. ■

Une conséquence importante de l'analyse qui vient d'être faite est le théorème suivant.

**Théorème 4.4 (Goemans et Williamson)**

Le problème de la coupe de poids maximum peut être formulé de la manière suivante :

$$Scoupe = \begin{cases} \text{Max} & \sum_{(ij) \in E} w_{ij} \frac{\arccos(v_i^t v_j)}{\pi} \\ \text{s.c.} & v_i \in S_p^n \quad \forall i \in X \end{cases}$$

Preuve :

Notons  $Z_{Qcoupe}^*$  et  $Z_{Scoupe}^*$  les valeurs optimales respectives des programmes  $Qcoupe$  et  $Scoupe$ . La fonction objective de  $Scoupe$  est équivalente à  $\frac{1}{2} \sum_{(ij) \in E} w_{ij}(1 - v_i^t v_j)$  si les

veteurs  $v_i$  sont de dimension 1. Autrement dit  $Z_{Scoupe}^* = Z_P^*$ ,  $Z_P^*$  étant la valeur optimale du programme  $P$  qui est une relaxation de  $Qcoupe$  et donc  $Z_{Scoupe}^* = Z_P^* \geq Z_{Qcoupe}^*$  par définition d'une relaxation.

$Z_{Scoupe}^*$  est exactement l'espérance mathématique de la coupe que peut fournir l'algorithme aléatoire, donc il existe nécessairement une coupe de valeur supérieure ou égale à  $Z_{Scoupe}^*$  et donc  $Z_{Scoupe}^* \leq Z_{Qcoupe}^*$ . ■

Une relaxation du polytope des coupes d'un graphe est étudiée par M. Laurent et S. Poljak dans [56]. Les auteurs montrent de plus comment on peut optimiser sur ce corps convexe appelé ellipsoïde. Toutefois ils montrent que déterminer si l'optimum est atteint sur un sommet de l'ellipsoïde est un problème  $\mathcal{NP}$ -complet.

### 4.3 Le problème de la $k$ -coupe de poids maximum

Le problème de la  $k$ -coupe consiste à déterminer un ensemble d'arêtes d'un graphe  $G = (X, E)$  de telle sorte que la suppression de ces arêtes sépare le graphe  $G$  en exactement  $k$  composantes connexes. Nous allons voir une extension de l'algorithme de Goemans et Williamson pour ce problème, celle-ci est présentée dans [31]. Cet algorithme étant valide pour des poids négatifs il peut donc être appliqué au calcul de la  $k$ -coupe de poids minimum si les poids sont positifs. Ce dernier problème peut nous aider à résoudre le problème d'affectation de sites sur des boucles SDH, ce problème consistant à déterminer une partition du graphe des demandes.

Considérons une partition  $P$  de l'ensemble des sommets du graphe  $G = (X, E)$ ;  $P = X_1, \dots, X_k$ , on définit le poids de la  $k$ -coupe ainsi déterminée par :

$$w(P) = \sum_{1 \leq r \leq s \leq k} \sum_{i \in X_r} \sum_{j \in X_s} w_{ij}$$

Le problème de  $k$ -coupe de poids maximum s'énonce alors de la manière suivante :

$$k - coupe = \begin{cases} Max & w(P) \\ s.c. & |P| = k \end{cases}$$

Afin d'obtenir un modèle associant à chaque sommet  $j$  du graphe un vecteur  $y_j$  qui pourra prendre sa valeur dans un ensemble de  $k$  vecteurs  $\{a_1, \dots, a_k\}$  où  $a_i = b_i - \frac{\sum_{i=1}^k b_i}{k}$ , les vecteurs  $b_i$ ,  $i \in \{1, \dots, k\}$  étant les sommets d'un simplexe de  $\mathbb{R}^{k-1}$  qu'on supposera normé de telle sorte que  $\|a_i\| = 1$ . On peut vérifier que ces vecteurs  $a_i$  satisfont  $a_i^t a_j = \frac{-1}{k-1}$  si  $i \neq j$ . Le problème de la  $k$ -coupe de poids maximum se reformule alors de la manière suivante :

$$k - coupe = \begin{cases} Max & \frac{k}{k-1} \sum_{(ij) \in E} w_{ij} (1 - y_i^t y_j) \\ s.c. & y_i \in \{a_1, \dots, a_k\}, \forall i \in X \end{cases}$$

Pour définir une relaxation semi-définie positive de  $k$ -coupe on remplace  $y_i$  par  $v_i \in S_p^k$  la sphère unitaire de  $\mathbb{R}^k$  et on obtient :

$$sd - k - coupe = \begin{cases} Max & \frac{k}{k-1} \sum_{(ij) \in E} w_{ij} (1 - v_i^t v_j) \\ s.c. & v_i \in S_p^k, \forall i \in X \\ & v_i^t v_j \geq \frac{-1}{k-1}, i \neq j \end{cases}$$

L'algorithme aléatoire de résolution du problème de la  $k$ -coupe de poids maximum consiste alors en la réalisation des trois étapes suivantes :

1. Résoudre  $sd - k - coupe$  de façon à obtenir les vecteurs  $v_1, \dots, v_n$ .
2. Tirer aléatoirement  $k$  vecteurs  $z_1, \dots, z_k$ .

3. Réaliser une partition  $X_1, \dots, X_k$  de  $X$  de la manière suivante :

$$X_i = \{j \in X \mid z_i^t v_j \geq z_l^t v_j, i \neq l\}$$

Si on note  $W$  la valeur d'une  $k$ -coupe produite par l'algorithme aléatoire,  $W^*$  la valeur d'une solution optimale au problème  $k$ -coupe et  $W_{sd}$  la valeur d'une solution optimale de  $sd$ - $k$ -coupe on a :

$$E(W) = \sum_{(ij) \in E} w_{ij} pr(y_i \neq y_j)$$

La probabilité  $pr(y_i \neq y_j)$  ne dépend que de l'angle que forment entre eux les vecteurs  $v_i$  et  $v_j$  :  $pr(y_i \neq y_j) = \frac{\arccos(v_i^t v_j)}{\pi}$ , ainsi obtient on :

$$\frac{E(W)}{W^*} \geq \frac{E(W)}{W_{sd}} = \frac{\sum_{(ij) \in E} w_{ij} \frac{\arccos(v_i^t v_j)}{\pi}}{\frac{k-1}{k} \sum_{(ij) \in E} w_{ij} (1 - v_i^t v_j)} \geq \text{Min} \left\{ \frac{k}{k-1} \frac{\theta}{\pi(1 - \cos \theta)}, \cos \theta \in \left[ \frac{-1}{k-1}, 1 \right] \right\}$$

#### 4.4 Formes hermitiennes et problèmes combinatoires

Nous présentons ici une application de la programmation sur le cône des formes hermitiennes semi-définies positives à la définition de relaxation de problèmes d'optimisation combinatoire. Cette application est due à M. Goemans et D. Williamson et elle concerne le problème  $Max\ 2 - lin - mod - 3$ , [41]. Une application semblable pour le problème plus général de satisfaction de contraintes  $Max\ k - lin - mod - p$  a été produite par L. Engebretsen durant sa thèse, [27].

Considérons  $n$  variables  $x_u$ ,  $u \in \{1, \dots, n\}$ ,  $c \in \{0, 1, 2\}$  et  $m$  équations  $x_u - x_v = c(mod.3)$  ou inéquations  $x_u - x_v \neq c(mod.3)$  dans lesquelles ne sont présentes simultanément que deux des variables. A chaque équation ou inéquation on associe un poids  $w_j$ . Le problème  $Max\ 2 - lin - mod - 3$  consiste à déterminer une affectation des variables  $x_u \in \{0, 1, 2\}$  de sorte à maximiser la somme des poids des équations ou inéquations qui sont satisfaites. Afin de modéliser le problème considérons l'ensemble des racines cubiques de l'unité  $\mathcal{R}_3 = \{1, e^{i\frac{2\pi}{3}}, e^{i\frac{4\pi}{3}}\}$  et associons à chaque variable  $x_u$  une variable  $y_u = e^{ix_u \frac{2\pi}{3}}$ . On peut alors remarquer que :

- $x_u \in \{0, 1, 2\} \implies y_u \in \mathcal{R}_3$  ;
- $y_u$  and  $y_v \in \mathcal{R}_3 \implies y_u \cdot y_v \in \mathcal{R}_3$  ;
- $x_u - x_v = c(mod.3) \iff y_u \cdot y_v = e^{ic\frac{2\pi}{3}}$  ;
- $x_u - x_v \neq c(mod.3) \iff y_u \cdot y_v \in \mathcal{R}_3 - \{e^{ic\frac{2\pi}{3}}\}$ .

Ainsi peut-on estimer la contribution de la  $j^{ieme}$  equation :

$$w_j \left( \frac{1}{3} + \frac{1}{3} e^{-ic\frac{2\pi}{3}} y_u \cdot y_v + \frac{1}{3} e^{ic\frac{2\pi}{3}} y_v \cdot y_u \right)$$

et celle de  $j^{ieme}$  inéquation :

$$w_j \left( \frac{2}{3} - \frac{1}{3} e^{-ic\frac{2\pi}{3}} y_u \cdot y_v + \frac{1}{3} e^{ic\frac{2\pi}{3}} y_v \cdot y_u \right)$$

et donc pouvons-nous formuler le problème de la manière suivante :

$$\left\{ \begin{array}{l} \text{Max} \quad \sum_{j \in J=} w_j \left( \frac{1}{3} + \frac{1}{3} e^{-ic \frac{2\pi}{3}} y_u \cdot y_v + \frac{1}{3} e^{ic \frac{2\pi}{3}} y_v \cdot y_u \right) \\ \quad + \sum_{j \in J\neq} w_j \left( \frac{2}{3} - \frac{1}{3} e^{-ic \frac{2\pi}{3}} y_u \cdot y_v + \frac{1}{3} e^{ic \frac{2\pi}{3}} y_v \cdot y_u \right) \\ \text{s.t.} \quad y_u \in \mathcal{R}_3 \quad \forall u \in \{1, \dots, n\} \end{array} \right.$$

Pour définir une relaxation semi-définie positive on procède de la manière suivante :

- on recherche des variables complexes de modules égal à 1 ;
- on remplace la condition  $y_u \in \mathcal{R}_3$  par la condition  $y_u \cdot y_v \in \mathcal{T}_3$  le triangle du plan complexe dont les sommets sont les points de  $\mathcal{R}_3$ .

Cette dernière condition peut s'écrire  $\alpha y_u \cdot y_v + \bar{\alpha} y_v \cdot y_u \geq -1$ . Maintenant considérons une matrice hermitienne  $Y$  telle que  $Y_{uv} = y_u \cdot y_v$  et exprimons les contraintes de la relaxation :

- $|y_u| = 1 \iff e_u e_u^t \cdot Y = E_{uu} \cdot Y = 1$
- $\alpha \in \mathcal{T}_3 \iff \alpha = e^{ic \frac{2\pi}{3}} \quad \forall c \in \{0, 1, 2\}$ , aussi les contraintes de triangle peuvent s'écrire :

$$e^{ic \frac{2\pi}{3}} (e_u e_v^t) \cdot Y + e^{-ic \frac{2\pi}{3}} (e_v e_u^t) \cdot Y = A_{uv}^c \geq -1 \quad \forall c \in \{0, 1, 2\}$$

remarquons que  $A_{uv}^c$  est une matrice hermitienne.

Considérons la matrice  $C$  définie par  $C_{uv} = w_j e^{-ic \frac{2\pi}{3}}$  et  $C_{vu} = w_j e^{ic \frac{2\pi}{3}}$ , par construction cette matrice est hermitienne. La relaxation semi-définie du problème peut alors se formuler comme suit :

$$\left\{ \begin{array}{l} \text{Max} \quad C \cdot Y \\ \text{s.t.} \quad A_{uv}^c \cdot Y \geq -1 \quad \forall u, \forall v \in \{1, \dots, n\}, \quad \forall c \in \{0, 1, 2\} \\ \quad \quad E_{uu} \cdot Y = 1 \quad \forall u \in \{1, \dots, n\} \\ \quad \quad Y \in H_n \\ \quad \quad Y \succeq 0 \end{array} \right.$$

Après résolution de cette relaxation les auteurs conçoivent un algorithme aléatoire, dont l'espérance est supérieure à 0.79373 fois l'optimum, reprenant un schéma semblable à celui pour l'algorithme aléatoire de résolution de la coupe de poids maximum :

- calcul des vecteurs  $y_u$  en utilisant une factorisation de Cholesky ;
- calcul d'un vecteur  $p \in \mathbb{C}^n$  tiré aléatoirement avec une distribution uniforme ;
- affecter aux variables leur valeur en appliquant la règle :

$$x_u = \begin{cases} 0 & \text{if } \text{Arg}(p \cdot y_u) \in [0, \frac{2\pi}{3}[ \\ 1 & \text{if } \text{Arg}(p \cdot y_u) \in [\frac{2\pi}{3}, \frac{4\pi}{3}[ \\ 2 & \text{if } \text{Arg}(p \cdot y_u) \in [\frac{4\pi}{3}, 0[ \end{cases}$$

Le problème de la 3-coupe de poids maximum peut se réduire à *Max2 - lin - mod3* de la façon suivante. Soit  $G = (V, E)$  un graphe,  $\{V_0, V_1, V_2\}$  une partition de  $V$ . A chaque sommet  $u \in V$  affectons une variable  $x_u = k$  si  $u \in V_k$ ,  $k \in \{0, 1, 2\}$ , on associe à chaque arête  $e \in E$  une inéquation  $x_u - x_v \neq 0 \pmod{3}$ .

## 4.5 Flot et délai

Dans cette section nous appliquons la programmation semi-définie positive à la résolution de problème de routage de flot. Dans un premier temps nous verrons comment minimiser le délai moyen, puis nous traiterons du problème du routage de coût minimum avec contrainte de délai de bout en bout.

Nous traitons ici de problème de réseaux de télécommunication utilisant la transmission par paquets. Dans de tels réseau les paquets devant être émis sur une artère de transmission sont d'abord rangés dans une file d'attente (tampon de mémoire) puis émis sur l'artère, de sorte que le temps nécessaire à la transmission d'un paquet est la somme du temps d'attente et du temps de service. Le délai de bout en bout est la somme des temps de transmission sur les artères du chemin de routage du paquet. La fonction de délai que nous utilisons est celle établie par L. Kleinrock, [51], qui permet d'évaluer le temps de traversée d'un arc lorsque les arrivées se font suivant une loi de Poisson et que la durée du service suit une loi exponentielle. La fonction de Kleinrock s'exprime par  $t_e = \frac{\phi_e}{C_e - \phi_e}$ , où  $t_e$  est le délai sur l'arc  $e$ ,  $C_e$  sa capacité et  $\phi_e$  le flot qui y circule. On peut voir qu'elle est représentable par une matrice semi-définie positive, sdp représentable dans un vocabulaire plus technique, en effet :

$$\begin{aligned} t_e \geq \frac{\phi_e}{C_e - \phi_e} &\iff (t_e + 1)(C_e - \phi_e) - C_e \geq 0 \\ &\iff \begin{pmatrix} t_e + 1 & \sqrt{C_e} \\ \sqrt{C_e} & C_e - \phi_e \end{pmatrix} \succeq 0 \end{aligned}$$

Remarquons que la fonction de Kleinrock possède une asymptote verticale en  $\phi_e = C_e$ , rendant inutiles les contraintes de capacité dans les problèmes de flot compatible. Pour une étude exhaustive des fonctions sdp représentables le lecteur peut se reporter aux travaux d'A. Nemirovski, [73]. Remarquons encore que  $\begin{pmatrix} t_e + 1 & \sqrt{C_e} \\ \sqrt{C_e} & C_e - \phi_e \end{pmatrix} \succeq 0$  est équivalent à dire qu'il existe deux vecteurs  $v_1$  et  $v_2$  réalisant une factorisation de Gram de la matrice, via une normalisation on peut choisir ces vecteurs tels que  $\|v_1\| = \|v_2\| = 1$ ; aussi voit-on que la contrainte de délai peut s'exprimer par  $\begin{pmatrix} 1 & x \\ x & 1 \end{pmatrix} \succeq 0$  avec  $x = 1 - t_e(C_e - \phi_e) + \phi_e$ .

### 4.5.1 Flot de délai moyen minimum

Le problème que nous considérons ici consiste à déterminer le flot qui minimise la somme de délais sur les arcs. Soit  $\Pi$  l'ensemble des chemins joignant deux sommets  $s$  et  $t$  du graphe,  $d$  la valeur du flot devant transiter entre  $s$  et  $t$  et  $\phi_p$  la quantité de flot circulant sur le chemin  $p \in \Pi$ , le problème du flot de délai moyen minimum peut alors se formuler de la manière suivante :

$$Fdm = \begin{cases} \text{Min} & \sum_{e \in E} \frac{\sum_{p \ni e} \phi_p}{C_e - \sum_{p \ni e} \phi_p} \\ \text{s.c.} & \sum_{p \in \Pi} \phi_p \geq d \\ & \phi_p \geq 0 \forall p \end{cases} \quad (4.4)$$

Ce problème a été étudié par P. Mahey et A. Ouorou, [66], [65], [74], voir aussi l'état de l'art [75] et par F. Boyer, [13]. Les méthodes de résolution qu'ils ont employées, décomposition proximale, méthode de Benders, se sont avérées très efficaces tant que la charge du réseau n'est pas trop forte. Par contre ces méthodes éprouvent des problèmes de convergence lorsque la charge du réseau se situe au niveau de la branche asymptotique de la fonction de délai. P. Mahey et A. Ouorou ont proposé un algorithme d'élimination de cycle pour les problèmes de flot non linéaire, [67]. Nous proposons donc de traiter dans ce cas le problème comme programme semi-défini positif, le problème  $Fdmm$  se formule alors comme suit :

$$Fdmm = \begin{cases} \text{Min} & \sum_{e \in E} t_e \\ \text{s.c.} & \sum_{p \in \Pi} \phi_p \geq d \\ & \begin{pmatrix} t_e + 1 & \sqrt{C_e} \\ \sqrt{C_e} & C_e - \sum_{p \ni e} \phi_p \end{pmatrix} \succeq 0 \quad \forall e \in E \\ & \phi_p \geq 0 \quad \forall p \end{cases} \quad (4.5)$$

**Lemme 4.3** *Le problème  $Fdmm$  peut être résolu en temps polynomial.*

Preuve :

Notons  $A$  la matrice d'incidence sommets-arcs du graphe  $G$  sur lequel est défini le problème de flot devant circuler entre les sommets  $s$  et  $t$ . On note  $b$  le vecteur dont toutes les composantes sont nulles sauf relatives à  $s$  qui vaut  $-1$  et celle relative à  $t$  qui vaut  $+1$ . Le problème  $Fdmm$  peut alors se reformuler de la manière suivante :

$$Fdmm = \begin{cases} \text{Min} & \sum_{e \in E} t_e \\ \text{s.c.} & A\phi = db \\ & \begin{pmatrix} t_e + 1 & \sqrt{C_e} \\ \sqrt{C_e} & C_e - \phi_e \end{pmatrix} \succeq 0 \quad \forall e \in E \\ & \phi_e \geq 0 \quad \forall e \end{cases}$$

où  $\phi$  désigne le vecteur des flots circulant sur chaque arc. On peut remarquer que le problème ainsi défini est un programme semi-défini positif qui comporte un nombre polynomial de variables et un nombre polynomial de contraintes, d'où le résultat du lemme. ■

#### 4.5.2 Flot de coût minimum et délai borné

Le problème qui est étudié ici consiste à déterminer le routage de moindre coût d'un flux circulant entre deux points, en imposant que le temps mis par un élément de ce flux pour se rendre à destination reste borné par une quantité qui peut être le délai maximum acceptable. Ce type de problème trouve des applications dans les réseaux de télécommunication, les réseaux routiers et d'une manière plus générale dans tous les réseaux sujets à une congestion.

Pour formaliser ce problème considérons un réseau  $N = (G, C)$  où  $G = (X, E)$  est un graphe et  $C \in \mathbb{R}^{|E|}$  un vecteur de capacité. De plus nous considérons que le graphe est muni de deux métriques. La première associe à chaque arc  $e$  un coût,  $k_e$ , par unité de

flot ; cette métrique est donc linéaire. La seconde associe à chaque arc un temps,  $t_e$ , correspondant au temps mis par un élément de flot pour traverser l'arc  $e$ . De plus on impose que le temps mis pour parvenir à destination reste inférieur à une valeur  $t$  quelque soit le chemin emprunté. On note  $\Pi$  l'ensemble des chemins joignant deux sommets  $s$  et  $t$  du graphe,  $d$  la valeur du flot devant transiter entre  $s$  et  $t$  et  $\phi_p$  la quantité de flot circulant sur le chemin  $p \in \Pi$ . Le problème du flot de coût minimum et délai borné,  $Fcldb$ , peut alors se formuler de la manière suivante :

$$Fcldb = \begin{cases} \text{Min} & \sum_{e \in E} k_e \sum_{p \ni e} \phi_p \\ \text{s.c.} & \sum_{p \in \Pi} \phi_p \geq d & 4.6.1 \\ & \sum_{p \ni e} \phi_p \leq C_e, \forall e \in E & 4.6.2 \\ & \sum_{e \in p} t_e \leq t, \forall p \in \Pi, \phi_p > 0 & 4.6.3 \\ & \phi_p \geq 0 \forall p \end{cases} \quad (4.6)$$

Soit  $\Pi^{\leq t} \subseteq \Pi$  l'ensemble des  $s-t$  chemins tels que  $\sum_{e \in p} t_e \leq t$ , nous pouvons alors reformuler le problème  $Fcldb$  de la manière suivante :

$$Fcldb = \begin{cases} \text{Min} & \sum_{e \in E} k_e \sum_{p \in \Pi^{\leq t}, p \ni e} \phi_p \\ \text{s.c.} & \sum_{p \in \Pi^{\leq t}} \phi_p \geq d \\ & \sum_{p \ni e} \phi_p \leq C_e, \forall e \in E \\ & \phi_p \geq 0 \forall p \in \Pi^{\leq t} \end{cases}$$

**Lemme 4.4** Si  $t_e = 1, \forall e \in E$  le problème  $Fcldb$  peut être résolu en temps polynomial. Si  $\forall e \in E t_e \in \mathbb{N}$  le problème  $Fcldb$  est  $\mathcal{NP}$ -difficile.

Preuve :

Considérons le problème dual du problème  $Fcldb$ , celui-ci se formule de la manière suivante :

$$Fcldb^* = \begin{cases} \text{Max} & \alpha d - \sum_{e \in E} \lambda_e C_e \\ \text{s.c.} & \alpha - \sum_{e \in p} \lambda_e \leq \sum_{e \in p} k_e, \forall p \in \Pi^{\leq t} \\ & 0 \leq \lambda_e \forall e \in E, 0 \leq \alpha \end{cases}$$

Le problème  $Fcldb^*$  a une solution si, chaque arc étant pondéré par  $k_e - \lambda_e$ , le plus court chemin tel que  $\sum_{e \in p} t_e \leq t$  a un poids supérieur ou égal à  $\alpha$ . Le problème de séparation sur le problème dual est donc équivalent à un problème de plus court chemin avec contrainte de longueur, or il est connu que ce problème est  $\mathcal{NP}$ -difficile sauf si une des deux pondérations vaut 1 sur tous les arcs, voir [35], problème ND 30. Le problème de savoir si  $Fcldb$  a une solution optimale bornée est donc  $\mathcal{NP}$ -complet sauf si  $t_e = 1 \forall e \in E$ , d'où le résultat du



lemme. ■

Remarquons toutefois que si  $t_e \in \mathbb{N} - \{0\}$  le problème peut être résolu en temps pseudo-polynomial en remplaçant l'arc  $e$  par un chemin comportant  $t_e$  arcs et en affectant le coût  $k_e$  à un seul des arcs composant ce chemin. Ainsi voit-on que le problème peut être résolu de manière pseudo-polynomiale pour toute pondération positive.

Nous allons maintenant considérer le cas où  $t_e$  est une fonction du flot circulant sur l'arc  $e$ , car les pondérations constantes ne peuvent être envisagées que dans le cas d'un réseau vide, c'est-à-dire sans congestion. Le problème du flot de coût minimum avec contrainte de délai peut alors se reformuler de la manière suivante :

$$Fcmdk = \left\{ \begin{array}{l} \text{Min} \quad \sum_{e \in E} k_e \sum_{p \ni e} \phi_p \\ \text{s.c.} \quad \sum_{p \in \Pi} \phi_p \geq d \\ \sum_{e \in p} t_e \leq t, \forall p \in \Pi \\ \begin{pmatrix} t_e + 1 & \sqrt{C_e} \\ \sqrt{C_e} & C_e - \sum_{p \ni e} \phi_p \end{pmatrix} \succeq 0 \forall e \in E \\ \phi_p \geq 0 \forall p \end{array} \right. \quad \begin{array}{l} 4.7.1 \\ 4.7.2 \\ 4.7.3 \end{array} \quad (4.7)$$

**Remarque :** Si on compare le problème *Fcmdk* au problème *Fcmdb* on peut voir qu'il est sur-contraint par rapport à ce dernier. En effet nous faisons peser la contrainte de délai sur l'ensemble des chemins et non sur ceux qui supportent du flot, la définition d'un ensemble semblable à  $\Pi^{\leq t}$  s'avère problématique parce que pour le définir il faut tenir compte de l'ensemble des combinaisons de flot. Toutefois on peut obvier partiellement cet inconvénient. On peut initialement considérer un sous-ensemble de chemins tels que les flots sur chacun de ces chemins sont strictement positifs, cet ensemble formant une base. Alors en procédant par génération de colonnes on va faire, à chaque étape, entrer une variable en base et en faire sortir une. La variable entrante portera un flot strictement positif, dans le cas contraire son entrée est inutile. La variable sortante peut être déterminée à l'avance, on la retire alors de l'ensemble des chemins sur lequel on définit le problème. Ceci définit une heuristique pour le cas où on veut contraindre que les chemins supportant du flot.

La contrainte de capacité 4.6.2 ne figure pas explicitement dans la formulation du problème *Fcmdk*, elle est contenue dans la contrainte de semi-définie positivité associée à chaque arc, les termes diagonaux d'une matrice semi-définie positive étant positifs ou nuls.

Afin de rendre ce qui suit plus clair nous donnons la structure de la matrice des variables du problème *Fcmdk* ainsi que celle des matrices permettant d'exprimer les contraintes linéaires. Nous montrerons ainsi que le problème *Fcmdk* peut s'écrire comme un problème d'optimisation sur le cône des matrices semi-définies positives. Afin d'obtenir des contraintes d'égalité nous réécrivons les contraintes en employant des variables d'écart lorsque c'est nécessaire. Ainsi la contrainte de satisfaisabilité de la demande devient  $\sum_{p \in \Pi} \phi_p = d$ . La contrainte de délai sur un chemin s'écrit de la manière suivante en lui

associant une variable d'écart  $b_p$  :  $\sum_{e \in p} t_e + b_p = t$  ; enfin la contrainte de capacité s'écrit





avec  $c_{p_j} = \sum_{e \in p_j} k_e$ . Le problème  $Fcmdk$  peut alors se reformuler de la manière suivante :

$$Fcmdk = \begin{cases} \text{Min} & C \cdot X \\ \text{s.c.} & A_d \cdot X = d \\ & A_{t_p} \cdot X = t, \forall p \in \Pi \\ & A_{C_e} \cdot X = C_e \forall e \in E \\ & A_{\sqrt{C_e}} \cdot X = 2\sqrt{C_e} \forall e \in E \\ & X \succeq 0 \end{cases}$$

Nous allons maintenant définir le programme dual du problème  $Fcmdk$ , pour cela associons une variable  $\alpha$  à la contrainte 4.7.1, une variable  $\beta_p$  à chacune des contraintes de type 4.7.2 et une matrice  $\begin{pmatrix} x_e & z_e \\ z_e & y_e \end{pmatrix}$  à chaque contrainte de type 4.7.3. Le programme dual de  $Fcmdk$  se formule alors de la manière suivante :

$$Fcmdk^* = \begin{cases} \text{Max} & \alpha d - t \sum_{p \in \Pi} \beta_p - \sum_{e \in E} (x_e + C_e y_e + 2\sqrt{C_e} z_e) \\ \text{s.c.} & \alpha - \sum_{e \in p} y_e \leq \sum_{e \in p} k_e, \forall p \in \Pi \\ & - \sum_{p \ni e} \beta_p + x_e = 0, \forall e \in E \\ & \begin{pmatrix} x_e & z_e \\ z_e & y_e \end{pmatrix} \succeq 0, \\ & \alpha \geq 0, \beta_p \geq 0 \forall p \in \Pi \end{cases} \quad (4.8)$$

Nous avons vu à la section 3.5 que l'existence d'une solution strictement réalisable du programme dual permet d'assurer que le saut de dualité est nul, aussi regardons ce qu'il en est pour le couple de problème  $Fcmdk, Fcmdk^*$ .

**Lemme 4.5** *Si il existe un chemin  $p \in \Pi$  pour lequel la contrainte de délai est satisfaite de manière lâche alors le problème  $Fcmdk$  a un intérieur relatif à la variété linéaire  $A_d \cdot X = d$  non vide.*

Preuve :

Si un chemin satisfait la contrainte de délai de manière lâche et tous les autres la satisfont de manière serrée alors il est clair qu'on peut retirer une fraction, fut-elle infinitésimale, de flot sur chacun de ces chemins de sorte qu'ils vérifient à leur tour la contrainte de délai de manière lâche, aussi pouvons nous dire qu'il existe une solution où toutes les variables  $b_j, j \in \{1, \dots, |\Pi|\}$  sont strictement positives. On peut considérer sans perte de généralité qu'aucun des flots  $\phi_p$  est nul, en effet si tel n'est pas le cas on peut retirer de l'ensemble  $\Pi$  les chemins supportant un flot nul. De même on peut considérer que le délai sur un arc  $t_e$  est strictement positif. Par ailleurs le terme  $\eta_e^2$  ne peut pas être nul car dans ce cas le délai sur l'arc  $e$  tend vers l'infini et la contrainte de délai ne serait pas respectée. Ainsi nous pouvons affirmer que si il existe un chemin pour lequel la contrainte de délai est vérifiée de manière lâche on peut construire une matrice  $X$  dont les termes diagonaux sont tous strictement positifs et donc une matrice  $X \succ 0$ . ■

**Corollaire 4.3** *Les programmes  $Fcmdk$  et  $Fcmdk^*$  ont une valeur optimale commune bornée.*

De même que le problème  $Fcldb$ , le problème  $Fcldb$  compte un nombre exponentiel de variables, autant que de chemins entre deux sommets. Par contre il comporte un nombre exponentiel de contraintes, contraintes de délai sur les chemins de type 4.7.2. Pour ces raisons nous recourons à un algorithme de génération de colonnes, notons que la génération de colonnes consiste en la génération de chemins, aussi quand on génère une variable  $\phi_p$  on peut simultanément générer la contrainte ayant trait à ce chemin. On peut donc affirmer que la complexité du problème  $Fcldb$  est du même ordre que celle de la génération des variables  $\phi_p$ . Considérons un sous-ensemble de chemins  $\Pi_0$  tel que le problème  $Fcldb$  défini sur ce sous-ensemble a une solution réalisable.

**Lemme 4.6** *Soit  $p \in \Pi - \Pi_0$  un chemin tel que  $\sum_{e \in p} k_e + y_e < \alpha$  et  $\sum_{e \in p} t_e < t$ . On note  $\phi_0^*$  une solution optimale de  $Fcldb$  défini sur le sous-ensemble de chemin  $\Pi_0$  et  $\phi_1^*$  une solution optimale de  $Fcldb$  sur le sous-ensemble de chemins  $\Pi_1 = \Pi_0 \cup \{p\}$ . Alors  $k^t \phi_0^* > k^t \phi_1^*$ .*

Preuve :

Le fait que  $\sum_{e \in p} t_e < t$  autorise une augmentation du flot sur le chemin  $p$ , d'autre part on est sûr que cette augmentation n'entraînera pas la violation de la contrainte sur un autre chemin supportant du flot. Le coût réduit de la variable  $\phi_p$ ,  $\sum_{e \in p} k_e + y_e < \alpha$  étant négatif il est clair que l'accroissement de la valeur de la variable  $\phi_p$  produit une décroissance de la fonction objective. ■

Nous allons voir maintenant une condition d'optimalité pour le problème  $Fcldb$ .

**Lemme 4.7** *Soit  $\phi^*$  un point extrême du domaine sur lequel est défini le problème  $Fcldb$ , alors il existe au plus un chemin  $p \in \Pi$  tel que  $\sum_{e \in p} t_e < t$  et  $\phi_p^* > 0$*

Preuve :

Supposons qu'il existe deux chemins  $p$  et  $p' \in \Pi$  tels que  $\phi_p^* > 0$ ,  $\phi_{p'}^* > 0$ ,  $\sum_{e \in p} t_e < t$  et

$\sum_{e \in p'} t_e < t$  dans une solution optimale  $\phi^*$ . Si  $\sum_{e \in p} k_e < \sum_{e \in p'} k_e$  il est clair qu'en augmentant le

flot sur le chemin  $p$  et en le diminuant sur  $p'$  on obtient une solution  $\phi'$  telle  $\sum_{e \in E} k_e \sum_{pi \ni e} \phi'_{pi} <$

$\sum_{e \in E} k_e \sum_{pi \ni e} \phi^*_{pi}$ , ce qui contredit l'optimalité de  $\phi^*$ . Si  $\sum_{e \in p} k_e = \sum_{e \in p'} k_e$  il est clair que  $\phi^*$

comme combinaison linéaire convexe d'une solution  $\phi$  où  $\sum_{e \in p} t_e = t$  et d'une solution  $\phi'$  où

$\sum_{e \in p'} t_e = t$ , ce qui contredit l'hypothèse  $\phi^*$  est un point extrême du domaine de  $Fcldb$ . ■

**Lemme 4.8** *Dans toute solution optimale du programme dual de  $Fcldb$ ,  $Fcldb^*$ ,  $y_e = 0$ ,  $\forall e \in p$ , où  $p$  est un chemin pour lequel la contrainte de délai n'est pas serrée,  $\sum_{e \in p} t_e < t$*

Preuve :

Supposons qu'il existe un arc  $e \in p$ ,  $p$  étant un chemin qui ne sature pas la contrainte de délai pour lequel  $y_e > 0$ . Le coût marginal du chemin  $p$  est  $\sum_{e \in p} k_e + y_e$  et alors nous

avons  $\sum_{e \in p} k_e < \sum_{e \in p} k_e + y_e$ . Ce qui introduit un saut de dualité et contredit le théorème de dualité forte dont nous avons vu au lemme 4.5 que les hypothèses sont vérifiées. ■

**Théorème 4.5** *Soit  $\Pi_j$  le sous-ensemble de chemins sur lequel est défini le problème  $Fcmdk$  après la génération de  $j$  variables de chemins  $\phi_p$ ; on note  $\phi^*(j)$  une solution optimale du problème ainsi défini. Si  $\forall p \in \Pi - \Pi_j$  tel que  $\sum_{e \in p} t_e < t$  on a  $\alpha \leq \sum_{e \in p} (k_e + y_e)$ , alors  $\phi^*(j)$  est une solution optimale de  $Fcmdk$ .*

Preuve :

Supposons qu'il existe  $p_{j+1} \in \Pi$  tel que  $\phi^*(j)$  n'est pas une solution optimale de  $Fcmdk$ , alors  $\sum_{e \in p_{j+1}} k_e + y_e < \alpha$ . Par hypothèse pour tout chemin  $p$  tel que  $\sum_{e \in p} t_e < t$  on a  $\alpha \leq$

$\sum_{e \in p} (k_e + y_e)$  aussi peut-on dire que  $\sum_{e \in p_{j+1}} t_e \geq t$ . Supposons qu'à chacune des  $j$  itérations nous avons généré la variable  $\phi_p$  correspondant au plus court chemin au sens des coûts réduits.

Soit  $\Pi'$  le sous-ensemble des chemins qui ont au moins un arc en commun avec  $p_{j+1}$  et tel que  $\phi_{p'} > 0$ ,  $p' \in \Pi'$ . On note  $\phi(\Pi')$  la quantité minimale de flot qu'il faut retrancher des chemins de  $\Pi'$  pour que  $p_{j+1}$  devienne admissible,  $C(\Pi')$  désigne le coût moyen de routage d'une unité de flot sur les chemins de  $\Pi'$ . Par le lemme 4.7 nous savons qu'il existe au plus un chemin  $p_j$  qui ne sature pas la contrainte de délai et qui supporte un flot non nul, supposons que ce chemin existe et que sa capacité résiduelle est supérieure à  $\phi(\Pi')$ , si ce n'est pas le cas  $p_{j+1}$  ne peut pas être rendu réalisable et le cas est trivial. De plus on peut encore supposer que tout chemin de  $\Pi'$  supporte un flot non nul, en effet si tel n'est pas le cas on peut retirer de  $\Pi'$  les chemins de flot nul sans changer la solution  $\phi^*(j)$ .  $p_j$  est donc l'unique chemin sur lequel la quantité de flot peut croître. Par ailleurs on peut dire que  $\sum_{e \in p_j} k_e \geq C(\Pi')$ , en effet si tel n'est pas le cas  $p_j$  aurait été saturé et

$\phi^*(j)$  ne serait pas optimale. D'autre part le fait qu'à chaque étape la variable de chemin générée correspond au plus court chemin au sens des coûts réduits nous permet d'écrire que  $\sum_{e \in p_{j+1}} k_e \geq \sum_{e \in p} k_e \forall p \in \Pi(j)$ , en effet si tel n'est pas le cas  $p_{j+1}$  aurait été généré avant

l'étape  $j$  car comme nous l'avons vu au lemme 4.8 les coûts marginaux se confondent avec les coûts lorsque les chemins ne sont pas saturés. Si  $p_{j+1}$  transporte une quantité  $\epsilon$  de flot prise sur un chemin  $\bar{p} \in \Pi(j) - \{p_j\}$  la variation de la fonction objective a pour expression :

$$\delta = \left[ \sum_{e \in p_j} k_e - C(\Pi') \right] \phi(\Pi') + \left[ \sum_{e \in p_{j+1}} k_e - \sum_{e \in \bar{p}} k_e \right] \epsilon$$

Aussi pouvons conclure au fait que  $p_{j+1}$  ne fait pas décroître la fonction objective et donc que  $\phi^*(j)$  est une solution optimale pour  $Fcmdk$ .

**Corollaire 4.4** *Le problème Fcmdk est NP-difficile.*

Preuve :

Par le théorème précédent nous pouvons dire que le problème de génération de colonnes pour Fcmdk est équivalent à la recherche d'un plus court chemin au sens de coûts  $k_e$  et dont le délai est borné par  $t - \epsilon$ , or nous avons vu que ce problème est NP-complet. ■

### Le problème de génération de colonnes

Nous venons de voir que le problème de génération de colonnes consiste à calculer un plus court chemin au sens des coûts réduits et tel que le délai demeure inférieur à la valeur du délai maximum. Ce type de problème est connu sous le nom de problème du plus court chemin avec contraintes de ressources, il a été notamment étudié par J.E. Beasley et N. Christofides dans [9] où les auteurs proposent une relaxation lagrangienne du problème. Formulons donc se problème de plus court chemin avec contrainte de ressources. Soit  $x_e$  la variable booléenne qui vaut 1 si l'arc  $e$  est dans le plus court chemin recherché et 0 sinon. On note  $\gamma_e$  le coût de l'arc  $e$ , pour notre problème de séparation il vaut le coût réduit de l'arc  $e$ . On désigne par  $t_e$  la ressource consommée si on emprunte l'arc  $e$ ,  $t$  est la quantité maximale de ressource qui peut être consommée sur un chemin. On obtient alors le modèle ci-dessous :

$$Pcr = \begin{cases} \text{Min} & \sum_{e \in E} \gamma_e x_e \\ \text{s.c.} & \sum_{e \in \delta^+(S)} x_e \geq 1, \forall S \subset V, s \in S, t \notin S \\ & \sum_{e \in E} t_e x_e \leq t \\ & x \in \{0, 1\}^{|E|} \end{cases}$$

**Remarque :**  $\sum_{e \in E} t_e x_e \leq t$  est équivalent à  $\sum_{e \in p} t_e x_e \leq t$  si  $p$  est le chemin recherché car  $x_e = 1$  si et seulement si  $e \in p$ .

On définit une relaxation lagrangienne du problème  $Pcr$  par dualisation de la contrainte de ressource et on obtient le lagrangien du problème qui s'écrit comme suit :

$$\mathcal{L}(x, \lambda) = \sum_{e \in E} \gamma_e x_e - \lambda \left( \sum_{e \in E} t_e x_e - t \right)$$

puis la fonction duale :

$$\omega(\lambda) = \begin{aligned} & \text{Min} \mathcal{L}(x, \lambda) \\ & \sum_{e \in \delta^+(S)} x_e \geq 1, \forall S \subset V, s \in S, t \notin S \\ & 0 \leq x \leq 1 \end{aligned}$$

**Remarque :** La substitution de la contrainte  $0 \leq x \leq 1$  à la contrainte d'intégrité  $x \in \{0, 1\}^{|E|}$  est légale car il est connu que le polyèdre des chemins et celui des coupes

sont des polyèdres de type bloquant qui ont la propriété  $\mathbb{Q}^+$  max-flow-min-cut, ou d'une manière équivalente qu'ils vérifient les hypothèses du théorème de Lehman. Le problème dual s'énonce alors de la manière suivante :  $Max\{\omega(\lambda), \lambda \in \mathbb{R}\}$ . La recherche d'un point selle pour le problème dual peut se faire d'une manière classique par un algorithme de sous-gradient, fournissant une heuristique pour le problème de séparation.



## Chapitre 5

# Lift-and-project

Lift-and-project est une méthode qui consiste à transformer un problème de  $n$  variables entières en un problème de  $n^2$  variables et à projeter ce dernier sur l'espace d'origine. Cette méthode permet en  $n$  étapes de générer l'enveloppe convexe des points à coordonnées entières du domaine sur lequel le problème d'optimisation est défini.

Dans la suite de ce chapitre nous étudions en détail la méthode de lift-and-project en regardant les connexions qui existent avec la programmation semi-définie positive. Puis nous étudierons un schéma séquentiel de lift-and-project tel qu'il est défini dans [6].

### 5.1 La construction de Lovász et Schrijver

Cette construction qui est exposée dans [64] permet de mettre en valeur les relations qui existent entre lift-and-project et la programmation semi-définie positive.

Nous avons vu dans la section 3.3 la définition des cônes de matrices  $M(K)$  et  $M^+(K)$  ainsi que différentes propriétés vérifiées par ces matrices. La définition de ces deux cônes consiste en l'étape de lift, c'est à dire celle qui consiste à transformer le problème initial défini dans  $\mathbb{R}^{n+1}$  en un problème défini dans  $\mathbb{R}^{(n+1)(n+1)}$ . En remarquant que  $x_i x_j \geq 0$ ,  $x_i(1-x_j)$  et  $(1-x_i)(1-x_j)$  fournissent après linéarisation,  $x_i x_j = y_{ij}$ ,  $i \neq j$  et  $x_i^2 = y_{ii}$ , des inégalités valides pour les problèmes en 0-1, on aurait pu réaliser l'étape de lift en multipliant le système  $[-b, A]x \geq 0$  par  $x_i$  et  $1-x_i$ ,  $\forall i \in \{1, \dots, n\}$ .

Après avoir vu l'étape de lift nous allons voir l'étape de projection et la façon dont est construite l'enveloppe convexe,  $K_1^0$ , des points de l'ensemble défini par  $K \cap \{x \in \mathbb{R}^{n+1} | x_0 = 1\} \cap \{0, 1\}^{n+1}$ . Soit  $K^0 = \text{cône}(K \cap \{0, 1\}^{n+1})$ , on a alors  $K_1^0 = K^0 \cap \{x | x_0 = 1\}$ .

Rappelons que  $K \subseteq Q$  est le cône défini par  $K = \{x \in \mathbb{R}^{n+1} | [-b, A]x \geq 0\}$ , où  $Q$  est le cône de  $\mathbb{R}^{n+1}$  engendré par les vecteurs à composantes 0-1 avec  $x_0 = 1$ .

Les projections de  $M(K)$  et  $M^+(K)$  sont respectivement définies par :

$$N(K) = \{Y e_0 | Y \in M(K)\} = \{\text{diag}(Y) | Y \in M(K)\} \quad (5.1)$$

et

$$N^+(K) = \{Y e_0 | Y \in M^+(K)\} = \{\text{diag}(Y) | Y \in M^+(K)\} \quad (5.2)$$

En effet rappelons-nous que  $x$  appartenant à  $\{0, 1\}^n$  nous avons posé  $y_{ii} = x_i^2 = x_i$ .

**Lemme 5.1 (Lovász et Schrijver)**

Si  $K \subseteq Q$  alors :

$$K^0 \subseteq N^+(K) \subseteq N(K) \subseteq K.$$

Preuve :

1.  $K^0 \subseteq N^+(K)$

Soit  $x$  un vecteur de  $K$  à composantes en  $0 - 1$ ,  $x \neq 0$ . Considérons la matrice  $Y = xx^t$ , il est clair que  $Y \in M^+(K)$ . Partant de là on a immédiatement  $x = Ye_0 \in N^+(K)$ .

2.  $N^+(K) \subseteq N(K)$

Par construction  $M^+(K) = M(K) \cap U_{psd} \implies M^+(K) \subseteq M(K)$ , par définition  $N^+(K)$  et  $N(K)$  sont les projections de  $M^+(K)$  et  $M(K)$  suivant une même direction définie par  $e_0$ , aussi on a  $N^+(K) \subseteq N(K)$ .

3.  $N(K) \subseteq K$

Soit  $x \in N(K)$  alors il existe  $Y \in M(K)$  telle que  $x = \text{diag}(Y) = Ye_0$ . Par hypothèse  $K \subseteq Q$  aussi  $Q^* \subseteq K^*$  et par le lemme 3.1  $Q^* = \text{cône}\{e_i, f_i\}$  aussi peut-on dire que  $e_0 \in K^*$ . On a vu que les matrices de  $M(K)$  vérifient  $u^t Y v \geq 0$ ,  $\forall u \in K^*$  et  $\forall v \in Q^*$ , ce qui est équivalent à dire que  $YQ^* \subseteq K$  et donc  $x \in K$ . ■

Avant de poursuivre la construction de  $K^0$  nous allons voir deux propriétés satisfaites par les matrices de  $M(K)$  qui seront utiles par la suite.

**Lemme 5.2 (Lovász et Schrijver)**

Toute matrice  $Y \in M(K)$  vérifie :

1. Toute colonne de  $Y$  est un vecteur de  $K$ .
2. la différence entre la première colonne de  $Y$  et toute autre de ses colonnes est un vecteur de  $K$ .

Preuve :

On a vu que toute matrice  $Y \in M(K)$  vérifie  $u^t Y v \geq 0$ ,  $\forall u \in K^*$  et  $\forall v \in Q^*$  et que ceci est équivalent à dire que  $YQ^* \subseteq K$ , de plus on sait que  $Q^* = \text{cône}\{e_i, f_i\}$ . On peut alors affirmer que :

1.  $Ye_i \in K$ , toute colonne de  $Y$  est un vecteur de  $K$ .
2.  $Yf_i = Y(e_0 - e_i) \in K$ , la différence des colonnes 0 et  $i$  est dans  $K$ . ■

Les deux lemmes suivants donnent une interprétation géométrique de  $N(K)$  en le situant par rapport aux hyperplans d'appui de  $Q$ . On définit  $H_i = \{x \in \mathbb{R}^{n+1} \mid x_i = 0\}$  et  $G_i = \{x \in \mathbb{R}^{n+1} \mid x_i = x_0\}$ , rappelons-nous que nous avons posé dès le début  $x_0 = 1$ . Il est clair que  $H_i$  et  $G_i$  sont des hyperplans d'appui de  $Q$  et que toutes les facettes de  $Q$  peuvent être décrites par de tels hyperplans.

**Lemme 5.3 (Lovász et Schrijver)**

Pour tout cône  $K \subseteq Q$  et pour toute coordonnée  $i \in \{1, \dots, n\}$  on a :

$$N(K) \subseteq (K \cap H_i) + (K \cap G_i)$$

Preuve :

Soit  $x \in N(K)$  alors il existe une matrice  $Y \in M(K)$  telle que  $x = Ye_0$ . Soit  $y_i$  la  $i^{\text{eme}}$  colonne de  $Y$ , on a vu au lemme 5.2 que  $y_i \in K$  et que  $y_{.0} - y_i \in K$ . D'autre part étant donné que  $Y$  est symétrique et que  $\text{diag}(Y) = Ye_0$  on a  $y_{i0} = y_{ii} = y_{0i}$  ce qui permet d'affirmer que  $y_i \in G_i$  et que  $y_{.0} - y_i \in H_i$ . Aussi peut-on conclure :

$$x = Ye_0 = y_{.0} = y_i + (y_{.0} - y_i) \in (K \cap H_i) + (K \cap G_i)$$

■

**Remarques :**

1. Le lemme ci-dessus ne signifie pas que  $N(K)$  touche  $H_i$  et  $G_i$  il signifie que tout vecteur de  $N(K)$  peut s'écrire comme la somme d'un vecteur de  $K$  dont la  $i^{\text{eme}}$  coordonnée vaut 1 et d'un vecteur de  $K$  dont la  $i^{\text{eme}}$  coordonnée vaut 0.
2. Une autre conséquence est que si  $K \cap G_i = \{0\}$  alors  $N(K) \subseteq H_i$ , en particulier si  $K$  ne touche des facettes disjointes de  $Q$  qu'à l'origine  $N(K) = \{0\}$ . En exploitant le caractère semi-défini positif des matrices de  $M^+(K)$  on peut énoncer un lemme semblable pour  $N^+(K)$ , mais son interprétation géométrique est mal définie.
3. Une implication intéressante de ce lemme est la possibilité de concevoir un algorithme géométrique en définissant l'opérateur  $N_0(K) = \cap_{i=1}^n ((K \cap G_i) + (K \cap H_i))$ . On peut voir que  $N_0(K) = M_0(K)e_0$  où  $M_0(K)e_0$  est le cône des matrices qui vérifient  $y_{0i} = y_{i0}$ ,  $\text{diag}(Y) = Ye_0$  et  $u^t Y v \geq 0 \forall u \in K^*, \forall v \in Q^*$ . La propriété  $y_{0i} = y_{i0}$  est moins forte que  $Y = Y^t$  aussi  $M(K) \subseteq M_0(K)$  et donc  $N(K) \subseteq N_0(K)$ .
4. De plus on peut voir que si une inégalité  $a^t x \leq b$  est valide pour  $K \cap H_i$  et pour  $K \cap G_i$  alors elle est valide pour  $N(K)$ .

**Lemme 5.4 (Lovász et Schrijver)**

$$N(K \cap H_i) = N(K) \cap H_i.$$

Preuve :

1.  $N(K \cap H_i) \subseteq N(K) \cap H_i$   
Soit  $x \in N(K \cap H_i)$  alors il existe une matrice  $Y \in M(K \cap H_i)$  telle que  $x = Ye_0$ . Par construction de  $M(K \cap H_i)$  la matrice  $Y$  satisfait  $u^t Y v \geq 0 \forall u \in (K \cap H_i)^*, \forall v \in Q^*$ , autrement dit  $YQ^* \subseteq K \cap H_i$ . D'autre part on sait que  $Q^* = \text{cône}\{e_i, f_i\}$  ce qui permet de dire que  $y_j \in K \cap H_i$  et  $y_{.0} - y_j \in K \cap H_i$ .  $K \cap H_i$  est un cône convexe donc stable par addition aussi  $x = y_j + (y_{.0} - y_j) \in K \cap H_i$ . Mais on a aussi  $y_j \in K$  et  $y_{.0} - y_j \in K$  ce qui revient à dire que  $Y \in M(K)$  et donc  $x \in N(K)$ , de plus on a vu que  $y_j \in H_i$  et  $y_{.0} - y_j \in H_i$ ; on peut donc conclure que  $x \in N(K) \cap H_i$ .
2.  $N(K \cap H_i) \supseteq N(K) \cap H_i$   
La preuve de cette inclusion reprend les mêmes arguments que ceux employés pour celle de la précédente.

■

**Remarque :** Une conséquence intéressante de ce lemme est que  $N(K \cap H_i)$  ne dépend pas de la manière dont il est obtenu que ce soit comme cône de  $\mathbb{R}^{n+1}$  ou comme cône de  $H_i$ . D'une manière similaire on peut établir un résultat équivalent pour  $N(K \cap G_i)$ .

**Lemme 5.5**  $N(K \cap G_i) = N(K) \cap G_i$ .

Preuve :

1.  $N(K \cap G_i) \subseteq N(K) \cap G_i$

Soit  $x \in N(K \cap G_i)$  alors il existe une matrice  $Y \in M(K \cap G_i)$  telle que  $x = Y e_0$ . Par construction de  $M(K \cap G_i)$  la matrice  $Y$  satisfait  $Y Q^* \subseteq K \cap G_i$ , de plus nous savons que  $Q^* = \text{cône}\{e_i, f_i\}$  donc  $y_{.j} \in K \cap G_i$  et  $y_{.0} - y_{.j} \in K \cap G_i$ . Aussi  $y_{.j}$  et  $y_{.0} - y_{.j}$  appartiennent à  $K$  et  $G_i$ , ce qui permet de dire que  $Y \in M(K)$  et par voie de conséquence  $x \in N(K)$ , de même  $x \in G_i$  et donc on peut conclure que  $x \in N(K) \cap G_i$ .

2.  $N(K \cap G_i) \supseteq N(K) \cap G_i$

Ce sens se démontre avec les mêmes arguments que ceux employés précédemment. ■

M. Goemans et L. Tunçel [39] ont prouvé les deux lemmes précédents dans une forme légèrement plus générale et avec des arguments semblables.

**Lemme 5.6 (Goemans et Tunçel)**

*Soit  $F$  une face quelconque de  $Q$ , alors  $N(K \cap F) = N(K) \cap F$ .*

Toutefois pour obtenir ce résultat depuis les lemmes 5.4 et 5.5 il suffit de remarquer que les faces du cône  $Q$  peuvent toutes s'écrire comme intersection d'hyperplans  $H_i$  et  $G_i$ .

Nous verrons à la section 6.2.3 quelles sont les applications de ces lemmes lorsque nous employons les coupes produites par l'opérateur  $N(K)$  dans un schéma de type branch-and-cut.

Le théorème suivant est le théorème central de ce chapitre, c'est lui qui décrit la méthode récursive de construction de l'enveloppe convexe des points à composantes 0-1 du domaine.

**Théorème 5.1 (Lovász et Schrijver)**

*On définit récursivement  $N^t(K)$  par  $N^0(K) = K$  et  $N^t(K) = N(N^{t-1}(K))$ . Alors  $N^n(K) = K^0$ .*

Preuve :

Considérons le cube unitaire  $Q'$  contenu dans l'hyperplan d'équation  $x_0 = 1$ . Soit  $t$  un index parcourant  $\{1, \dots, n\}$ . Considérons une face quelconque de  $Q'$  de dimension  $n - t$ , soit  $F$  cette face ; on définit alors  $\bar{F}$  comme étant l'union des faces parallèles à  $F$ . Remarquons que  $F \subseteq \bar{F}$ . Si  $F$  est une face de dimension 0 alors  $\bar{F} = Q'$  et on a  $\text{cône}(K \cap \bar{F}) = K^0$ , ce qui correspond à  $t = n$ . On a vu que  $K^0 \subseteq N(K)$  et donc par récurrence  $K^0 \subseteq N^t(K)$ ,  $t \in \{1, \dots, n\}$ . Pour démontrer le théorème il suffit de montrer que  $N^n(K) \subseteq K^0$  ou bien que  $\forall t \in \{1, \dots, n\} N^t(K) \subseteq \text{cône}(K \cap \bar{F})$ , ce qui peut être fait par récurrence.

Pour  $t = 1$  on peut voir que  $N^1(K) \subseteq \text{cône}(K \cap \bar{F})$  est équivalent au lemme 5.3 puisque

$H_i$  et  $G_i$ ,  $i \in \{1, \dots, n\}$  sont des facettes de  $Q'$ , aussi peut-on considérer que pour  $t = 1$  on a :

$$N(K) \subseteq \text{cône}(K \cap \bar{F}), \quad F \text{ étant une face de dimension } n - 1. \quad (5.3)$$

Notons  $F'$  une face de dimension  $n - (t - 1)$  de  $Q'$  contenant  $F$  une face de dimension  $n - t$  et soit  $i \in \{1, \dots, n\}$  tel que  $F = F' \cap H_i$ , alors l'hypothèse de récurrence permet d'écrire  $N^{t-1}(K) \subseteq \text{cône}(K \cap \bar{F}')$ . Alors en appliquant le résultat du lemme 5.3 on a :

$$\begin{aligned} N^t(K) = N(N^{t-1}(K)) &\subseteq \text{cône}(N^{t-1}(K) \cap (H_i \cup G_i)) \\ &\subseteq \text{cône}([\text{cône}(K \cap \bar{F}') \cap H_i] \cup [\text{cône}(K \cap \bar{F}') \cap G_i]) \end{aligned} \quad (5.4)$$

$H_i$  est un hyperplan support de  $\text{cône}(K \cap \bar{F}')$  et son intersection avec ce cône est engendrée par son intersection avec l'ensemble générant ce cône :

$$\text{cône}(K \cap \bar{F}') \cap H_i = \text{cône}(K \cap \bar{F}' \cap H_i) \subseteq \text{cône}(K \cap \bar{F}')$$

D'une manière identique on a  $\text{cône}(K \cap \bar{F}') \cap G_i \subseteq \text{cône}(K \cap \bar{F}')$ . Alors par 5.3 et 5.4 on a établi que  $\forall t \in \{1, \dots, n\} N^t(K) \subseteq \text{cône}(K \cap \bar{F})$ . ■

La méthode lift-and-project telle qu'elle est décrite ici ne permet vraisemblablement pas de concevoir des algorithmes efficaces, nous verrons plus loin comment elle peut être adaptée à cette fin. Son inefficacité procède de la taille des matrices qui croît avec le carré du nombre des variables et du fait qu'elle travaille sur les cônes  $M(K)$  et  $M^+(K)$  dont la description n'est pas aisée. En effet même si  $M(K)$  est un cône polyédral pour peu que  $K$  soit un cône polyédral,  $U_{psd}$  n'étant pas polyédral  $M^+(K)$  ne l'est pas a priori.

### 5.1.1 Détermination d'inégalités valides

Nous avons dit qu'une source d'inefficacité de la méthode telle qu'elle est exposée dans la section précédente peut être la description des cônes  $M(K)$  et  $M^+(K)$ , nous allons voir comment on peut obtenir une meilleure description de ces cônes en générant des inégalités valides.

Générer une inégalité valide pour un cône  $K$  c'est déterminer une inéquation de la forme  $\alpha^t x \geq 0$  qui est vérifiée pour tous les vecteurs  $x \in K$ . La définition que nous avons donnée de cône polaire permet de dire que déterminer une inégalité pour un cône  $K$  valide revient à chercher un vecteur  $\alpha \in K^*$ . Nous allons donc chercher à identifier les cônes polaires de  $N(K)$  et  $N^+(K)$ . Nous avons défini  $N(K)$ , respectivement  $N^+(K)$ , comme l'ensemble  $\{Y e_0, Y \in M(K)\}$ , respectivement  $\{Y e_0, Y \in M^+(K)\}$ . Cette définition permet d'énoncer le lemme suivant :

**Lemme 5.7**  $N(K)^* e_0^t = M(K)^*$  et  $N^+(K)^* e_0^t = M^+(K)^*$ .

Preuve :

$$\begin{aligned} \text{Par définition du cône polaire } w \in N(K)^* &\iff \forall y \in N(K) \quad w^t y \geq 0 \\ \text{Par définition de } N(K) &\iff \forall Y \in M(K) \quad w^t (Y e_0) \geq 0 \\ &\iff \forall Y \in M(K) \quad w^t (Y e_0) \geq 0 \text{ et } e_0^t Y w \geq 0 \\ &\iff \forall Y \in M(K) \quad Y \cdot e_0 w^t \geq 0 \text{ et } Y \cdot w e_0^t \geq 0 \\ &\iff \forall Y \in M(K) \quad Y \cdot \frac{e_0 w^t + w e_0^t}{2} \geq 0 \\ &\iff e_0 w^t + w e_0^t \in M(K)^* \end{aligned}$$

Le deuxième item du lemme se démontre de manière identique. ■

Nous allons donc poursuivre en définissant les cônes polaires  $M(K)^*$  et  $M^+(K)^*$ . Mais avant nous allons voir ce qu'est le cône polaire de l'intersection de deux cônes.

**Lemme 5.8** *Soit  $K_1$  et  $K_2$  deux cônes fermés alors  $(K_1 \cap K_2)^* = K_1^* + K_2^*$ .*

Preuve :

1.  $K_1^* + K_2^* \subseteq (K_1 \cap K_2)^*$

Considérons un vecteur  $x = k_1^* + k_2^* \in K_1^* + K_2^*$ . Soit  $y \in K_1 \cap K_2$ , formons le produit scalaire  $y^t x$  :

$$y^t x = y^t k_1^* + y^t k_2^*$$

Ce qui montre que  $x \in K_1^* + K_2^* \implies x \in (K_1 \cap K_2)^*$

2.  $(K_1 \cap K_2)^* \subseteq K_1^* + K_2^*$

Etant donné que  $K_1 \subseteq K_1 + K_2$  et  $K_2 \subseteq K_1 + K_2$  on peut dire que  $(K_1 \cap K_2) \subseteq K_1 + K_2$  et surtout :

$$\left. \begin{array}{l} K_1 \subseteq K_1 + K_2 \implies (K_1 + K_2)^* \subseteq K_1^* \\ K_2 \subseteq K_1 + K_2 \implies (K_1 + K_2)^* \subseteq K_2^* \end{array} \right\} \implies (K_1 + K_2)^* \subseteq K_1^* \cap K_2^*$$

Ce qui se réécrit de la manière suivante si on remplace  $K_1$  par  $K_1^*$  et  $K_2$  par  $K_2^*$  :

$$(K_1^* + K_2^*)^* \subseteq K_1^{**} \cap K_2^{**} = K_1 \cap K_2$$

Soit encore en prenant le polaire des parties gauche et droite de l'expression ci-dessus :

$$(K_1^* + K_2^*)^{**} = K_1^* + K_2^* \supseteq (K_1 \cap K_2)^*$$

On définit  $U_1$  l'espace vectoriel des matrices carrées d'ordre  $n + 1$ ,  $[w_{ij}]$ ,  $i, j \in \{0, \dots, n\}$  telles que  $w_{00} = 0$ ,  $w_{0j} = -w_{jj}$ ,  $\forall j \in \{1, \dots, n\}$  et  $w_{ij} = 0$  si  $i \neq 0$  et  $i \neq j$ . On remarquera que  $U_1$  admet pour base l'ensemble des matrices  $f_i e_i^t$ ,  $i \in \{1, \dots, n\}$ .

**Lemme 5.9** *L'espace vectoriel  $U_1$  est orthogonal à l'espace vectoriel des matrices  $Y$  vérifiant  $Y e_0 = \text{diag}(Y)$ .*

Preuve :

Soit  $M \in U_1$  et  $Y$  une matrice telle que  $Y e_0 = \text{diag}(Y)$ , formons le produit scalaire de ces deux matrices, il vient alors :

$$\begin{aligned} M \cdot Y &= \sum_{j=1}^n m_{0j} y_{0j} + \sum_{j=1}^n m_{jj} y_{jj} && \text{car tous les autres coefficients de } M \text{ sont nuls} \\ &= \sum_{j=1}^n (m_{0j} + m_{jj}) y_{0j} && \text{car } y_{0j} = y_{jj} \text{ par définition de } Y \\ &= 0 && \text{car } m_{0j} = -m_{jj} \text{ par définition de } M \end{aligned}$$

On note  $U_{asym}$  l'espace vectoriel des matrices antisymétriques,  $M = -M^t$  qui est naturellement orthogonal à l'espace des matrices symétriques. Alors on peut énoncer le théorème suivant :

**Théorème 5.2** *Les cônes polaires de  $M(K)$  et  $M^+(K)$  sont définis par :*

1.  $M(K)^* = U_1 + U_{asym} + \text{cône}\{uv^t, u \in K^*, v \in Q^*\}$ .
2.  $M^+(K)^* = U_1 + U_{asym} + U_{psd} + \text{cône}\{uv^t, u \in K^*, v \in Q^*\}$ .

Preuve :

1. Par construction  $M(K) = \text{cône}\{Y \mid \text{diag}(Y) = Ye_0\} \cap \text{cône}\{Y \mid Y = Y^t\} \cap \text{cône}\{Y \mid u^t Y v \geq 0, \forall u \in K^*, \forall v \in Q^*\}$ , aussi par les lemmes 5.8 et 5.9 peut on écrire :

$$M(K)^* = U_1 + U_{asym} + (\text{cône}\{Y \mid u^t Y v \geq 0, \forall u \in K^*, \forall v \in Q^*\})^*$$

Maintenant en notant que  $u^t Y v = uv^t \cdot Y$  on voit que  $(\text{cône}\{Y \mid u^t Y v \geq 0, \forall u \in K^*, \forall v \in Q^*\})^* = \text{cône}\{uv^t, u \in K^*, v \in Q^*\}$ . Ainsi la partie 1 du théorème est acquise.

2. La partie 2 se démontre de la même façon et tenant compte en plus du fait que  $U_{psd}$  est autopolaire.

Maintenant nous pouvons nous intéresser aux conditions d'appartenance d'un vecteur à  $N(K)^*$  et  $N^+(K)^*$ .

**Théorème 5.3 (Lovász et Schrijver)**

*Soit  $K \subset Q$  un cône de  $\mathbb{R}^{n+1}$  et  $w \in \mathbb{R}^{n+1}$ .*

1.  $w \in N(K)^*$  si et seulement si il existe  $n$  vecteurs  $a_1, \dots, a_n \in K^*$ , un réel  $\lambda$  et une matrice antisymétrique  $A$  tels que  $a_i + \lambda e_i + Ae_i \in K^*, \forall i \in \{1, \dots, n\}$  et

$$w = \sum_{i=1}^n a_i + A\mathbf{1}.$$

2.  $w \in N^+(K)^*$  si et seulement si il existe  $n$  vecteurs  $a_1, \dots, a_n \in K^*$ , un réel  $\lambda$ , une matrice antisymétrique  $A$  et une matrice semi-définie positive  $B$  tels que  $a_i + \lambda e_i +$

$$Ae_i + Be_i \in K^*, \forall i \in \{1, \dots, n\} \text{ et } w = \sum_{i=1}^n a_i + A\mathbf{1} + B\mathbf{1}.$$

Preuve :

Si  $w \in N(K)^*$  alors  $we_0^t \in M(K)^*$ . On a vu que :

$$M(K)^* = U_1 + U_{asym} + \text{cône}\{xy^t, x \in K^*, y \in Q^*\}$$

Par ailleurs on sait que  $U_1$  est g n r  par les matrices de la forme  $e_i f_i^t$ ,  $i \in \{1, \dots, n\}$ , aussi peut-on  crire :

$$we_0^t = \sum_i x_j y_j^t + \sum_{i=1}^n \lambda_i e_i f_i^t + A, \quad x_j \in K^*, \quad y_j \in Q^*, \quad A \in U_{asym}$$

De plus on sait que  $Q^* = \text{c\^one}\{e_i, f_i\}$ , d'o  l'expression de  $we_0^t$  si on  crit  $y_j$  comme combinaison lin aire des  $e_i$  et des  $f_i$  :

$$we_0^t = \sum_{i=1}^n x_i e_i^t + \sum_{i=1}^n \bar{x}_i f_i^t + \sum_{i=1}^n \lambda_i e_i f_i^t + A, \quad x_i \in K^*, \quad \bar{x}_i \in K^*$$

En multipliant l'expression ci-dessus par  $e_j$  par la droite on obtient :

$$0 = x_j - \bar{x}_j - \lambda_j e_j + A e_j$$

Maintenant en multipliant l'expression de  $we_0^t$  par  $e_0$  on obtient :

$$\left. \begin{array}{l} w = \sum_{i=1}^n \bar{x}_i + \sum_{i=1}^n \lambda_i e_i + A e_0 \\ \bar{x}_i = x_i - \lambda_i e_i + A e_i \end{array} \right\} \Rightarrow w = \sum_{i=1}^n x_i + \sum_{i=1}^n A e_i + A e_0 = \sum_{i=1}^n x_i + A \mathbf{1}$$

Et  $\bar{x}_i = x_i - \lambda_i e_i + A e_i \in K^*$ . Comme  $e_i \in K^*$  cette condition demeure valide si on fait d cro tre  $\lambda_i$  jusqu'   $\lambda$ . La n cessit  de la condition est ainsi  tablie, sa suffisance s' tablit de fa on semblable.

Le point num ro 2 du th or me se d montre comme le point num ro 1. ■

Le th or me pr c dent nous fournit les moyens de s parer sur les c nes  $N(K)$  et  $N^+(K)$  et donc de concevoir des algorithmes bas s sur des m thodes de coupes. Maintenant que nous savons s parer sur les c nes  $N(K)$  et  $N^+(K)$  nous nous int ressons au fait de savoir quand cette s paration peut s'effectuer en temps polynomial.

On appelle oracle de s paration sur  $K$  une routine s'ex cutant en temps polynomial qui,  tant donn  un vecteur rationnel  $x$ , retourne une des r ponses suivantes :

- $x \in K$
- un vecteur  $w \in K^*$  tel que  $w^t x < 0$

#### **Th or me 5.4 (Lov sz et Schrijver)**

*Si nous disposons d'un oracle de s paration sur  $K$  alors le probl me de s paration sur  $N(K)$ , comme celui sur  $N^+(K)$ , peut  tre r solu en temps polynomial.*

Preuve :

Par hypoth se nous poss dons un oracle de s paration pour  $K$ , montrons qu'alors nous poss dons un algorithme polynomial pour le c ne  $M(K)$ .

Les conditions  $Y = Y^t$  et  $\text{diag}(Y) = Y e_0$  peuvent  tre v rifi es en temps polynomial, de m me on trouve ais ment un hyperplan de s paration, par exemple si  $Y \neq Y^t$  il suffit de prendre une matrice antisym trique  $A$  telle que  $A \cdot Y < 0$ .

Pour la s paration sur la condition  $u^t Y v \geq 0$ ,  $\forall u \in K^*$ ,  $\forall v \in Q^*$  on a d j  vu que cette condition est  quivalente    $Y v \in K$ ,  $\forall v \in Q^*$ . Or on sait que  $Q^* = \text{c\^one}\{e_i, f_i\}$ , il suffit



donc de vérifier que  $Ye_i \in K$  et  $Yf_i \in K$  ce qui peut être fait en utilisant l'oracle de séparation pour  $K$ . Etant donné que  $N(K) = M(K)e_0$  on peut affirmer que le problème de séparation sur  $N(K)$  peut être résolu en temps polynomial.

Pour le problème de séparation sur  $N^+(K)$  il suffit de vérifier qu'en plus de ce qui a été vu pour  $N(K)$  on peut séparer en temps polynomial sur  $U_{psd}$ . Vérifier si une matrice  $A \in U_{psd}$  peut se faire en temps polynomial via la résolution du programme semi-défini positif  $Min\{A \cdot M, M \in U_{psd}\}$ , si le résultat de ce programme est strictement négatif alors  $A \notin U_{psd}$  et la matrice  $M$  donne l'équation d'un hyperplan séparant  $A$  de  $U_{psd}$ . ■

## 5.2 Un lift-and-project séquentiel

Nous présentons ici un schéma séquentiel de lift-and-project. Dans ce schéma les domaines des variables sont convexifiés les uns après les autres. Cette procédure séquentielle de convexification d'un ensemble de points à coordonnées entières est exposée dans [6]. Cette méthode est issue des résultats concernant la convexification d'une disjonction de polyèdres. Comme précédemment on définit  $K^0$  l'enveloppe convexe des points de l'ensemble défini par :

$$Plne = \left\{ \begin{array}{l} [-b, A](1, x^t)^t \geq 0 \\ x \in \{0, 1\}^n \end{array} \right.$$

$K$  le cône défini par la relaxation linéaire du système de  $Plne$  qui sera notée  $[-\tilde{b}, \tilde{A}](1, x^t)^t \geq 0$ . On définit alors une procédure en quatre étapes :

1. Choisir une valeur de l'indice  $j \in \{1, \dots, n\}$ .
2. Multiplier  $\tilde{A}x - \tilde{b} \geq 0$  par  $(1 - x_j)$  et  $x_j$  de manière à obtenir le système suivant :

$$\left\{ \begin{array}{ll} (1 - x_j) & (\tilde{A}x - \tilde{b} \geq 0) \\ x_j & (\tilde{A}x - \tilde{b} \geq 0) \end{array} \right.$$

3. Linéariser le système précédent en remplaçant :

- $x_i x_j, i \in \{1, \dots, n\}, i \neq j$  par  $y_j$ .
- $x_j^2$  par  $x_j$ .

Soit  $M_j(K)$  le polyèdre ainsi obtenu.

4.  $P_j(K)$  est la projection de  $M_j(K)$  sur l'espace des  $x$ .

**Remarque :** La linéarisation de l'étape 3 fournit entre autres les contraintes suivantes :

- $(0 \leq x_i)x_j \implies y_i \geq 0$
- $(0 \leq x_i)(1 - x_j) \implies y_i \leq x_i$
- $(x_i \leq 1)x_j \implies y_i \leq x_j, i \neq j$
- $(x_i \leq 1)(1 - x_j) \implies x_i + x_j - 1 \leq y_i, i \neq j$

Ces contraintes sont à rapprocher de celles vues sur les coefficients de la matrice  $Y$  à la section 3.3. D'autre part on note que le remplacement de  $x_j^2$  par  $x_j$  produit l'élimination de points fractionnaires.

La projection de l'étape quatre sur l'espace des  $x$  peut être définie comme suit, après avoir écrit  $M_j(K) = \{(x, y) \mid Bx + Dy \geq d\}$ :

$$P_j(K) = Proj_x(M_j(K)) = \{x \mid u^t D x \geq u^t d, \forall u \in Extr(W)\}$$

où  $Extr(W)$  désigne l'ensemble des rayons extrêmes du cône  $W$  défini par  $W = \{u \mid u^t B = 0, u \geq 0\}$ .

Nous allons voir au travers du théorème suivant comment la procédure énoncée précédemment réalise une convexification partielle.

**Théorème 5.5 (Balas, Ceria et Cornuéjols)**

$$P_j(K) = conv(K \cap \{x \in \mathbb{R}^n \mid x_j \in \{0, 1\}\})$$

Preuve :

1.  $P_j(K) \subseteq conv(K \cap \{x \in \mathbb{R}^n \mid x_j \in \{0, 1\}\})$ .

Supposons que  $K \cap \{x \mid x_j = 0\} = \emptyset$  alors  $x_j \geq \epsilon$  est une inégalité valide pour  $K$  pour certaine valeur de  $\epsilon$ . Par définition de  $M_j(K)$  l'inégalité  $(1 - x_j)(x_j - \epsilon) \geq 0$  est satisfaite pour tout  $x$  qui satisfait :

$$(1) = \begin{cases} (1 - x_j) & (\tilde{A}x - \tilde{b} \geq 0) \\ x_j & (\tilde{A}x - \tilde{b} \geq 0) \end{cases}$$

Aussi peut-on écrire :

$$\begin{aligned} (1 - x_j)(x_j - \epsilon) \geq 0 & \iff x_j - \epsilon - x_j^2 + \epsilon x_j \geq 0 \\ & \iff -\epsilon + \epsilon x_j \geq 0 && \text{en remplaçant } x_j^2 \text{ par } x_j \\ & \iff x_j \geq 1 \end{aligned}$$

$x_j \geq 1$  est alors une inégalité valide pour  $M_j(K)$  et  $P_j(K)$  et donc on a  $P_j(K) \subseteq conv(K \cap \{x \mid x_j = 1\})$  et par voie de conséquence  $P_j(K) \subseteq conv(K \cap \{x \in \mathbb{R}^n \mid x_j \in \{0, 1\}\})$ . D'une manière analogue on établit que si  $K \cap \{x \mid x_j = 1\} = \emptyset$  alors  $x_j \leq 0$  est une inégalité valide pour  $M_j(K)$  et  $P_j(K)$  et donc  $P_j(K) \subseteq conv(K \cap \{x \in \mathbb{R}^n \mid x_j \in \{0, 1\}\})$ .

Supposons maintenant que  $K \cap \{x \mid x_j = 0\}$  et  $K \cap \{x \mid x_j = 1\}$  sont tous deux non vides. Soit  $\alpha^t x \geq \beta$  une inégalité valide pour  $conv(K \cap \{x \in \mathbb{R}^n \mid x_j \in \{0, 1\}\})$ . Etant donné que  $\alpha^t x \geq \beta$  est valide pour  $K \cap \{x \mid x_j = 0\}$  il existe  $\lambda \geq 0$  tel que  $\alpha^t x + \lambda x_j \geq \beta$  est valide pour  $K$ , d'une manière symétrique puisque  $\alpha^t x \geq \beta$  est valide pour  $K \cap \{x \mid x_j = 1\}$  il existe  $\mu \geq 0$  tel que  $\alpha^t x + \mu(1 - x_j) \geq \beta$  est valide pour  $K$ . Puisque  $\alpha^t x + \lambda x_j \geq \beta$  et  $\alpha^t x + \mu(1 - x_j) \geq \beta$  sont valides pour  $K$  les inégalités :

$$\begin{aligned} (1 - x_j)(\alpha^t x + \lambda x_j - \beta) & \geq 0 \\ x_j(\alpha^t x + \mu(1 - x_j) - \beta) & \geq 0 \end{aligned}$$

sont satisfaites par tout  $x$  qui vérifie (1). En sommant les deux inégalités ci-dessus et en remplaçant  $x_j^2$  par  $x_j$  on obtient  $\alpha^t x \geq \beta$ . Ce qui montre que cette dernière est valide tant pour  $P_j(K)$  que pour  $M_j(K)$  et comme pour les deux points précédents on peut conclure que  $P_j(K) \subseteq \text{conv}(K \cap \{x \in \mathbb{R}^n \mid x_j \in \{0, 1\}\})$ .

2.  $P_j(K) \supseteq \text{conv}(K \cap \{x \in \mathbb{R}^n \mid x_j \in \{0, 1\}\})$ .

Soit  $x^* \in K \cap \{x \mid x_j \in \{0, 1\}\}$ . Relativement à  $x^*$  on peut appliquer les opérations des étapes 2 et 3, notons  $y_i^* = x_i^* x_j^*$  la variable de linéarisation de l'étape 2 alors puisque  $x^* \in K$  ( $x^* y^* \in M_j(K)$ ) par définition de celui-ci. D'autre part par définition de la projection  $x^* \in P_j(K)$ , aussi-a-t on établi que  $\text{conv}(K \cap \{x \in \mathbb{R}^n \mid x_j \in \{0, 1\}\}) \subseteq P_j(K)$ . ■

Avant de poursuivre la construction de l'enveloppe convexe,  $K^0$ , des points à coordonnées entières du domaine défini par *Plne* nous allons voir un lemme qui est à rapprocher du lemme 5.3.

**Lemme 5.10** *Soit  $S \in \mathbb{R}^n$  et  $H = \{x \in \mathbb{R}^n \mid \alpha^t x = \beta\}$  un hyperplan tel que  $\alpha^t x \geq \beta$  est vérifiée pour tout  $x \in S$ . Alors  $H \cap \text{conv}(S) = \text{conv}(H \cap S)$ .*

Preuve :

Dire que  $x \in H \cap \text{conv}(S)$  est équivalent à dire :

$$\begin{array}{ll} \alpha^t x = \beta & \text{car } x \in H \\ x = \sum_{i \in I} \lambda_i y_i, y_i \in S, \sum_{i \in I} \lambda_i = 1, \lambda_i \geq 0 & \text{car } x \in \text{conv}(S) \end{array}$$

Dire que  $y_i \in S$  est équivalent à dire  $\alpha y_i \geq \beta$ , par définition de  $H$ . Par ailleurs  $x = \sum_{i \in I} \lambda_i y_i$

et donc  $\alpha^t x = \sum_{i \in I} \alpha \lambda_i y_i = \beta$  ce qui permet de dire que  $\alpha y_i = \beta$  donc  $y_i \in H \cap S$ ,  $x$  peut donc

s'écrire comme combinaison linéaire convexe d'éléments de  $H \cap S$  et donc  $x \in \text{conv}(H \cap S)$ . ■

Comme pour la construction de Lovász et Schrijver nous avons une définition récursive de la construction de  $K^0$ , celle-ci est précisée par le théorème suivant.

**Théorème 5.6 (Balas, Ceria et Cornuéjols)**

*On définit récursivement  $P_t(K)$  par  $P_0(K) = K$  et  $P_t(K) = P_t(P_{t-1}(K))$ . Alors  $P_n(K) = K^0$ .*

Preuve :

La preuve de ce théorème va se faire par induction sur  $t$ . Il est clair que pour  $t = 1$  le théorème est équivalent au théorème 5.5.

Soit  $F_t = \{x \mid x_j \in \{0, 1\}, \forall j \in \{1, \dots, t\}\}$ . Supposons le théorème vrai jusqu'à l'indice  $n - 1$ , par l'hypothèse d'induction on a :

$$P_n(K) = P_n(\text{conv}(K \cap F_{n-1}))$$

En appliquant le résultat du théorème 5.5 on peut écrire que :

$$P_n(\text{conv}(K \cap F_{n-1})) = \text{conv}(\text{conv}(K \cap F_{n-1}) \cap \{x \mid x_n \in \{0, 1\}\})$$

Ce qui peut se réécrire comme suit si on remarque que  $\{x \mid x_n \in \{0, 1\}\} = \{x \mid x_n = 0\} \cup \{x \mid x_n = 1\}$  :

$$P_n(\text{conv}(K \cap F_{n-1})) = \text{conv}([\text{conv}(K \cap F_{n-1}) \cap \{x \mid x_n = 0\}] \cup [\text{conv}(K \cap F_{n-1}) \cap \{x \mid x_n = 1\}])$$

Et en utilisant le résultat du lemme 5.10 on peut écrire :

$$\begin{aligned} P_n(\text{conv}(K \cap F_{n-1})) &= \text{conv}((K \cap F_{n-1} \cap \{x \mid x_n = 0\}) \cup (K \cap F_{n-1} \cap \{x \mid x_n = 1\})) \\ &= \text{conv}(K \cap F_{n-1} \cap \{x \mid x_n \in \{0, 1\}\}) \\ &= \text{conv}(K \cap F_n) \end{aligned}$$

Ainsi le résultat du théorème est acquis. ■

### 5.2.1 Détermination d'inégalités valides

Afin de pouvoir concevoir un algorithme nous allons voir comment on peut obtenir une description de  $P_j(K)$  sous forme d'inégalités linéaires. Comme pour les opérateurs  $N(K)$  et  $N^+(K)$  ceci va nous conduire à nous intéresser aux conditions d'appartenance d'un vecteur à son cône polaire que nous noterons  $P_j(K)^*$ .

#### Théorème 5.7 (Balas, Ceria et Cornuéjols)

$P_j(K) = \{x \in \mathbb{R}^n \mid \alpha^t x \geq \beta, \forall (\alpha, \beta) \in P_j(K)^*\}$ , où  $P_j(K)^*$  est l'ensemble des vecteurs  $(\alpha, \beta) \in \mathbb{R}^{n+1}$  pour lesquels il existe des vecteurs  $u$  et  $v \in \mathbb{R}^{m+2n}$ ,  $u \geq 0$  et  $v \geq 0$  ainsi que des scalaires  $u_0$  et  $v_0$  vérifiant :

$$\begin{array}{rcl} \alpha & -u^t \tilde{A} & +u_0 e_j & = 0 \\ \alpha & & -v^t \tilde{A} & -v_0 e_j & = 0 \\ & u^t \tilde{b} & & & = \beta \\ & & v^t \tilde{b} & +v_0 & = \beta \end{array}$$

Preuve :

On a vu que par construction  $P_j(K) = \text{Proj}_x(M_j(K))$ , où  $M_j(K)$  est défini par le système :

$$\left. \begin{array}{l} \tilde{A}x + \tilde{b}x_j - \tilde{A}y - \tilde{b} \geq 0 \\ -\tilde{b}x_j + \tilde{A}y \geq 0 \end{array} \right\} \iff Dx + By \geq d$$

avec  $D = [\tilde{A}_j, -\tilde{0}_j]^t$ ,  $\tilde{A}_j$  étant la matrice obtenue en ajoutant  $\tilde{b}$  à la colonne  $j$  de  $\tilde{A}$ ,  $\tilde{0}_j$  est définie de manière analogue.  $B = [-\tilde{A}, \tilde{A}]^t$  et  $d = (\tilde{b}, 0)^t$ .

La projection de  $M_j(K)$  sur l'espace des  $x$  est donnée par :

$$\text{Proj}_x(M_j(K)) = \{x \mid (u^t, v^t)Dx \geq (u^t, v^t)d, \forall (u, v)^t \in \text{Extr}(W)\}$$

Dire que  $(u, v)$  est un rayon extrême du cône engendré par les lignes de  $B$  c'est dire que :

$$(u, v) \in \text{Extr}(W) \iff (u^t, v^t)B = 0 \iff -u^t \tilde{A} + v^t \tilde{A} = 0 \quad (5.5)$$

par ailleurs :

$$x \in P_j(K) \iff (u^t, v^t)Dx = u^t \tilde{A}x + u^t \tilde{b}e_j x - v^t \tilde{b}e_j x \geq (u^t, v^t)d = u^t \tilde{b} \quad (5.6)$$

En posant  $u_0 = -u^t \tilde{b} + v^t \tilde{b}$  l'équation 5.6 se réécrit comme suit :

$$(u^t \tilde{A} - u_0 e_j)x \geq u^t \tilde{b}$$

Donc  $x \in P_j(K)$  vérifie  $\alpha^t x \geq \beta$  avec  $\alpha = (u^t \tilde{A} - u_0 e_j)$  et  $\beta = u^t \tilde{b}$ . Aussi la première des équations définissant  $P_j(K)^*$  est satisfaite ainsi que la troisième. Si on pose  $u_0 = v_0$  alors comme  $u^t \tilde{A} - v^t \tilde{A} = 0$  on obtient  $\alpha = (u^t \tilde{A} - u_0 e_j) = (v^t \tilde{A} - v_0 e_j)$  ce qui justifie la deuxième des équations.

D'autre part on a posé  $u_0 = -u^t \tilde{b} + v^t \tilde{b} = v_0$  ce qui conduit directement à  $v_0 + v^t \tilde{b} = u^t \tilde{b} = \beta$ . ■

Maintenant que nous savons traiter le problème de séparation sur  $P_j(K)$  il est intéressant de voir dans quels cas les inégalités valides générées sont des facettes de  $P_j(K)$ .

**Théorème 5.8 (Balas, Ceria et Cornuéjols)**

*Si  $K$  est un polyèdre de pleine dimension,  $\{x \mid x_j = 0\}$  et  $\{x \mid x_j = 1\}$  sont non vides, alors pour toute constante  $\beta_0 \neq 0$ ,  $\alpha^t x \geq \beta_0$  est une facette de  $P_j(K)$  si et seulement si  $\alpha$  est point extrême de  $P_j(K)_{\beta_0}^*$  le polyèdre obtenu depuis  $P_j(K)^*$  en posant  $\beta = \beta_0$ .*

Preuve :

Dire que  $\alpha$  est un point extrême de  $P_j(K)_{\beta_0}^*$  c'est dire qu'il est l'intersection de  $n$  hyperplans distincts deux à deux. On peut donc trouver  $n$  vecteurs  $\alpha_1, \dots, \alpha_n$ , un par hyperplan, qui sont linéairement indépendants et qui vérifient  $\alpha_i x \geq \beta, \forall x \in P_j(K), i \in \{1, \dots, n\}$ . Considérons les systèmes d'équations linéaires :

$$\begin{array}{llll} \alpha_1 x = \beta_0 & \alpha_1 x = \beta_0 + \epsilon_1^k & \alpha_1 x = \beta_0 + \epsilon_1^n & \\ \alpha_2 x = \beta_0 + \epsilon_2^1 & \alpha_k x = \beta_0 & & \\ & & \alpha_{n-1} x = \beta_0 + \epsilon_{n-1}^n & \\ \alpha_n x = \beta_0 + \epsilon_n^1 & \alpha_n x = \beta_0 + \epsilon_n^k & \alpha_1 x = \beta_0 & \end{array}$$

L'indépendance linéaire des vecteurs  $\{\alpha_1, \dots, \alpha_n\}$  nous assure que chacun de ces systèmes possède une solution unique. Soit  $\{x_1, \dots, x_n\}$  les solutions respectives de chacun de ces systèmes. Si on choisit les  $\epsilon_i^j$  tels que les seconds membres sont affinement indépendants, les vecteurs  $\{x_i, i \in \{1, \dots, n\}\}$  sont aussi affinement indépendants. D'autre part on a  $\alpha^t x_i \geq \beta_0$  par définition de  $P_j(K)_{\beta_0}^*$ . C'est à dire qu'on a trouvé  $n$  vecteurs de  $P_j(K)$  affinement indépendants qui vérifient  $\alpha^t x \geq \beta_0$ ; donc cette inégalité définit une facette de  $P_j(K)$ . ■

Pour traiter le problème de séparation sur  $P_j(K)$  on est donc amené à résoudre un problème d'optimisation sur  $P_j(K)^*$ , par exemple si on veut trouver un hyperplan séparant un vecteur  $x$  de  $P_j(K)$  on doit résoudre un problème de la forme :

$$\text{Max} \{-x^t \alpha + \beta \mid (\alpha, \beta) \in P_j(K)^*\}$$

Or il se peut que le problème d'optimisation ci-dessus ne possède pas un optimum borné, pour pallier ce problème on va introduire un ensemble de normalisation dont le but est de produire une troncature de  $P_j(K)^*$  assurant au programme de maximisation un optimum borné. Le choix de cet ensemble de normalisation peut s'effectuer suivant divers critères et dépendamment du type de problème à traiter, ceci est très largement discuté dans [6], [7] et [17]. Nous ne présentons donc qu'une définition de l'ensemble de normalisation, celle-ci est universelle contrairement à d'autres qui réclament que le problème ait certaines caractéristiques. De plus c'est celle que nous avons utilisée pour notre implémentation de la méthode.

**Théorème 5.9** *Les rayons extrêmes de  $P_j^*(K)$  sont en bijection avec les sommets de  $P_j^*(K) \cap S$  si  $S = \{(u, v) \in \mathbb{R}^{2(n+m+p+1)} \mid \sum_{i=0}^{n+m+p} u_i + \sum_{i=0}^{n+m+p} v_i = 1\}$ .*

Preuve :

Etant donné que la direction de normalisation n'appartient pas à  $P_j^*(K)$ , celle-ci ne supprime aucun des rayons extrêmes. Si  $r$  est un rayon extrême de  $P_j^*(K)$  il ne peut pas s'écrire comme combinaison linéaire convexe de vecteurs de  $P_j^*(K)$ , et donc a fortiori de vecteurs  $P_j^*(K) \cap S$  car  $P_j^*(K) \cap S \subseteq P_j^*(K)$ , aussi par définition on peut dire que c'est un sommet de  $P_j^*(K) \cap S$ .

Maintenant considérons un sommet de  $v \in P_j^*(K) \cap S$  qui ne soit pas un rayon extrême de  $P_j^*(K)$  alors  $v$  peut s'écrire comme combinaison linéaire convexe de rayons extrêmes de  $P_j^*(K)$  et donc comme combinaison linéaire convexe de sommets de  $P_j^*(K) \cap S$ , d'où une contradiction. ■

Le théorème ci-dessus montre que tronquer  $P_j^*(K)$  par un tel ensemble de normalisation  $S$  préserve les rayons extrêmes, aussi par le théorème 5.8 nous pouvons dire qu'on peut atteindre tous les hyperplans supports de  $P_j^*(K)$ .

## Chapitre 6

# Applications de lift-and-project

Nous présentons ici deux applications de la méthode de lift-and-project. Dans la première section nous présentons une application de la construction de Lovász et Schrijver au problème du stable de poids maximum, cette étude permet d'aborder des points théoriques et de les illustrer par un exemple. Puis dans la deuxième partie de ce chapitre nous présentons un algorithme de coupes polyédrales dérivé de la construction de Lovász et Schrijver vue précédemment.

### 6.1 Le problème du stable de cardinalité maximum

Dans cette section nous présentons une application de la méthode de lift-and-project au problème du stable de poids maximum. La motivation de cette présentation est l'étude d'un rang pour les opérateurs  $N(K)$  et  $N^+(K)$  en se posant les questions :

- Quelle est la plus petite valeur de  $r$  telle que  $N^r(K)$ ,  $N^{+,r}(K)$ , décrit l'enveloppe convexe des stables d'un graphe  $G$ ?
- Pour quelles classes de graphe  $N^r(K)$ ,  $N^{+,r}(K)$ , décrit-il l'enveloppe convexe des stables?
- Quelles sont les classes d'inégalités qui sont vérifiées lorsqu'on a appliqué récursivement  $r$  fois l'opérateur  $N(K)$ ,  $N^+(K)$ ?

Les réponses à ces questions sont à la fois une mesure de la complexité du problème pour une classe d'instance et de l'efficacité de la méthode pour certaines instances du problème.

**Définition 6.1** *Le  $N$ -index d'une inégalité est la plus petite valeur de  $r$  pour laquelle les points de  $N^r(K)$  la satisfont.*

Le  $N$ -index d'un graphe est la plus grande des valeurs de  $N$ -index des facettes du polyèdre du stable de ce graphe. Nous établirons, quand nous le pourrons une comparaison entre la valeur du  $N$ -index et la valeur du rang de Chvátal, voir [72] pages 226 et 227.

Avant de développer cette étude nous allons compléter notre description du problème du stable. Nous avons vu précédemment que les stables d'un graphe  $G = (X, E)$  sont les solutions entières du système d'inéquations suivant :

$$\left. \begin{array}{l} x_i + x_j \leq 1 \quad \forall (ij) \in E \\ x_i \geq 0 \quad \forall i \in X \end{array} \right\} \quad (6.1)$$

Nous appellerons  $Frac(G)$  les points extrêmes du polytope défini par le système ci-dessus.  $Stab(G)$  désignera l'enveloppe convexe des stables du graphe  $G$ .

**Théorème 6.1**  $Frac(G) = Stab(G)$  si et seulement si  $G$  est un graphe biparti.

Preuve :

1.  $G$  est un graphe biparti alors  $Frac(G) = Stab(G)$ . Pour montrer ceci il nous suffit de montrer que si  $G$  est un graphe biparti les points extrêmes de  $Frac(G)$  sont à coordonnées entières, ce qui est immédiat car la matrice d'incidence sommets arêtes d'un graphe biparti est totalement unimodulaire.
2.  $Frac(G) = Stab(G)$  alors  $G$  est un graphe biparti. Considérons un graphe qui n'est pas biparti donc comportant un trou impair  $H$ . Alors  $x_i = \frac{1}{2} \forall i \in H$  et  $x_i = 0 \forall i \in G - H$  est un sommet de  $Frac(G)$ , ce qui montre que  $Frac(G) \neq Stab(G)$ . ■

**Corollaire 6.1** Le  $N$ -index d'un graphe  $G$  est égal à 0 si et seulement si  $G$  est un graphe biparti.

Si un graphe n'est pas un graphe biparti alors il contient au moins un cycle comportant un nombre impair de sommets, si ce cycle est sans corde on l'appellera trou impair. Il apparaît clairement que le nombre maximum de sommets d'un trou impair  $H$  figurant dans un ensemble stable est  $\frac{|H|-1}{2}$ . Ainsi on obtient une classe d'inégalités valides pour le problème du stable :

$$\sum_{i \in H} x_i \leq \frac{|H|-1}{2} \text{ pour tout trou impair } H \quad (6.2)$$

Les graphes pour lesquels les contraintes de types 6.1 et 6.2 suffisent à décrire le polyèdre des stables sont dits  $t$ -parfaits.

**Théorème 6.2 (Lovász et Schrijver)**

Le  $N$ -index d'un graphe  $G$   $t$ -parfait est égal à 1.

Avant de démontrer le résultat du théorème ci-dessus nous allons définir quelques opérations sur les graphes et adapter des résultats précédents au cas particulier du stable. Nous appellerons suppression d'un sommet  $v$  l'opération qui consiste à considérer le graphe  $G - \{v\}$  au lieu du graphe  $G$ , la contraction d'un sommet  $v$  consiste en la suppression de ce sommet et de ceux qui lui sont adjacents,  $\Gamma(v)$ . Remarquons que la suppression du sommet  $v$  est équivalente au fait de fixer la variable  $x_v$  à 0 et que la contraction de ce sommet est équivalente au fait de fixer la variable  $x_v$  à 1. Soit  $a^t x \leq b$  une inégalité valide pour  $Stab(G)$ , notons  $a_{X-\{v\}}$  la restriction de  $a$  au sous-ensemble de sommets  $X - \{v\}$  et  $a_{X-\Gamma(v)-\{v\}}$  la restriction de  $a$  au sous-ensemble  $X - \Gamma(v) - \{v\}$ . On peut remarquer que  $a_{X-\{v\}}^t x \leq b$  et  $a_{X-\Gamma(v)-\{v\}}^t x \leq b$  sont respectivement valides pour  $Stab(G_{\{v\}})$  et  $Stab(G - \Gamma(v) - \{v\})$ . Réciproquement si  $a^t x \leq b$  est une inégalité telle que pour un sommet  $v$  tant la suppression que la contraction fournissent des inégalités valides le polyèdre des stables des graphes correspondants alors  $a^t x \leq b$  est une inégalité valide pour  $Stab(G)$ .

**Lemme 6.1** Soit  $K$  un ensemble convexe tel que  $Stab(G) \subseteq K \subseteq Frac(G)$ . Si  $a^t x \leq b$  est une inégalité telle que tant la suppression que la contraction d'un sommet  $v$  fournissent une inégalité valide pour  $K$  alors  $a^t x \leq b$  est une inégalité valide pour  $N(K)$ .



Preuve :

Ceci dérive directement du lemme 5.3 qui dit que  $N(K) \subseteq (K \cap H_v) + (K \cap G_v)$  où  $H_v$  et  $G_v$  désignent respectivement les hyperplans d'équations  $x_v = 0$  et  $x_v = 1$ . ■

Preuve du théorème 6.2 :

Pour prouver le théorème nous devons montrer que les contraintes de type 6.2 dites contraintes de trous impairs sont valides pour  $N(\text{Frac}(G))$  et que celui-ci les satisfait toutes

1. Les contraintes de trous impairs sont valides pour  $N(\text{Frac}(G))$ .

Considérons un trou impair  $H$  et la contrainte qui lui est associée :  $\sum_{v \in H} x_v \leq \frac{|H| - 1}{2}$ .

On peut remarquer aisément que la suppression d'un sommet quelconque  $v_0$  de  $H$  fournit une inégalité valide pour  $\text{Frac}(G)$ , en effet si on supprime  $v_0$  de  $H$  on obtient une chaîne comportant  $|H| - 2$  arêtes et  $|H| - 1$  sommets dont au plus un sur deux pourra figurer dans un ensemble stable. D'une manière analogue on peut vérifier que la contraction fournit une inégalité valide pour  $\text{Frac}(G)$ . Aussi par le lemme 6.1 nous pouvons assurer que la contrainte associée à  $H$  est valide pour  $N(\text{Frac}(G))$ .

2.  $N(\text{Frac}(G))$  satisfait l'ensemble des contraintes de trous impairs. Considérons un vecteur  $x \in \mathbb{R}^n \cap \text{Frac}(G)$  qui satisfait les contraintes de trous impairs. Notre problème est de montrer qu'il existe une matrice  $Y = (y_{ij})$  appartenant à  $M(\text{Frac}(G))$ , c'est à dire que cette matrice vérifie les propriétés suivantes :

- (a)  $Y = Y^t$
- (b)  $y_{i0} = y_{ii} = x_i, \forall i \in \{1, \dots, n\}$   $y_{00} = 1$
- (c)  $Y f_k \in \text{Frac}(G)$  et  $Y e_k \in \text{Frac}(G)$

La dernière des propriétés nous permet d'écrire la double inégalité suivante  $\forall i, j, k \in X$  tels que  $(ij) \in E$  :

$$x_i + x_j + x_k - 1 \leq y_{ik} + y_{jk} \leq x_k \quad (6.3)$$

Le lemme suivant donne un critère d'existence d'une solution au système ci-dessus.

**Lemme 6.2** Soit  $H = (W, F)$  un graphe où deux valeurs  $0 \leq a(ij) \leq b(ij)$  sont associées à chacune des arêtes. Soit  $U \subseteq W$  un sous-ensemble de sommets donné, alors le système :

$$\begin{aligned} a(ij) &\leq y_i + y_j \leq b(ij) && (ij) \in F \\ y_i &\geq 0 && i \in W \\ y_i &= 0 && i \in U \end{aligned}$$

n'a pas de solution si et seulement si il existe une séquence de sommets  $s_1, \dots, s_p$  telle que  $s_i$  et  $s_{i+1}$  sont adjacents et qu'au moins une des propositions ci-dessous est vraie :

- (a)  $p \equiv 1 \pmod{2}$  et  $b(s_0 s_1) - a(s_1 s_2) + b(s_2 s_3) - \dots + b(s_{p-1} s_p) < 0$
- (b)  $p \equiv 0 \pmod{2}$ ,  $s_0 = s_p$  et  $b(s_0 s_1) - a(s_1 s_2) + b(s_2 s_3) - \dots + b(s_{p-1} s_p) < 0$

- (c)  $p \equiv 0 \pmod{2}$ ,  $s_p \in U$  et  $b(s_0s_1) - a(s_1s_2 + b(s_2s_3) - \dots + b(s_{p-1}s_p)) < 0$   
 (d)  $p \equiv 1 \pmod{2}$ ,  $s_0$  et  $s_p \in U$  et  $-a(s_0s_1) + b(s_1s_2 - a(s_2s_3) + \dots - a(s_{p-1}s_p)) < 0$

Pour ce qui nous concerne l'ensemble des sommets  $W$  est l'ensemble des paires  $\{i, j\}$  et on dit que deux paires  $\{i, j\}$  et  $\{k, l\}$  sont adjacentes dans  $H$  si  $i = k$  et si  $(jl) \in E$ . Les arêtes de  $H$  sont valuées comme suit :  $a(ij, jk) = x_i + x_j + x_k - 1$  et  $b(ij, jk) = x_k$ . Il est montré dans [64] que si  $x$  satisfait les contraintes de trous impairs alors aucune des situations ci-dessus peut survenir. Ce qui permet de conclure à l'existence d'une matrice  $Y \in M(\text{Frac}(G))$  vérifiant la condition 6.3. ■

Le N-index des contraintes de trous impairs est égal au rang de Chvátal de ces contraintes. Il n'en est pas toujours ainsi, mais d'une manière générale le N-index d'une classe d'inégalités est inférieur à son rang de Chvátal.

D'autres applications de lift-and-project sont détaillées dans [10] et dans [11].

## 6.2 Un algorithme de coupes polyédrales

Les méthodes de coupes polyédrales ont montré leur efficacité pour la résolution des problèmes d'optimisation combinatoire. Ces méthodes peuvent se ranger en deux catégories :

- Les méthodes dédiées à un problème dans lesquelles le problème de séparation est traité par un algorithme spécifique tenant compte de la structure du problème à résoudre.
- Les méthodes “généralistes” dans lesquelles le problème de séparation est traité par le même algorithme quelque soit le problème.

La combinaison des méthodes de coupes et de branchement permet d'accroître l'efficacité des algorithmes.

Le lecteur ne manquera pas de remarquer les similitudes qui existent entre la preuve que nous avons donné de la finitude de la méthode Gomory et la preuve de la finitude de l'algorithme que nous présentons ici. On notera la définition d'une hiérarchie de relaxations, voir 2.2 et la finitude de chacun des éléments. On remarquera encore la similitude des rôles joués par les demi-espaces  $H$  et  $H_0$  pour la méthode de Gomory et par les rayons extrêmes du cône polaire de  $N(K)$ .

Par ailleurs le lecteur pourra noter la différence qui existe entre l'algorithme de lift-and-project proposé par E. Balas, S. Ceria et G. Cornuéjols et celui que nous proposons ci-après. Tandis que E. Balas, S. Ceria et G. Cornuéjols traitent séquentiellement les variables binaires nous proposons, conformément à la construction de L. Lovász et A. Schrijver de les traiter simultanément.

### 6.2.1 Un algorithme de coupes polyédrales basé sur la construction de Lovász et Schrijver

Nous avons vu au lemme 5.1 que  $K^0 \subseteq N(K) \subseteq K$  et au théorème 5.1 que si on définit  $N^0(K) = K$  et  $N^t(K) = N(N^{t-1}(K))$  alors  $K^0 = N^n(K)$ . Ceci nous permet de définir

une hiérarchie de relaxation du problème *Plne* de la manière suivante :

$$K^0 = N^n(K) \subseteq \dots \subseteq N^{j+1}(K) \subseteq N^j(K) \subseteq \dots \subseteq N(K) \subseteq K \quad (6.4)$$

De plus le théorème 5.3 nous permet de traiter récursivement le problème de séparation sur  $N^j(K)$ . Aussi peut-on construire un algorithme de coupes polyédrales résolvant successivement chacune des relaxations.

Supposons que nous cherchions à minimiser une fonction objective  $c$  sur l'ensemble de points défini par *Plne*. Nous pouvons alors décrire un algorithme de coupes polyédrales de la manière suivante :

#### *Algorithme Magyaro-Batave*

1.  $x_0^* = \text{Argmin}\{c^t x \mid x \in K \cap \{x \mid x_0 = 1\}\}$
2.  $p = 1, N_1(K) = K$
3. Tant que  $\text{Sep}(p) = \text{Min}\{w_p^t x_{p-1}^* \mid w_p \in N(N^p(K))^* \cap S\} < 0$ 
  - (a)  $\bar{w}_p = \text{Argmin}\{w_p^t x_{p-1}^* \mid w_p \in N(N_p(K))^* \cap S\}$
  - (b)  $N_p(K) = N_p(K) \cap \{x \mid \bar{w}_p^t x \geq 0\}$
  - (c)  $x_{p-1}^* = \text{Argmin}\{c^t x \mid x \in N_p(K) \cap \{x \mid x_0 = 1\}\}$
  - (d)  $p = p + 1$

On prendra soin de ne pas confondre les cônes notés respectivement  $N^j(K)$  et  $N_p(K)$ . Ce dernier cône est redéfini à chaque étape de l'algorithme par ajout d'une contrainte violée. Etant donnée la manière dont sont générées ces contraintes il est clair que pour toute valeur de  $j$  il existe une valeur de  $p$  telle que  $x_p^* \in N_p(K)$  implique  $x_p^* \in N^j(K)$ . Remarquons encore qu'à l'instar de l'algorithme de coupes polyédrales de Balas, Ceria et Cornuéjols, de toute réalisation de la méthode Gomory-Chvátal, l'algorithme décrit ci-dessus ne génère pas l'enveloppe convexe des points à coordonnées 0-1 du problème, il en génère seulement le bon côté: savoir celui qui fait que la fonction objective appartient au cône des contraintes serrées par l'optimum entier du programme. Notons encore que  $S$  est un ensemble de normalisation dont l'emploi a pour but de nous assurer que les programmes linéaires liés aux problèmes de séparation ont des optima finis, l'utilisation de ce type d'ensembles sera discutée à la section 6.2.2.

Afin de prouver la finitude de l'algorithme *Magyaro-Batave* nous allons établir les deux lemmes suivants. Le premier sur la finitude de  $N(K)$  et le deuxième qui précisera la condition de sortie de la boucle Tant que.

**Lemme 6.3** *Si  $K$  est un cône polyédral alors  $N(K)$  est généré de manière finie.*

Preuve :

Etant donné que  $N(K) = M(K)e_0$  on peut dire que  $N(K)$  est polyédral si  $M(K)$  l'est. Nous avons vu au théorème 5.2 que :

$$M(K)^* = U_1 + U_{asym} + \text{cône}\{uv^t, u \in K^*, v \in Q^*\}$$

Il est clair que tant  $U_1$  que  $U_{asym}$  sont générés de manière finie. Par hypothèse  $K$  est polyédral et par construction  $Q$  est aussi polyédral, donc  $K^*$  et  $Q^*$  sont des cônes polyédraux. Nous savons que  $cône\{uv^t, u \in K^*, v \in Q^*\}$  a pour rayons extrêmes les matrices de la forme  $uv^t$ ,  $u$  décrivant l'ensemble des rayons extrêmes de  $K^*$  et  $v$  décrivant l'ensemble des rayons extrêmes de  $Q^*$ . Ces deux ensembles de rayons extrêmes étant finis on peut dire que  $cône\{uv^t, u \in K^*, v \in Q^*\}$  est un cône polyédral. Il s'ensuit que  $M(K)^*$  est polyédral et par conséquent  $N(K)$  l'est aussi. ■

**Lemme 6.4** *Soit  $C$  un cône et  $x$  un point extrême de  $C \cap \{x | x_0 = 1\}$ . Si  $x \notin \{0, 1\}^n$  alors  $x \notin N(C)$ .*

Preuve :

Par le lemme 5.3 de Lovász and Schrijver nous savons que pour toute coordonnée  $j$ ,  $N(C) \subseteq conv(C_0^j \cup C_1^j)$  où  $C_i^j = \{x \in C | x_j = i\}$ . Donc si  $x \notin \{0, 1\}^n$  on peut trouver une coordonnée  $j$  telle que  $0 < x_j < 1$ . De plus  $x$  étant un point extrême il ne peut pas s'exprimer comme combinaison linéaire convexe de vecteurs de  $C_0^j$  et de  $C_1^j$ , aussi peut-on affirmer  $x \notin conv(C_0^j \cup C_1^j)$  et par conséquent  $x \notin N(C)$ . ■

**Théorème 6.3** *Si  $K$  est un cône polyédral alors l'algorithme Magyaro-Batave résout le problème  $Min\{c^t x | x \in K \cap \{x | x_0 = 1\}, x \in \{0, 1\}^n\}$  en un nombre fini d'opérations.*

Preuve :

L'initialisation de l'algorithme et chaque étape de la boucle Tant que consistent en des programmes linéaires qui peuvent être résolu en un nombre fini d'opérations, ce nombre est même polynomial en fonction de la dimension de  $K$ . Pour montrer la finitude de l'algorithme il suffit de montrer que la boucle Tant que est parcourue un nombre fini de fois ou d'une manière équivalente que  $p$  a une valeur finie lorsque l'algorithme s'arrête.

Supposons que pour une valeur  $p_{j-1}$  de  $p$  l'algorithme ait déterminé un point  $x_{j-1}^* \in N^{j-1}(K)$  mais tel que  $x_{j-1}^* \notin N^j(K)$ . Il s'agit alors de montrer que partant de  $x_{j-1}^*$  on peut trouver un point  $x_j^* \in N^j(K)$  en un nombre fini d'étapes. A chaque fois que la boucle est parcourue l'algorithme génère un vecteur  $\bar{w}_p \in N(N_p(K))^*$  autrement dit tel que  $\bar{w}_p \in N(N^{j-1}(K))^* = N^j(K)^*$ ,  $\bar{w}_p$  étant un rayon extrême de  $N^j(K)^*$ . Les rayons extrêmes de  $N^j(K)^*$  correspondent aux faces de  $N^j(K)$ , on peut donc dire qu'à chaque fois que la boucle est parcourue une face de  $N^j(K)$  est générée. Les cônes  $N^j(K)$  et  $N^j(K)^*$  étant tous deux polyédraux on peut donc affirmer que le point  $x_j^* \in N^j(K)$  sera atteint en un nombre fini d'étapes : au plus le nombre de faces de  $N^j(K)$ .

Par le lemme 6.4 on peut dire que si  $x_{p-1}^* \notin \{0, 1\}^n$  alors  $x_{p-1}^* \notin N(N_p(K))$ , aussi l'algorithme continue jusqu'à ce que  $x_{p-1}^* \in \{0, 1\}^n$ , on a alors  $x_{p-1}^* \in N^n(K)$  et par conséquent  $Min\{\bar{w}_p^t x_{p-1}^* | \bar{w}_p \in N(N^n(K))^* \cap S\} \geq 0$ . Ce qui prouve que l'algorithme se termine en fournissant un point de  $K \cap \{x | x_0 = 1\} \cap \{0, 1\}^n$ . L'optimalité de ce point provient du fait que quelque soit la valeur de  $p$  le point  $x_{p-1}^*$  est optimum sur  $N_p(K)$ . ■

### Obtention de coupes pour l'algorithme Magyaro-Batave

Nous allons voir ici comment peut être résolu le problème de séparation en tenant compte du fait que nous ne disposons pas d'une description de  $N^{j-1}(K)^*$  puisque nous ne générons pas l'ensemble des contraintes décrivant  $N^{j-1}(K)$  mais seulement celles qui définissent le point optimum sur ce polyèdre.

Par soucis de simplicité de l'écriture nous notons  $Sep(j)$  le problème de séparation sur le cône  $N^j(K)$ . Ainsi sur  $N(K)$  a-t-on :

$$Sep(1) = \left\{ \begin{array}{l} Min \quad w_1^t x_0^* \\ w_1 \in N(K)^* \end{array} \right.$$

Par le théorème 5.3 nous savons que  $w_1 \in N(K)^*$  si et seulement si il existe  $n$  vecteurs  $a_1^1, \dots, a_n^1 \in K^*$ , un réel  $\lambda^1$  et une matrice  $\bar{A} \in U_{asym}$  tels que :

$$\begin{aligned} a_i^1 + \lambda^1 e_i + \bar{A} e_i &\in K^* \\ w_1 &= \sum_{i=1}^n a_i^1 + \bar{A} \mathbf{1} \end{aligned}$$

Si  $K = \{x \in \mathbb{R}^{n+1} \mid [-b, A]x \geq 0\} \subseteq Q$  alors  $K^*$  est le cône généré par les lignes de la matrice  $[-b, A]$ . Ainsi peut-on réécrire le problème  $Sep(j)$  comme suit :

$$Sep(1) = \left\{ \begin{array}{l} Min \quad \left( \sum_{i=1}^n a_i^1 + \bar{A} \mathbf{1} \right)^t x_0^* \\ a_i^1 + \lambda^1 e_i + \bar{A} e_i = u_i [-b, A] \\ u_i \geq 0 \end{array} \right.$$

Regardons maintenant le problème de séparation sur le cône courant  $N^j(K)$ , on a :

$$Sep(j) = \left\{ \begin{array}{l} Min \quad w_j^t x_{j-1}^* \\ w_j \in N^j(K)^* \end{array} \right.$$

D'une manière similaire à ce qui a été vu à pour  $N(K)$  on peut dire que :  $w_j \in N^j(K)^*$  si et seulement si il existe  $n$  vecteurs  $a_1^j, \dots, a_n^j \in N^{j-1}(K)^*$ , un réel  $\lambda^j$  et une matrice  $\bar{A}^j \in U_{asym}$  tels que :

$$\begin{aligned} a_i^j + \lambda^j e_i + \bar{A}^j e_i &\in N^{j-1}(K)^* \\ w_j &= \sum_{i=1}^n a_i^j + \bar{A}^j \mathbf{1} \end{aligned}$$

Aussi peut-on réécrire le problème de séparation  $Sep(j)$  de la manière suivante :

$$Sep(j) = \left\{ \begin{array}{l} Min \quad \left( \sum_{i=1}^n a_i^j + \bar{A}^j \mathbf{1} \right)^t x_{j-1}^* \\ a_i^j + \lambda^j e_i + \bar{A}^j e_i \in N^{j-1}(K)^* \\ a_i^j \in N^{j-1}(K)^*, \lambda \in \mathbb{R}, \bar{A}^j \in U_{asym} \end{array} \right.$$

Remarquons que si à la première itération nous possédons une description totale par ses générateurs du cône  $K^*$ , il n'en va pas de même pour le cône courant  $N^j(K)$ . En effet nous ne possédons pas une description totale du cône  $N^{j-1}(K)^*$  car nous ne générons pas  $N^{j-1}(K)$

mais seulement une relaxation de celui-ci que nous notons  $\bar{N}^{j-1}(K)$  et dont le cône polaire est  $\bar{N}^{j-1}(K)^*$ . Etant donné que  $N^{j-1}(K) \subseteq \bar{N}^{j-1}(K)$  on a  $\bar{N}^{j-1}(K)^* \subseteq N^{j-1}(K)^*$ . Il nous faut donc établir que les vecteurs de  $\bar{N}^{j-1}(K)^*$  permettent de déterminer au moins un hyperplan séparant  $x_{j-1}^*$  de  $N^j(K)$ . C'est ce qu'établit le théorème suivant.

**Théorème 6.4** *Si  $x_{j-1}^* = \text{Argmin}\{c^t x \mid x \in N^{j-1}(K) \cap \{x \mid x_0 = 1\}\}$  alors il existe un vecteur  $w_j \in N^j(K)^*$  tel que  $w_j^t x_{j-1}^* < 0$  si et seulement si le programme :*

$$\bar{S}ep(j) = \begin{cases} \text{Min} & \left( \sum_{i=1}^n a_i^j + \bar{A}^j \mathbf{1} \right)^t x_{j-1}^* \\ & a_i^j + \lambda e_i + \bar{A}^j e_i \in \bar{N}^{j-1}(K)^* \\ & a_i^j \in \bar{N}^{j-1}(K)^*, \lambda \in \mathbb{R}, \bar{A}^j \in U_{asym} \end{cases}$$

*a une valeur optimale strictement négative.*

Preuve :

Appliquons le lemme 6.4 à  $C = \bar{N}^{j-1}(K)$  et à  $x = x_{j-1}^*$  qui est un point extrême de  $C$ . Comme  $x \notin N(C)$  nous pouvons générer un hyperplan séparateur, aussi le programme  $\bar{S}ep(j)$  a-t-il une valeur optimale strictement négative. Ce qui permet de continuer le déroulement de l'algorithme. ■

Par hypothèse  $x_{j-1}^*$  est une solution optimale du programme  $\text{Min}\{c^t x \mid x \in N^{j-1}(K)\}$ . On peut supposer sans perte de généralité que  $x_{j-1}^*$  est un sommet des polyèdres  $\bar{N}^{j-1}(K)$  et  $N^{j-1}(K)$ , avec  $N^{j-1}(K) \subseteq \bar{N}^{j-1}(K)$ . Aussi peut-on dire qu'il existe  $n$  vecteurs  $v_1, \dots, v_n$  qui sont des rayons extrêmes de  $\bar{N}^{j-1}(K)^*$  et  $N^{j-1}(K)^*$  puisqu'ils correspondent aux facettes qui définissent le point extrême  $x_{j-1}^*$ ; on a  $v_i^t x_{j-1}^* = 0, \forall i \in \{1, \dots, n\}$ . Etant donné que  $w^t x_{j-1}^* \geq 0, \forall w \in N^{j-1}(K)^*$  et que  $x_{j-1}^*$  vérifient de manière lâche toute autre contrainte que celles définies par les vecteurs  $v_1, \dots, v_n$  nous pouvons dire que toute solution optimale tant du programme  $Sep(j)$  que du programme  $\bar{S}ep(j)$  contient les vecteurs  $v_i, i \in \{1, \dots, n\}$ . Par conséquent le programme  $Sep(j)$  peut se réécrire de la manière suivante en prenant  $a_i^j = v_i$  :

$$Sep(j) = \begin{cases} \text{Min} & (\bar{A}^j \mathbf{1})^t x_{j-1}^* \\ & a_i^j + \lambda e_i + \bar{A}^j e_i \in N^{j-1}(K)^* \\ & a_i^j \in N^{j-1}(K)^*, \lambda \in \mathbb{R}, \bar{A}^j \in U_{asym} \end{cases}$$

d'une manière analogue :

$$\bar{S}ep(j) = \begin{cases} \text{Min} & (\bar{A}^j \mathbf{1})^t x_{j-1}^* \\ & a_i^j + \lambda e_i + \bar{A}^j e_i \in \bar{N}^{j-1}(K)^* \\ & a_i^j \in \bar{N}^{j-1}(K)^*, \lambda \in \mathbb{R}, \bar{A}^j \in U_{asym} \end{cases}$$

Si on note  $[\bar{b}^{j-1}, \bar{A}^{j-1}]$  la matrice des contraintes associées à  $\bar{N}^{j-1}(K)$  le programme  $\bar{S}ep(j)$  devient alors :

$$\bar{S}ep(j) = \begin{cases} \text{Min} & (\bar{A}^j \mathbf{1})^t x_{j-1}^* \\ & a_i^j + \lambda e_i + \bar{A}^j e_i = u_i [\bar{b}^{j-1}, \bar{A}^{j-1}] \\ & a_i^j \in \bar{N}^{j-1}(K)^*, \lambda \in \mathbb{R}, \bar{A}^j \in U_{asym}, u_i \geq 0 \end{cases}$$

$\bar{N}^{j-1}(K)^*$  étant un cône il se peut que le problème  $\bar{Sep}(j)$  ait un optimum non borné, aussi pour s'assurer que le problème de séparation atteint son optimum pour une valeur finie nous le résolvons sur  $\bar{N}^{j-1}(K)^* \cap S$  où  $S$  est un ensemble de normalisation ayant pour but de tronquer  $\bar{N}^{j-1}(K)^*$ . Ceci sera discuté dans la section 6.2.2.

### Renforcement des relaxations par la programmation semi-définie positive

De même que nous avons défini une hiérarchie de relaxations linéaires nous pouvons définir une hiérarchie de relaxations semi-définies positives :

$$K^0 = N^{n,+}(K) \subseteq \dots \subseteq N^{j+1,+}(K) \subseteq N^{j,+}(K) \subseteq \dots \subseteq N^+(K) \subseteq K \quad (6.5)$$

D'autre part nous avons vu lors de leur définition que les cônes de matrices  $M(K)$  et  $M^+(K)$  sont tels que  $M^+(K) \subseteq M(K)$  et donc  $N^+(K) \subseteq N(K)$ . Par ailleurs le théorème 5.3 nous fournit les conditions de séparation sur  $N^+(K)$ . En effet  $w_+ \in N^+(K)$  si et seulement si il existe  $n$  vecteurs  $a_1^1, \dots, a_n^1 \in K^*$ , un réel  $\lambda^1 \in \mathbb{R}$ , une matrice  $\bar{A} \in U_{asym}$  et une matrice  $B \in U_{psd}$  tels que :

$$\begin{aligned} a_i^1 + \lambda^1 e_i + \bar{A} e_i + B e_i &\in K^* \\ w_+ &= \sum_{i=1}^n a_i^1 + \bar{A} \mathbf{1} + B \mathbf{1} \end{aligned}$$

Ainsi on peut dire que  $x \in N^+(K)$  si et seulement si le programme :

$$Sep^+(1) = \begin{cases} Min & w_+^t x \\ & a_i^1 + \lambda^1 e_i + \bar{A} e_i + B e_i \in K^* \\ & w_+ = \sum_{i=1}^n a_i^1 + \bar{A} \mathbf{1} + B \mathbf{1} \end{cases}$$

a une valeur optimale positive ou nulle.

A l'étape courante  $j$ .  $N^{j,+}(K) \subseteq N^j(K) \subseteq \bar{N}^j(K)$ , on définit le problème de séparation  $\bar{Sep}^+(j)$  comme suit :

$$\bar{Sep}^+(j) = \begin{cases} Min & w_+^t x \\ & a_i^j + \lambda^j e_i + \bar{A}^j e_i + B^j e_i \in \bar{N}^{j-1}(K)^* \\ & w_+ = \sum_{i=1}^n a_i^j + \bar{A}^j \mathbf{1} + B^j \mathbf{1} \\ & a_i \in \bar{N}^{j-1}(K)^*, \lambda \in \mathbb{R}, \bar{A}^j \in U_{asym}, B^j \in U_{psd} \end{cases} \quad (6.6)$$

Soit  $w_+$  un vecteur de  $N^{j,+}(K)^*$  pour lequel  $\bar{Sep}^+(j) < 0$ , on définit alors le problème d'optimisation renforcé :

$$Sop = Min \{c^t x \mid x \in N^j(K) \cap \{x \mid w_+^t x \geq 0\}\} \quad (6.7)$$

L'opérateur  $N^+(K)$  n'étant pas polyédral a priori, il est illusoire d'en décrire l'enveloppe convexe par des inégalités linéaires. Aussi nous ne générerons des coupes que tant que deux instances successives de  $Sop$  présentent un accroissement de leur valeur supérieur à un certain  $\epsilon$ .

M. Goemans et L. Tunçel,[39], se sont intéressés aux conditions qui sont nécessaires, suffisantes, pour que les contraintes de semi-définie positivité apportent un réel renforcement

de l'opérateur  $N(K)$  quand on cherche à résoudre un problème d'optimisation en variables 0-1. L'aide des contraintes de semi-définie positivité peut survenir dans trois cas :

- $M^+(K) \subsetneq M(K)$  ;
- $N^+(K) \subsetneq N(K)$  ;
- $Max\{c^t x, x \in N^+(K)\} > Max\{c^t x, x \in N(K)\}$

En réalité les résultats qu'ils ont obtenus sont des résultats négatifs qui précisent quand les contraintes de semi-définie positivité sont d'aucune aide, les deux principaux sont les suivants :

**Théorème 6.5 (Goemans et Tunçel)**  $M^+(K) = M(K)$  si et seulement si  $U_{psd} \subseteq \text{cône}\{uv^t, u \in \text{Extr}(K^*), v \in \text{Extr}(Q^*)\} + U_1$ , où  $U_1$  est l'espace orthogonal à l'espace des matrice  $\{Y \mid \text{diag}(Y) = Y e_0\}$  et  $\text{Extr}(C)$  désigne l'ensemble des rayons extrêmes du cône  $C$ .

Ceci a comme corollaire que si  $U_{psd} \subseteq \text{cône}\{uv^t, u \in \text{Extr}(K^*), v \in \text{Extr}(Q^*)\} + U_1$  alors  $\forall t \in \{1, \dots, n\} N^t(K) = N^{+,t}(K)$ .

**Théorème 6.6 (Goemans et Tunçel)**  $N^+(K) = N(K)$  si et seulement si pour tout vecteur  $s \in \mathbb{R}^{n+1}$  on a :

$$\text{Diag}(s) \in \text{cône}\{uv^t, u \in \text{Extr}(K^*), v \in \text{Extr}(Q^*)\} + U_1 + U_{psd} \implies \text{Diag}(s) \in \text{cône}\{uv^t, u \in \text{Extr}(K^*), v \in \text{Extr}(Q^*)\} + U_1$$

### 6.2.2 Ensembles de normalisation

Nous avons vu précédemment que le problème de séparation  $\bar{S}ep(j)$  peut s'écrire de la manière suivante :

$$\bar{S}ep(j) = \begin{cases} \text{Min} & (\bar{A}^j \mathbf{1})^t x_{j-1}^* \\ & \lambda e_i + \bar{A}^j e_i = u_i [\tilde{b}^{j-1}, \tilde{A}^{j-1}] \\ & \lambda \in \mathbb{R}, \bar{A}^j \in U_{asym}, u_i \geq 0 \end{cases}$$

Ce programme linéaire étant défini sur un cône il peut avoir un optimum qui n'est pas fini, aussi nous cherchons à tronquer ce cône de telle sorte que le problème  $\bar{S}ep(j)$  ait un optimum fini.

$$\left. \begin{array}{l} \lambda e_i + \bar{A}^j e_i = u_i [\tilde{b}^{j-1}, \tilde{A}^{j-1}] \\ u_i \geq 0 \end{array} \right\} \iff \lambda e_i + \bar{A}^j e_i \in \bar{N}^{j-1}(K)^*$$

Nous voulons donc tronquer le cône  $\bar{N}^{j-1}(K)^*$ , une manière de faire est de restreindre  $\bar{N}^{j-1}(K)^*$  à l'ensemble des vecteurs qui sont une combinaison linéaire convexe de ses rayons extrêmes qui sont données par les vecteurs lignes de la matrice  $[\tilde{b}^{j-1}, \tilde{A}^{j-1}]$ . Dans ce cas le problème  $\bar{S}ep(j)$  se réécrit comme suit :

$$\bar{S}ep(j) = \begin{cases} \text{Min} & (\bar{A}^j \mathbf{1})^t x_{j-1}^* \\ & \lambda e_i + \bar{A}^j e_i = u_i [\tilde{b}^{j-1}, \tilde{A}^{j-1}] \\ & \|u_i\|_1 = 1 \\ & \lambda \in \mathbb{R}, \bar{A}^j \in U_{asym}, u_i \geq 0 \end{cases}$$



### 6.2.3 Utilisation dans un algorithme de type Branch-and-cut

Les algorithmes de type Branch-and-cut sont parmi les algorithmes les plus efficaces pour résoudre des problèmes de programmation linéaires en nombres entiers. Ces algorithmes combinent à la fois les techniques de coupes polyédrales et la séparation par évaluation progressive, branch-and-bound. Lors du déroulement de l'itération courante d'un algorithme de type Branch-and-cut la relaxation linéaire du problème *Plne* est enrichie de contraintes valides pour *Plne* et certaines des variables sont fixées soit à 0 soit à 1. La relaxation linéaire de *Plne* est définie de la manière suivante :

$$Plne.rl = \begin{cases} Min & c^t x \\ s.t. & x \in V_0 = K \cap \{x \in \mathbb{R}^{n+1} | x_0 = 1\} \cap \{x | 0 \leq x \leq 1\} \end{cases} \quad (6.8)$$

#### Un schéma de Branch-and-cut

Désignons par  $V$  l'ensemble courant des inégalités valides pour *Plne*, nous supposons que le système d'inégalités linéaires,  $Mx \geq 0$ , qui définit  $V$  contient au moins les inégalités de la relaxation linéaire de *Plne*. Soit  $X_0$  et  $X_1 \subseteq \{1, \dots, n\}$  les ensembles de variables respectivement fixées à 0 et à 1.

Soit  $K(V, X_0, X_1) = \{x | Mx \geq 0; x_i = 0 \ i \in X_0; x_i = 1 \ i \in X_1\}$  et  $LP(V, X_0, X_1)$  le programme linéaire  $Min\{c^t x | x \in K(V, X_0, X_1)\}$ . Nous supposons que ce programme atteint un optimum fini. Les sommets actifs de l'arbre d'énumération seront représentés dans la liste  $S$ , on note  $UB$  la valeur de la meilleure solution courante du programme *Plne*. Le schéma de Branch-and-cut peut alors s'énoncer de la manière suivante :

1.  $S := \{X_0 = \emptyset, X_1 = \emptyset\}$ ;  $V := V_0$ ;  $UB := +\infty$
2. Choisir une paire  $(X_0, X_1)$ ,  $S := S - \{X_0, X_1\}$ . Si  $S = \emptyset$  alors fin.
3. Résoudre  $LP(V, X_0, X_1)$ 
  - (a) Si  $LP(V, X_0, X_1)$  n'a pas de solution alors aller en 2.
  - (b)  $\bar{x} := ArgLP(V, X_0, X_1)$  Si  $c^t \bar{x} \geq UB$  aller en 2.  
Si  $\bar{x} \in \{0, 1\}^n$ ,  $x^* := \bar{x}$ ,  $UB := c^t \bar{x}$ , aller en 2.
4. Si générer des coupes alors aller en 5  
sinon aller en 6
5. Générer des coupes de la forme  $w^t x \geq 0$  valides pour *Plne* telles que  $w^t \bar{x} < 0$ . L'ensemble  $V$  est alors défini par le système  $[M, \alpha]^t x \geq 0$ .
6. Séparer sur les variables : choisir une valeur de l'index  $j \in \{1, \dots, n\}$  telle que  $0 < \bar{x}_j < 1$ . Générer les sous-problèmes correspondants  $LP(V, X_0 \cup \{j\}, X_1)$  et  $LP(V, X_0, X_1 \cup \{j\})$ . Calculer les minorants pour les nœuds  $(X_0 \cup \{j\}, X_1)$  et  $(X_0, X_1 \cup \{j\})$  en résolvant les deux programmes linéaires.  $S := S \cup \{X_0 \cup \{j\}, X_1\} \cup \{X_0 \cup \{j\}, X_1\}$ . Aller en 2.

Remarquons que si une coupe est générée en un nœud  $(X_0, X_1)$  elle ne demeure valide que pour les nœuds où les variables fixées conservent la même valeur. Il est donc intéressant de pouvoir rendre une coupe générée en un nœud valide en tous les autres nœuds de l'arbre de Branch-and-cut.

### Lifting des coupes générées à un nœud

Nous avons aux lemmes 5.4 et 5.5 que  $N(K \cap H_i) = N(K) \cap H_i$  et que  $N(K \cap G_i) = N(K) \cap G_i$ . Considérons un nœud de l'arbre de branc-and-cut où nous effectuons un branchement sur la variable  $x_i$ , notons  $K$  le cône défini par l'ensemble des contraintes générées avant le nœud où s'effectue le branchement sur la variable  $x_i$ . Remarquons que fixer la variable  $x_i$  à 0 est équivalent à intercepter le cône  $K$  par l'hyperplan  $H_i = \{x \in \mathbb{R}^{n+1} \mid x_i = 0\}$ .

**Lemme 6.5**  $w \in N(K \cap H_i)^* \iff w \in N(K)^*$  et  $w \in N(K \cap G_i)^* \iff w \in N(K)^*$

Preuve :

Nous avons vu que  $N(K \cap H_i) = N(K) \cap H_i$ , lemme 5.4, donc leurs cônes polaires respectifs sont liés par la relation :

$$N(K \cap H_i)^* = N(K)^* + H_i^* \quad (6.9)$$

$H_i$  est la droite d'équation  $x_j = 0, \forall j \neq i, H_i$  contient donc 0. Par la relation 6.9 on peut écrire que  $\forall u \in N(K)^*$  et  $\forall v \in H_i^*$  on a  $u + v \in N(K \cap H_i)^*$  en particulier si on prend  $v = 0$  et donc  $u \in N(K)^* \implies u \in N(K \cap H_i)^*$ . Par ailleurs l'équation 6.9 peut se réécrire  $N(K)^* = N(K \cap H_i)^* - H_i^*$ , aussi  $\forall w \in N(K \cap H_i)^*$  et  $\forall v \in H_i^*$  on  $w - v \in N(K)^*$  en particulier si  $v = 0$  on a  $w \in N(K)^*$ .

La deuxième assertion du lemme se montre en remplaçant la coordonnée  $x_i$  par la coordonnée  $1 - x_i$ . ■

Intersecter le cône  $K$  par l'hyperplan  $H_i$ , respectivement  $G_i$ , est exactement ce qui est fait en un sommet de l'arbre de Branch-and-cut lorsqu'on fixe  $x_i$  à 0, respectivement à 1. Aussi une conséquence intéressante, dans le cadre d'un algorithme de type Branch-and-cut, est que si une coupe est générée après la fixation d'une variable, à 0 ou 1, cette coupe est valide pour  $N(K)$  et donc pour  $N^n(K)$  qui est l'enveloppe convexe des points solutions de *Plne*.

Ces faits peuvent être vus comme une conséquence du théorème suivant dû à M. Goemans et L. Tunçel, [39] :

#### **Théorème 6.7 (Goemans et Tunçel)**

$N^{t,+}(K) \subseteq N(K) \subseteq \tilde{N}_0^t(K)$ , où :

$$\tilde{N}_0^t(K) = \bigcap_{\mathcal{I} \subseteq \{1, \dots, n\}, |\mathcal{I}|=t} \sum_{(\mathcal{I}_0, \mathcal{I}_1), \mathcal{I}_0 \cup \mathcal{I}_1 = \mathcal{I}, \mathcal{I}_0 \cap \mathcal{I}_1 = \emptyset} K \cap \{x \mid x_i = 0 \ i \in \mathcal{I}_0, x_i = 1 \ i \in \mathcal{I}_1\}$$

Sommairement l'opérateur  $\tilde{N}_0^t(K)$  peut être vu comme l'intersection sur l'ensemble des sous-ensembles de variables de cardinalité  $t$ , de l'ensemble des branchements possibles sur  $t$  variables. Etant donné que  $\tilde{N}_0^t(K)^* \subseteq N^{t,+}(K)^* \subseteq N^t(K)^*$  on peut dire qu'une coupe générée après  $t$  branchements quelconques est valide pour  $N^{t,+}(K)^*$  et  $N^t(K)^*$ , cette conséquence du théorème de Goemans et Tunçel peut être déduite du lemme 6.5 en posant  $K = N(K)$  à chaque fois qu'on effectue un branchement ; l'opérateur  $\tilde{N}_0^t(K)$  a toutefois d'autres propriétés géométriques que nous n'utilisons pas ici.

## Chapitre 7

# Optimisation dans un treillis

Définir un programme linéaire en nombres entiers sur un ensemble  $S$  c'est avoir la volonté de déterminer quelles sont les occurrences des éléments de cet ensemble qui doivent composer une sélection ; cette sélection devant respecter certaines contraintes, par exemple pour le remplissage d'un sac à dos ne pas prendre plus d'objets que le sac peut en contenir, et atteindre l'optimum d'une certaine fonction objective. On peut remarquer intuitivement que la réponse à ce problème de sélection consistera à dire que nous prenons  $x_1$  fois l'élément 1,  $x_2$  fois l'élément 2, etc... Ainsi peut-on dire que la réponse est une combinaison linéaire à coefficients entiers positifs des singletons de l'ensemble  $S$ . Cette combinaison linéaire peut être vue comme une fonction définie sur le treillis formé par l'ensemble des sous-ensembles de  $S$ ,  $T = 2^S$ , muni d'une loi additive notée  $\vee$  et d'une loi multiplicative notée  $\wedge$ . Ces deux lois induisent un ordre sur le treillis, ordre défini par la relation  $a \leq b \iff a = a \wedge b$ . Dans un premier temps nous allons étudier certaines propriétés de ces fonctions, lesquelles seront exprimées par des relations avec la matrice  $\mathcal{Z}$  et son inverse appelée matrice de Möbius du treillis. Ensuite nous verrons comment ces fonctions peuvent nous ramener aux cônes de matrices  $M(K)$  et  $M^+(K)$  définis au paragraphe 3.3. Enfin nous étudierons la projection du cône de ces fonctions sur le sous-espace des singletons ce qui nous permettra un retour sur les résultats exposés au paragraphe 5.1. Nous illustrons notre propos en considérant le treillis  $T = 2^S$  où  $S = \{a, b, c\}$ .

### 7.1 Fonctions fortement décroissantes

Après avoir défini les matrices  $\mathcal{Z}$  et de Möbius associées à un treillis  $T$  nous définissons ce que sont les fonctions fortement décroissantes, lesquelles permettent de décrire les différentes combinaisons d'éléments d'un ensemble. Si la sélection ne peut pas comporter plus d'une occurrence de chaque élément il est clair que la fonction recherchée est à coefficients 0 ou 1.

Considérons un treillis  $T$  et associons lui une matrice  $\mathcal{Z}$  définie par :

$$\mathcal{Z}_{ij} = \begin{cases} 1 & \text{si } i \leq j \\ 0 & \text{si non} \end{cases} \quad (7.1)$$

Si les lignes et les colonnes de la matrice  $\mathcal{Z}$  sont indexées par les éléments du treillis en respectant l'ordre induit sur le treillis, cette matrice est triangulaire supérieure et ses

termes diagonaux sont égaux à 1. Ainsi pour le treillis  $T = 2^{\{a,b,c\}}$  de notre exemple on obtient :

$$\mathcal{Z}_{\{a,b,c\}} = \begin{array}{cccccccccc} & \emptyset & \{a\} & \{b\} & \{b,a\} & \{c\} & \{c,a\} & \{c,b\} & \{c,b,a\} & \\ & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & \emptyset \\ & & 1 & & 1 & & 1 & & 1 & \{a\} \\ & & & 1 & 1 & & & 1 & 1 & \{b\} \\ \mathcal{Z}_{\{a,b,c\}} = & & & & 1 & & & & 1 & \{b,a\} \\ & & & & & 1 & 1 & 1 & 1 & \{c\} \\ & & & & & & 1 & & 1 & \{c,a\} \\ & & & & & & & 1 & 1 & \{c,b\} \\ & & & & & & & & 1 & \{c,b,a\} \end{array}$$

On peut remarquer que si  $\mathcal{Z}_n$  est la matrice associée au treillis  $T_n = 2^S$  avec  $|S| = n$ , alors la matrice  $\mathcal{Z}_{n+1}$  associée au treillis  $T_{n+1} = 2^{S \cup \{s_{n+1}\}}$  est de la forme :  $\mathcal{Z}_{n+1} = \begin{pmatrix} \mathcal{Z}_n & \mathcal{Z}_n \\ 0 & \mathcal{Z}_n \end{pmatrix}$ .

Soit  $x \in \mathbb{R}^{|T|}$  on a alors  $(\mathcal{Z}x)_i = \sum_{i \leq j} x_j$ , d'une manière similaire  $(\mathcal{Z}^t x)_i = \sum_{j \leq i} x_j$ , où  $(v)_i$  désigne la  $i^{\text{ème}}$  coordonnée du vecteur  $v$ .

Il apparaît clairement que la matrice  $\mathcal{Z}$  associée au treillis  $T$  est régulière, on appelle matrice de Möbius du treillis  $T$  la matrice  $\mathcal{M} = \mathcal{Z}^{-1}$ . La matrice de Möbius du treillis de l'exemple est :

$$\mathcal{M}_{\{a,b,c\}} = \begin{pmatrix} 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ & 1 & & -1 & & 1 & & -1 \\ & & 1 & -1 & & & 1 & -1 \\ & & & 1 & & & & 1 \\ & & & & 1 & 1 & 1 & 1 \\ & & & & & 1 & & -1 \\ & & & & & & 1 & -1 \\ & & & & & & & 1 \end{pmatrix}$$

Si  $\mathcal{M}_n$  est la matrice de Möbius du treillis  $T_n$ , alors la matrice  $\mathcal{M}_{n+1}$  du treillis  $T_{n+1}$  est de la forme :  $\mathcal{M}_{n+1} = \begin{pmatrix} \mathcal{M}_n & -\mathcal{M}_n \\ 0 & \mathcal{M}_n \end{pmatrix}$ .

Maintenant nous allons construire un cône de  $\mathbb{R}^{|T|}$  qui sera lié au treillis par les matrices  $\mathcal{Z}$  et  $\mathcal{M}$ , nous verrons comment ce cône permet de construire les différents objets manipulés aux paragraphes 3.3 et 5.1.

Soit  $x \in \mathbb{R}^{|T|}$  une fonction linéaire définie sur le treillis  $T$  et  $\mathcal{Z}$  la matrice associée à ce treillis, celle-ci étant régulière nous pouvons dire qu'il existe  $y \in \mathbb{R}^{|T|}$  tel  $x = \mathcal{Z}y$  et donc  $y = \mathcal{Z}^{-1}x = \mathcal{M}x$  ou d'une manière équivalente  $y_i = \sum_{i \leq j} \mathcal{M}_{ij}x_j$ ; la fonction  $y$  ainsi définie est appelée inverse de Möbius supérieure de la fonction  $x$ . Une fonction  $x \in \mathbb{R}^{|T|}$  est dite fortement décroissante si  $\mathcal{M}x \geq 0$ .

### **Théorème 7.1 (Lovász et Schrijver)**

*Etant donné un treillis  $T$ , l'ensemble des fonctions  $x \in \mathbb{R}^{|T|}$  fortement décroissantes forme un cône dont les rayons extrêmes sont les vecteurs colonnes,  $z_j$ , de la matrice  $\mathcal{Z}$  associée au treillis  $T$ . On note  $\mathcal{H}$  ce cône.*

Preuve :

Considérons  $x_1$  et  $x_2 \in \mathbb{R}^{|T|}$  deux fonctions fortement décroissantes sur le treillis  $T$ , on a alors :

$$\left. \begin{array}{l} \mathcal{M}x_1 \geq 0, \mathcal{M}x_2 \geq 0 \\ \mathcal{M}x_1 + \mathcal{M}x_2 = \mathcal{M}(x_1 + x_2) \end{array} \right\} \implies \mathcal{M}(x_1 + x_2) \geq 0 \implies x_1 + x_2 \in \mathcal{H}$$

Par ailleurs  $\forall \lambda \in \mathbb{R}^+$  et  $\forall x \in \mathbb{R}^{|T|}$  on a  $\mathcal{M}(\lambda x) = \lambda(\mathcal{M}x) \geq 0 \implies \lambda x \in \mathcal{H}$ . Aussi peut-on conclure que  $\mathcal{H}$  est un cône convexe.

Par définition si  $x \in \mathcal{H}$  on a  $\mathcal{M}x \geq 0$  et par définition de la matrice de Möbius on a  $x = \mathcal{Z}(\mathcal{M}x)$ , ce qui montre que  $x$  peut s'écrire comme combinaison linéaire à coefficients positifs des colonnes de la matrice  $\mathcal{Z}$ . ■

### Corollaire 7.1 (Lovász et Schrijver)

Le cône polaire du cône  $\mathcal{H}$  est généré par les vecteurs lignes de la matrice de Möbius du treillis,  $\mathcal{H}^* = \text{cône}\{m_i, i \in \{1, \dots, |T|\}\}$ .

Preuve :

On a vu que par définition  $x \in \mathcal{H} \iff \mathcal{M}x \geq 0$ . La nécessité de la condition montre que  $m_i \in \mathcal{H}^*$ ,  $\forall i \in \{1, \dots, |T|\}$ , tandis que la suffisance de cette condition montre que toute combinaison linéaire à coefficients non négatifs des vecteurs lignes de la matrice  $\mathcal{M}$  appartient à  $\mathcal{H}^*$ . Ainsi peut-on conclure que les vecteurs lignes de la matrice  $\mathcal{M}$  sont les rayons extrêmes de  $\mathcal{H}^*$ . ■

Pour toute fonction linéaire  $x \in \mathbb{R}^{|T|}$  on définit la matrice  $W^x$  par :  $W_{ij}^x = x(i \vee j)$ . On peut remarquer que si  $y = \mathcal{M}x$  alors  $W^x = \mathcal{Z} \text{Diag}(y) \mathcal{Z}^t$ .

### Lemme 7.1 (Lovász et Schrijver)

$x \in \mathbb{R}^{|T|}$  est une fonction fortement décroissante si et seulement si  $W^x$  est une matrice semi-définie positive.

Preuve :

1. Nécessité de la condition.

Par hypothèse  $x$  est une fonction fortement décroissante donc  $y = \mathcal{M}x \geq 0$  et on a vu que  $W^x$  peut s'écrire  $\mathcal{Z} \text{Diag}(y) \mathcal{Z}^t$ , ainsi peut-on conclure à la semi-définie positivité de la matrice  $W^x$ .

2. Suffisance de la condition.

Se montre par des arguments similaires en remarquant que si  $x$  n'est pas fortement décroissante il existe une composante  $y_i$  de  $y$  qui est strictement négative, ce qui induit un terme diagonal strictement négatif de  $W^x$ . ■

Illustrons notre propos en contruisant la matrice  $W^x$  pour  $x = (8, 7, 6, 5, 4, 3, 2, 1)$  sur le treillis  $2^{\{a,b,c\}}$ , on obtient alors :

$$W^x = \begin{pmatrix} 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 \\ 7 & 7 & 5 & 5 & 3 & 3 & 1 & 1 \\ 6 & 5 & 6 & 5 & 2 & 1 & 2 & 1 \\ 5 & 5 & 5 & 5 & 1 & 1 & 1 & 1 \\ 4 & 3 & 2 & 1 & 4 & 3 & 2 & 1 \\ 3 & 3 & 1 & 1 & 3 & 3 & 1 & 1 \\ 2 & 1 & 2 & 1 & 2 & 1 & 2 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

On peut remarquer que la première ligne est égale à la première colonne et à la diagonale, ceci indépendamment de  $x$ . En effet quel que soit  $x$  on a  $x(\emptyset \vee i) = x(i \vee \emptyset) = x(i \vee i) = x(i)$  car  $\emptyset \vee i = i \vee \emptyset = i \vee i = i$ .

## 7.2 Projection sur l'espace des singletons et du sous-ensemble vide

Nous allons voir dans ce paragraphe comment on peut générer l'enveloppe convexe des vecteurs caractéristiques d'un sous-ensemble d'éléments d'un treillis en projetant un cône inclus dans  $\mathcal{H}$  sur l'espace généré par les singletons, ce qui permettra de montrer un résultat dû à Sherali et Adams,[81]. Ensuite nous montrerons comment construire les cônes  $M(K)$  et  $M^+(K)$  à partir des matrices  $W^x$  définies ci-dessus.

Soit  $F$  un sous-ensemble du treillis  $T$ , on note  $\text{cône}(F)$  le cône généré par les vecteurs colonnes de la matrice  $\mathcal{Z}$ ,  $z_j, j \in F$ . Les vecteurs  $z_j, j \in T$  étant des rayons extrêmes de  $\mathcal{H}$  et étant linéairement indépendants on peut écrire que :

$$\text{cône}(F) = \{x \in \mathbb{R}^{|T|} \mid m_i^t x = 0 \text{ si } i \notin F, m_i^t x \geq 0 \text{ si } i \in F\} \quad (7.2)$$

Notons  $I$  l'ensemble des éléments du treillis  $T$  correspondant aux singletons et à l'ensemble vide.  $\text{Proj}_I(K)$  désigne la projection du cône  $K$  sur l'espace engendré par les éléments de  $I$ ,  $\mathbb{R}^{|I|}$ . Si  $F \subset T$  et  $\chi_i$  désigne le vecteur caractéristique de l'élément  $i \in T$  on peut énoncer le théorème suivant :

### **Théorème 7.2 (Lovász et Schrijver)**

$$\text{conv}(\{\chi_i, i \in F\}) = \text{Proj}_I(\text{cône}(F)) \cap \{x \mid x_\emptyset = 1\}$$

Preuve:

Par définition :

$$\text{cône}(F) = \text{cône}(\{z_j, j \in F\})$$

aussi peut-on écrire :

$$\begin{aligned} \text{Proj}_I(\text{cône}(F)) &= \text{cône}(\{\text{Proj}_I(z_j), j \in F\}) \\ &= \{x \in \mathbb{R}^{|I|} \mid x = \sum_{j \in F} \lambda_j \text{Proj}_I(z_j), \lambda_j \geq 0 \forall j\} \end{aligned}$$

En remarquant que  $Proj_I(z_j) = (1, \chi_j^t)^t$  on peut écrire que :

$$Proj_I(\text{cône}(F)) = \{x \in \mathbb{R}^{|I|} \mid x = \sum_{j \in F} \lambda_j (1, \chi_j^t)^t, \lambda_j \geq 0 \forall j\}$$

Soit en intersectant  $Proj_I(\text{cône}(F))$  avec l'hyperplan d'équation  $x_\emptyset = 1$  :

$$\begin{aligned} Proj_I(\text{cône}(F)) \cap \{x \mid x_\emptyset = 1\} &= \{x \in \mathbb{R}^{|I|} \cap \{x \mid x_\emptyset = 1\} \mid x = \sum_{j \in F} \lambda_j \chi_j, \lambda_j \geq 0, \sum_{j \in F} \lambda_j = 1\} \\ &= \text{conv}(\{\chi_i, i \in F\}) \end{aligned}$$

Le théorème suivant dû à Sherali et Adams peut alors être vu comme une conséquence directe du théorème précédent et de la description de  $\text{cône}(F)$  par le système d'inégalités 7.2. ■

**Théorème 7.3 (Sherali et Adams)**

Si  $F \subseteq 2^S$  alors  $\text{conv}(\{\chi_i, i \in F\})$  est la projection sur l'espace engendré par les singletons du cône défini par les contraintes :

$$x_\emptyset = 1, m_i^t x = 0 \text{ si } i \notin F, m_i^t x \geq 0 \text{ si } i \in F$$

Reprenons notre exemple sur le treillis  $2^{\{a,b,c\}}$  en considérant  $F = \{\{c, a\}, \{c, b\}\}$ . Nous pouvons voir que la projection de  $F$  sur le sous-espace des singletons et de l'ensemble vide est l'ensemble des vecteurs caractéristiques des éléments de  $F$ .

	$\emptyset$	$\{a\}$	$\{b\}$	$\{b, a\}$	$\{c\}$	$\{c, a\}$	$\{c, b\}$	$\{c, b, a\}$	
	1	1	1	1	1	<b>1</b>	<b>1</b>	1	$\emptyset$
	0	1	0	1	0	<b>1</b>	<b>0</b>	1	$\{a\}$
	0	0	1	1	0	<b>0</b>	<b>1</b>	1	$\{b\}$
$\mathcal{Z}_{\{a,b,c\}} =$	0	0	0	1	0	0	0	1	$\{b, a\}$
	0	0	0	0	1	<b>1</b>	<b>1</b>	1	$\{c\}$
	0	0	0	0	0	1	0	1	$\{c, a\}$
	0	0	0	0	0	0	1	1	$\{c, b\}$
	0	0	0	0	0	0	0	1	$\{c, b, a\}$

Remarquons que  $Proj_I(\mathcal{H}) = Q$ ,  $Q$  étant le cône engendré par les vecteurs à coordonnées 0-1 avec  $x_\emptyset = 1$ .

Considérons une fonction  $x \in \mathcal{H}$ , nous avons vu que la matrice  $W^x$  est semi-définie positive et de plus vérifie  $W^x e_\emptyset = \text{diag}(W^x)$ . Maintenant définissons le produit de vecteurs  $u \vee v$  par :

$$(u \vee v)_k = \sum_{i,j \mid i \vee j = k} u_i v_j$$

On peut alors remarquer que  $e_i \vee e_j = e_{i \vee j}$  et que  $m_i \vee m_i = m_i$ . La définition du produit ci-dessus nous permet de dire que :

$$\begin{aligned} (u \vee v)^t x &= \sum_k x_k \sum_{i,j \mid i \vee j = k} u_i v_j \\ &= \sum_i u_i \sum_j x_{i \vee j} v_j \\ &= u^t W^x v \end{aligned}$$

Ce qui permet d'écrire que  $(u \vee u)^t x = u^t W^x u \geq 0 \iff x$  est fortement décroissante. Par ailleurs dire que  $x \in \text{cône}(F)$  c'est dire que :

$$m_i^t x = (m_i \vee m_i)^t x = m_i^t W^x m_i \begin{cases} = 0 & \text{si } i \notin F \\ \geq 0 & \text{si } i \in F \end{cases} \quad (7.3)$$

Par ailleurs on peut établir que :  $m_i \vee m_j = 0$  si  $i \neq j$  et donc :

$$(m_i \vee m_j)^t x = m_i^t W^x m_j = 0, \text{ si } i \neq j \quad (7.4)$$

Ces deux dernières relations nous permettent alors d'écrire que :

$$\begin{aligned} (m_i \vee m_j)^t x &\iff m_i^t W^x m_j \geq 0, \forall m_i \in \text{cône}(F)^*, \forall m_j \in \mathcal{H}^* \\ &\iff W^x \mathcal{H}^* \subseteq \text{cône}(F) \end{aligned} \quad (7.5)$$

Si nous considérons une matrice  $Y$  égale au mineur de  $W^x$  induit par l'ensemble vide et les singletons nous obtenons une matrice carrée d'ordre  $n + 1$  qui vérifie  $Y = Y^t$  et  $Y e_0 = \text{diag}(Y)$ . La condition  $W^x \mathcal{H}^* \subseteq \text{cône}(F)$  fournit la condition  $Y Q^* \subseteq K$  vue à la définition 3.2, tandis que  $Y \succeq 0$  est une relaxation de  $W^x \succeq 0 \iff x \in \mathcal{H}$ .

### 7.3 Totale unimodularité de la matrice $\mathcal{Z}$

Nous allons voir maintenant une caractéristique importante de la matrice  $\mathcal{Z}$  d'un treillis de la forme  $2^S$ .

**Théorème 7.4** *Soit  $T = 2^S$  un treillis et  $\mathcal{Z}$  l'inverse de la matrice de Möbius de ce treillis, alors la matrice  $\mathcal{Z}$  est totalement unimodulaire.*

Preuve :

On peut aisément vérifier que la matrice  $\mathcal{Z}_1 = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$  est totalement unimodulaire. De

même la matrice  $\mathcal{Z}_2 = \begin{pmatrix} \mathcal{Z}_1 & \mathcal{Z}_1 \\ 0 & \mathcal{Z}_1 \end{pmatrix}$ .

Soit  $\mathcal{Z}_n$  la matrice associée au treillis des partitions d'un ensemble à  $n$  éléments. Considérons le système d'équations linéaires  $\mathcal{Z}_n x = b$ ,  $b$  étant un vecteur dont les coordonnées sont des entiers. On peut alors réécrire les quatres dernières lignes de ce système de la manière suivante :

$$(0, \mathcal{Z}_2) x = (b_{2^n-3}, b_{2^n-2}, b_{2^n-1}, b_{2^n})^t$$

La totale unimodularité de la matrice  $\mathcal{Z}_2$  et l'intégrité du vecteur  $b$  nous permettent de conclure à l'intégrité des composantes  $x_i$ ,  $i \in \{2^n - 3, 2^n - 2, 2^n - 1, 2^n\}$ . Maintenant considérons les huit dernières lignes du système d'équations linéaires  $\mathcal{Z}_n x = b$ . Ces lignes forment le système suivant :

$$\begin{pmatrix} 0 & \mathcal{Z}_2 & \mathcal{Z}_2 \\ 0 & 0 & \mathcal{Z}_2 \end{pmatrix} x = (b_{2^n-7}, \dots, b_{2^n})^t$$

Il apparait clairement que les solutions de ce système d'équations linéaires sont entières. Supposons que nous ayons établi que les  $2^{n-1}$  dernières lignes de la matrice  $\mathcal{Z}_n$  forment une matrice totalement unimodulaire. Le système d'équations peut se réécrire sous la forme suivante :

$$\begin{pmatrix} Z_{n-1} & Z_{n-1} \\ 0 & Z_{n-1} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$



aussi peut-on écrire que  $Z_{n-1}x_1 = b_1 - Z_{n-1}x_2$  est un vecteur à coordonnées entières. Le caractère totalement unimodulaire de la matrice  $Z_n$  est donc acquis. ■

## 7.4 Bases de Hilbert des cônes $M(K)$ et $M^+(K)$

S'il est clair que les cônes  $N(K)$  et  $N^+(K)$  possèdent la même base de Hilbert, ils ont le même ensemble de point à coordonnées entières, la question de savoir si  $M(K)$  et  $M^+(K)$  partagent une telle base admet une réponse beaucoup moins directe. C'est à cette question que nous répondons maintenant.

**Définition 7.1** *On appelle base de Hilbert d'un polyèdre  $P$ , respectivement base de Hilbert entière, un ensemble  $H = \{h_1, \dots, h_m\}$  de vecteurs de  $P$ , respectivement à coordonnées entières, tels que tout vecteur  $y$  de  $P$  à coordonnées entières peut s'écrire  $y = H\lambda$  où  $H$  est la matrice formée par les vecteurs de  $H$  et  $\lambda$  est un vecteur dont les composantes sont des entiers naturels.*

Nous allons dans un premier temps montrer que tout polyèdre admet une base de Hilbert puis nous verrons comment calculer un tel ensemble de vecteurs en utilisant les propriétés du cône  $M(K)$ .

**Lemme 7.2 (Gordan)** *Soit  $\mathcal{I} \neq \emptyset$  un ensemble de vecteurs de dimension  $n$  dont les coordonnées sont des entiers naturels, alors il existe un sous-ensemble minimal unique de  $\mathcal{I}$ ,  $\bar{\mathcal{I}} = \{i_1, \dots, i_m\}$ , tel que  $\forall i \in \mathcal{I}$  il existe  $k \in \{1, \dots, m\}$  tel que  $i \geq i_k$ .*

Preuve :

La preuve se fait par induction sur la dimension des vecteurs de  $\mathcal{I}$ . Le cas  $n = 1$  est trivial, supposons donc le lemme vrai pour des vecteurs de dimension  $n - 1$ . Considérons un ensemble de vecteurs de dimension  $n - 1$  :  $\mathcal{I}' = \{x = (x_1, \dots, x_{n-1})\}$  tels qu'il existe un entier naturel  $y$  tel que  $(x_1, \dots, x_{n-1}, y) \in \mathcal{I}$ . Par l'hypothèse d'induction il existe un ensemble  $\{i_1, \dots, i_m\} \subseteq \mathcal{I}'$  tel que  $\forall x \in \mathcal{I}'$  il existe  $k \in \{1, \dots, m\}$  tel que  $x \geq i_k$ . De plus par construction de  $\mathcal{I}'$  on peut dire que  $\forall k \in \{1, \dots, m\}$  il existe un entier naturel  $y$  tel que  $(i_k, y) \in \mathcal{I}$ . Pour tout indice  $k \in \{1, \dots, m\}$  on note  $\bar{y}_k = \text{Min}\{y \mid (i_k, y) \in \mathcal{I}\}$ . De plus on définit  $\mathcal{I}'_k = \{(x_1, \dots, x_{n-1}, k) \in \mathcal{I}\}$ ; alors par l'hypothèse d'induction on sait qu'il existe  $m$  vecteurs  $u_k^1, \dots, u_k^m$  tels que pour tout  $x_k \in \mathcal{I}'_k$  il existe  $j \in \{1, \dots, m\}$  tel que  $x_k \geq u_k^j$ . Alors il apparaît clairement que l'ensemble :

$$\eta = \bigcup_{k=1}^m \{(u_k^1(1), \dots, u_k^m(n-1), k)\} \cup \{(i_1, \bar{y}_1)\} \cup \dots \cup \{(i_m, \bar{y}_m)\}$$

est tel que  $\forall i \in \mathcal{I}$ ,  $\exists e \in \eta$  tel que  $i \geq e$ . De plus la définition ci-dessus de l'ensemble  $\eta$  montre sa finitude.

Considérons maintenant un ensemble  $S = \{x \in \mathcal{I} \mid \nexists i \in \mathcal{I} ; i \leq x\}$ . Clairement on a  $S \subseteq \eta$ , en effet par définition un élément  $x$  de  $S$  est minimal pour la relation d'ordre sur  $\mathcal{I}$  et donc s'il n'est pas dans  $\eta$  on peut dire qu'il existe  $x \in \mathcal{I}$  tel qu'il n'existe pas de vecteur  $e \in \eta$  vérifiant  $e \leq x$ , ce qui conduit à une contradiction. Réciproquement supposons qu'il existe  $e \in \eta$  tel que  $e \notin S$ ,  $e$  ne peut pas être de la forme  $(i_k, \bar{y}_k)$  par définition même de  $\bar{y}_k$ . On peut donc dire que sous cette hypothèse  $e$  est de la forme  $(u_k^1(1), \dots, u_k^m(n-1), k)$ ,

or les vecteurs  $u_k^j$  ont été choisis de sorte qu'ils minorent les vecteurs de l'hyperplan d'équation  $x - n = k$ , donc si  $e = (u_k^1(1), \dots, u_k^m(n-1), k) \notin S$  on a une contradiction. Pour finir remarquons que la définition même de  $S$  conduit à la minimalité de  $\eta$ . ■

Afin de poursuivre notre investigation des cônes  $M(K)$  et  $M^+(K)$  nous allons étudier un résultat dû à Gordan et van der Corput, [44], [84]

**Théorème 7.5 (Gordan, van der Corput)** *Tout cône polyédral rationnel admet une base de Hilbert entière, si ce cône est un cône pointé il existe une unique base de Hilbert entière.*

Preuve :

Considérons un cône polyédral  $K$  généré par l'ensemble de vecteurs  $\{k_1, \dots, k_n\}$ ; pour effectuer la preuve du théorème ci-dessus on peut supposer sans perte de généralité que ces  $n$  vecteurs sont des vecteurs à coordonnées entières. Considérons maintenant l'ensemble  $\bar{P} = \{p_1, \dots, p_q\}$  des vecteurs à coordonnées entières du polytope :

$$P = \{x \in \mathbb{R}^n \mid x = \sum_{i=1}^n \lambda_i k_i, 0 \leq \lambda_i \leq 1\}$$

étant donnée que  $\{k_1, \dots, k_n\} \subseteq \bar{P}$  et que  $P \subseteq K$  on peut d'ores et déjà affirmer que  $\{p_1, \dots, p_q\}$  est un ensemble générateur de  $K$ . Soit  $i$  un vecteur à coordonnées entières de  $K = \text{cône}(\{k_1, \dots, k_n\})$ , alors on peut dire qu'il existe  $\mu_1, \dots, \mu_n$  tels que :

$$i = \sum_{j=1}^n \mu_j k_j = \sum_{j=1}^n [\mu_j] k_j + \sum_{j=1}^n (\mu_j - [\mu_j]) k_j \quad (7.6)$$

Partant de là on peut écrire que :

$$\sum_{j=1}^n (\mu_j - [\mu_j]) k_j = i - \sum_{j=1}^n [\mu_j] k_j$$

et donc que les deux membres de l'expression ci-dessus sont des entières. Comme nous l'avons déjà vu  $\{k_1, \dots, k_n\} \subseteq \{p_1, \dots, p_q\}$ , de plus 7.6 montre qu'on peut écrire  $i$  comme combinaison linéaire à coefficients dans  $\mathbb{N}$  des vecteurs  $\{k_1, \dots, k_n\}$ , aussi pouvons conclure que  $\{p_1, \dots, p_q\}$  forme une base de Hilbert de  $K$ .

Maintenant considérons un cône pointé,  $K$ , que nous supposons contenu dans l'orthant positif, cette hypothèse est faite sans perte de généralité car si le cône  $K$  est contenu dans plusieurs orthants on peut le considérer comme la somme de plusieurs cônes, chacun d'eux contenu dans un unique orthant, puis via une rotation considérer qu'il sont contenus dans l'orthant positif; la base de Hilbert de chacun se déduit par rotation de la base de Hilbert calculée dans l'orthant positif. Donc considérons un ensemble de points de  $K$ ,  $G = \{g_1, \dots, g_p\}$  qui satisfait les hypothèses du lemme de Gordan. Montrons que tout vecteur de  $K$  à composantes dans  $\mathbb{N}$  peut s'écrire comme combinaison linéaire à coefficients entières positifs des vecteurs de  $G$ . Par le point précédent du théorème nous savons que tout cône admet une base de Hilbert entière, soit  $H = \{h_1, \dots, h_m\}$  une telle base du cône  $K$ . Pour tout vecteur  $v$  de  $K$  dont les coordonnées appartiennent à  $\mathbb{N}$  on a alors :

$$v = \sum_{j=1}^m \alpha_j h_j, \alpha_j \in \mathbb{N} \forall j \in \{1, \dots, m\}$$

Etant donné que  $G$  satisfait les hypothèses du lemme de Gordan on peut affirmer que  $G \subseteq H$ . Supposons que  $H - G \neq \emptyset$  alors il existe  $j \in \{1, \dots, m\}$  tel qu'il existe  $h \in H$  vérifiant  $g_j < h$ . Soit  $d_{j_0} = h - g_{j_0}$ , on peut sûrement trouver un vecteur  $d_{j_0}$  qui appartient au cône  $K$  puisque  $h$  peut s'écrire comme somme de deux vecteurs de  $K$ . Si  $d_{j_0} \in G$  le cas est trivial et la preuve terminée. Supposons donc que  $d_{j_0} \notin G$ , on construit alors la suite  $d_{j_t} = d_{j_{t-1}} - g_{j_t}$  où  $g_{j_t} < d_{j_{t-1}}$  jusqu'à un indice  $t^*$  tel que  $d_{j_{t^*}} = d_{j_{t^*-1}} - g_{j_{t^*}} \in G$ , ce qui ne peut pas manquer de se produire puisque par hypothèse nous sommes dans l'orthant positif et que  $G$  est l'ensemble des minorants des vecteurs à coordonnées entières du cône  $K$ . Il vient alors que  $d_{j_{t^*-1}}$  est la somme de deux éléments de  $G$  et plus généralement que  $d_{j_t}$ ,  $t \in \{0, \dots, t^*\}$  est la somme d'éléments de  $G$ , avec de possibles répétitions, aussi peut-on affirmer que  $h$  peut s'écrire comme combinaison linéaire à coefficients dans  $\mathbb{N}$  d'éléments de  $G$ . Ceci prouve que  $G$  est une base de Hilbert pour  $K$ ; son unicité provient du fait que  $G$  satisfait le lemme de Gordan. ■

Une preuve légèrement différente de ce théorème, notamment en ce qui concerne le deuxième point du théorème peut être lue dans [79] page 233. La dernière partie de la preuve ci-dessus nous permet d'énoncer le corollaire suivant.

**Corollaire 7.2** *Si  $K$  est un cône polyédral pointé inclus dans  $\mathbb{R}^{n,+}$  alors il existe un sous-ensemble minimal unique des vecteurs à coordonnées entières  $\{g_1, \dots, g_m\}$  tel que  $\forall v \in K \cap \mathbb{N}^n$  il existe  $j \in \{1, \dots, m\}$  tel que  $g_j \leq v$ ; de plus  $\{g_1, \dots, g_m\}$  est l'unique base de Hilbert de  $K$ .*

Les résultats précédents permettent d'énoncer le lemme suivant concernant les cônes  $M(K)$  et  $M^+(K)$ .

**Lemme 7.3** *Soit  $K \subseteq Q$  un cône polyédral de  $\mathbb{R}^{n+1}$ . Les cônes  $M(K)$  et  $M^+(K)$  sont des cônes pointés. Ils possèdent la même et unique base de Hilbert entière.*

Preuve :

Par construction de  $M(K)$  si  $Y \in M(K)$  alors  $u^t Y v \geq 0 \forall u \in K^*$  et  $\forall v \in Q^*$  ainsi nous pouvons dire que  $Y \in M(K) \implies -Y \notin M(K)$ , donc le seul espace vectoriel contenu dans  $M(K)$  est celui de la matrice nulle et par définition  $M(K)$  est un cône pointé. Par ailleurs le fait que  $M^+(K) \subseteq M(K)$  permet de dire que  $M^+(K)$  est aussi un cône pointé. Le théorème 7.5 permet alors d'affirmer que tous deux possèdent une unique base de Hilbert entière.

Pour achever la preuve du lemme il faut montrer que si  $H$  est une base de Hilbert entière de  $M(K)$  alors  $H$  est aussi une base de Hilbert de  $M^+(K)$  et réciproquement. Supposons que  $H$  et  $H^+$  soient les bases de Hilbert entières respectivement de  $M(K)$  et  $M^+(K)$ . Nous avons vu, lemme 5.2, que si  $Y \in M(K)$  toute colonne de  $Y$  est un vecteur de  $K$ ; étant donné que  $K \subseteq Q$  nous pouvons dire que tous les coefficients de  $Y$  sont positifs ou nuls. Nous savons aussi par le théorème 5.1 que  $\text{conv}(K \cap \{0, 1\}^n) = N^n(K) = N^{+,n}(K)$ , étant donné que  $N(K)$  et  $N^+(K)$  comportent les mêmes vecteurs à coordonnées 0-1 que  $\text{conv}(K \cap \{0, 1\}^n)$  nous pouvons dire qu'ils ont une base de Hilbert entière commune. De plus par construction  $N(K) = M(K)e_0$  et  $N^+(K) = M^+(K)e_0$ , aussi peut-on dire que les matrices de  $H$  et  $H^+$  ont comme diagonale un vecteur à coordonnées 0-1 pris dans la base de Hilbert commune à  $N(K)$  et  $N^+(K)$ . Si  $Y \in H^+$  la diagonale dominance des matrices semi-définies positives impose que  $Y$  est une matrice à coefficients 0-1.

Supposons qu'il existe  $Y^+ \in H^+$  telle que  $Y^+ \notin H$ .  $M(K)$  et  $M^+(K)$  étant des cônes

pointés  $H$  et  $H^+$  satisfont le lemme de Gordan, par conséquent il existe  $Y \in H$  telle que pour au moins un couple d'indice  $(i, j)$   $y_{ij} = y_{ji} < y_{ij}^+ = y_{ji}^+$ . Comme  $Y^+$  a tous ses coefficients en 0-1 on peut dire que  $y_{ij} = y_{ji} = 0$ . La matrice  $Y$  ainsi obtenue depuis  $Y^+$  préserve la diagonale dominante, elle demeure donc semi-définie positive. On a alors une contradiction car  $Y < Y^+$  et  $Y \succeq 0$  font que  $H^+$  ne satisfait pas le lemme de Gordan. Supposons maintenant qu'il existe  $Y \in H$  telle que  $Y \notin H^+$ . Si  $Y \succeq 0$  alors  $H^+$  ne satisfait pas le lemme de Gordan, d'où une contradiction. Donc supposons que  $Y \notin U_{psd}$ ,  $Y$  étant symétrique et ayant tous ses coefficients positifs ou nuls on peut dire qu'elle ne vérifie pas la diagonale dominante, ce qui est contraire aux propriétés des matrices de  $M(K)$  vu au lemme 3.2

■

## Chapitre 8

# Cônes de matrices et cônes de métriques

Pour conclure nous traçons quelques perspectives pour nos futurs travaux de recherche. Pour ce qui suit nous appuyons notre réflexion sur [23] et [37].

Dans ce chapitre nous montrons quelles relations existent entre les cônes de matrices que nous avons manipulés jusqu'à présent et certains cônes de métriques, en particulier nous montrons que le cône  $M(K)$  est inclus dans un cône isomorphe au cône des métriques de coupes,  $Cut_n$ , lui-même isomorphe au cône des métriques de type  $l_1$ , voir lemme 8.1. Comme application nous modélisons le problème du graphe partiel  $k$ -arête-connexe sur le cône  $Cut_n$ , puis par plongement dans un cône de métrique approprié, celui des métriques de type  $l_2$ , nous définissons une relaxation semi-définie positive de ce problème.

Nous avons vu précédemment que les matrices semi-définies positives peuvent s'écrire comme matrices de Gram. Chacun peut constater que la dimension des vecteurs de la factorisation de Gram est directement reliée au rang de la matrice qu'on veut ainsi factoriser. Cette factorisation peut être vue comme un plongement d'espace métrique dans un autre espace métrique. Si les vecteurs de Gram ne sont pas de dimension suffisante le plongement s'effectuera en introduisant une distorsion sur la distance séparant deux points, on trouvera un exemple relatif au multiflot dans [59]. Toutefois il peut être intéressant d'effectuer un tel plongement car on peut obtenir un algorithme se déroulant plus rapidement sans que l'altération de la qualité des solutions soit trop importante.

Si la dimension des vecteurs de Gram est un facteur de performance, le type de métrique,  $l_p$ ,  $1 \leq p \leq \infty$ , intervient aussi. Si les métriques de type  $l_1$  semblent les plus propices pour les problèmes d'optimisation en variables binaires il se trouve que le problème de séparation sur le cône formé par ces métriques, et donc celui d'optimisation, est  $\mathcal{NP}$ -difficile. Tandis que les normes de type  $l_2$  forment un cône sur lequel ces deux problèmes peuvent être résolus en temps polynomial.

On peut aisément pressentir que lift-and-project, tout comme la méthode Gomory, réalise un plongement d'un polytope dans l'hypercube. Ce plongement consiste à déterminer quels sont les vecteurs de l'hypercube  $H_n$  qui sont les points extrêmes de l'enveloppe convexe des points à coordonnées 0-1 du polytope. Plus particulièrement nous allons nous intéresser ici aux conditions de plongement des cônes de matrices  $M(K)$  et  $M^+(K)$  dans l'hypercube. Au vu de ce qui précède il est clair qu'un tel plongement existe étant donné que les projections de la  $n^{ième}$  fermeture de ces deux cônes,  $N^n(K)$  et  $N^{+,n}(K)$ , consistent en

l'enveloppe convexe de points à coordonnées 0-1 d'un polytope  $K \cap \{x \mid x_0 = 1\}$ .  
 Considérons  $V_n$  un ensemble de  $n$  éléments et  $V_{n+1} = V_n \cup \emptyset$ . Soit  $S \subseteq V_n$  on définit la matrice  $Y(S)$  dont les lignes et les colonnes sont indexées par les éléments de  $V_{n+1}$  par son terme générique :  $Y(S)_{ij} = 1$  si  $i$  et  $j$  appartiennent à  $S$  et 0 sinon. On peut aisément voir que les matrices de cette forme ont les propriétés suivantes :

$$\begin{array}{ll}
 Y(S)_{00} = 1 & \text{en supposant que } \emptyset \subseteq \emptyset. \\
 Y(S)_{0i} = Y(S)_{ii} & \text{car } \forall S \emptyset \subseteq S \text{ et si } i \in S \text{ alors } i \text{ et } \emptyset \in S. \\
 Y(S)_{ij} \leq Y(S)_{ii} & \text{car si } i \notin S \text{ alors } Y(S)_{ij} = 0 \forall j. \\
 Y(S)_{ij} \geq Y(S)_{ii} + Y(S)_{jj} - Y(S)_{00} & \text{car } Y(S)_{ij} = Y(S)_{ii} = Y(S)_{jj} \text{ si } i \text{ et } j \in S. \\
 Y(S)_{ij} = Y(S)_{ji} & \text{trivialement.}
 \end{array}$$

De telles matrices sont appelées matrices de corrélation. Notons  $Cor_n$  le cône défini par :

$$Cor_n = \{Y \mid Y = \sum_{S \subseteq V_n} \lambda_S Y(S), \lambda_S \geq 0 \forall S \subseteq V_n\} \quad (8.1)$$

**Lemme 8.1**  $Y \in M(K) \implies Y \in Cor_n$

Preuve :

Immédiate depuis la définition de  $M(K)$ , le lemme 3.2 et les propriétés des matrices de  $Cor_n$  vues ci-dessus. ■

Définissons une semi-métrique  $\delta(S)$  par  $\delta(S)_{ij} = 1$  si  $|S \cap \{ij\}| = 1$  et 0 sinon. On note  $Cut_n$  le cône défini par :

$$Cut_n = \{\delta \mid \delta = \sum_{S \subseteq V_n} \lambda_S \delta(S), \lambda_S \geq 0 \forall S \subseteq V_n\} \quad (8.2)$$

Considérons l'application  $\xi$  de  $Cor_n$  dans  $Cut_n$  définie par :

$$\begin{array}{l}
 \xi : Cor_n \longrightarrow Cut_n \\
 Y \longrightarrow d = \begin{cases} d_{i0} = Y_{ii} \\ d_{ij} = Y_{ii} + Y_{jj} - 2Y_{ij} \end{cases}
 \end{array}$$

Il est montré dans [23] que l'application  $\xi$  est une bijection de  $Cor_n$  dans  $Cut_n$ . Aussi pouvons-nous dire que toute matrice  $Y \in M(K)$  correspond d'une manière unique à une semi-métrique de coupe  $\delta \in Cut_n$ .

**Lemme 8.2** Soit  $V_n$  un ensemble de  $n$  éléments et  $E_n = \{\{ij\}, i, j \in V_n, i \neq j\}$ . Soit  $d \in \mathbb{R}^{|E_n|}$  et  $(V_n, d)$  l'espace métrique associé. Alors les deux propositions suivantes sont équivalentes.

1.  $d = \sum_{S \subseteq V_n} \lambda_S \delta(S)$ .
2.  $(V_n, d)$  est isométriquement plongeable dans l'hypercube, c'est-à-dire qu'il existe  $n$  vecteurs  $u_1, \dots, u_n \in \{0, 1\}^n$  tels que  $\|u_i - u_j\|_1 = d_{ij}$ .

Preuve :

1.  $1 \implies 2$

Considérons une semi-métrique  $d \in \text{Cut}_n$ , alors on peut dire qu'il existe  $m$  sous-ensembles  $S_1, \dots, S_m \in V_m$  tels que  $d = \sum_{k=1}^m \delta(S_k)$ . Maintenant considérons  $m$  vecteurs de dimension  $n$  :  $v_1, \dots, v_m$  tels que  $v_k$  est le vecteur caractéristique du sous-ensemble  $S_k$ . Soit  $u_i$  le vecteur tel que sa  $k^{\text{ième}}$  composante vaut 1 si la  $i^{\text{ième}}$  composante de  $v_k$  vaut 1 et 0 sinon. On voit que  $u_i$  représente les sous-ensembles dans lesquels  $i$  est présent ; on peut donc dire que  $\|u_i - u_j\|_1$  vaut le nombre de sous-ensembles dans lesquels  $i$  ou  $j$  sont présents séparément. Aussi peut-on écrire que  $\|u_i - u_j\|_1 = \sum_{k=1}^m \delta(S_k)_{ij}$ . On a donc réalisé un plongement de  $(V_n, d)$  dans l'hypercube muni d'une métrique  $l_1$ .

2.  $2 \implies 1$

La réciproque se montre avec des arguments semblables. En effet considérons  $n$  vecteurs de l'hypercube de dimension  $m$ ,  $u_1, \dots, u_n$ . Soit  $U$  la matrice dont les lignes sont les vecteurs  $u_i$ ,  $i \in \{1, \dots, n\}$ , et considérons  $m$  sous-ensembles de  $V_n$  dont les vecteurs caractéristiques sont les colonnes de  $U$ , alors avec les mêmes arguments que précédemment on peut dire que  $\|u_i - u_j\|_1 = \sum_{k=1}^m \delta(S_k)_{ij}$ . Ce qui montre que la métrique définie sur l'hypercube appartient à  $\text{Cut}_n$ . ■

**Corollaire 8.1** *Soit  $X$  un ensemble de  $n$  vecteurs et  $Y \in M(K)$ , alors  $(X, Y)$  est isométriquement plongeable dans l'hypercube.*

Nous allons voir que le problème de séparation sur les métriques de type  $l_1$  est  $\mathcal{NP}$ -complet, ceci fut démontré par D. Avis et M. Deza dans [3].

**Théorème 8.1 (Avis et Deza)** *Le problème de séparation sur le cône  $\text{Cut}_n$  est  $\mathcal{NP}$ -complet.*

Preuve :

Soit  $G = (V, E)$  un graphe non orienté, alors  $\text{Max-cut}(G) = \text{Max} \left\{ \sum_{(x,y) \in E} \delta(S)_{xy}, \delta(S) \in \text{Cut}_n \right\}$ . Ce problème est connu pour être  $\mathcal{NP}$ -difficile, aussi par le théorème  $\text{SEP} = \text{OPT}$  on peut conclure que le problème de séparation sur  $\text{Cut}_n$  est  $\mathcal{NP}$ -complet. ■

Lorsque nous avons regardé les problèmes d'optimisation sur les treillis nous avons vu, implicitement, qu'on peut toujours réaliser un plongement dans  $2^S$  dont la dimension est  $2^n$ . Toutefois un plongement est toujours possible dans un espace de plus petite dimension.

**Théorème 8.2** *Soit  $(V_n, d)$  un espace métrique. Si  $(V_n, d)$  est isométriquement plongeable dans  $l_1^p$ , l'espace métrique de dimension  $p$  muni d'une norme  $l_1$ , alors  $C_n^2$  majore  $p$ .*

Preuve :

Soit  $d$  une métrique sur  $V_n$  qui soit  $l_1$  plongeable. Alors  $d \in \text{Cut}_n$  et donc :

$$d_{ij} = \sum_{S \subseteq V_n} \lambda_S \delta(S)_{ij} = \sum_{S | \delta(S)_{ij}=1} \lambda_S, \forall i, \forall j \in V_n$$

Ce système possède  $C_n^2$  contraintes et  $2^n$  variables, aussi on ne peut pas trouver une famille de vecteurs indépendants comportant plus de  $C_n^2$  vecteurs, d'où le théorème. ■

**Remarque :**

$C_n^2 = \frac{n(n-1)}{2}$  est une dimension polynomiale, il peut être intéressant pour certains problèmes  $\mathcal{NP}$ -difficile de se poser la question de savoir où intervient le saut de complexité. Si certes  $C_n^2$  reste polynomial, il croît avec le carré de  $|V_n|$ . Aussi la dimension de l'espace peut devenir beaucoup trop grande pour résoudre des cas pratiques. On peut alors fixer la dimension de l'espace de plongement. Ce dernier s'effectuera alors avec une certaine distorsion. Si  $\phi$  désigne le plongement, on aura :

$$d_{ij} \leq d(\phi(i), \phi(j)) \leq \alpha d_{ij}, \alpha \geq 0$$

Il nous semble intéressant d'étudier dans quelle mesure le plongement dans un espace de dimension réduit peut donner un  $\alpha$  approximation de la solution optimale d'un problème. Par exemple pour le problème du stable de poids maximum peut-on trouver une fonction  $\alpha(p)$  qui nous donne un  $\alpha(p)$  approximation du stable si les labels orthogonaux, qui sont les vecteurs de Gram d'une matrice semi-définie positive, sont pris dans un espace de dimension  $p$ ?

Si il semble illusoire de vouloir déterminer quel est le meilleur plongement dans un espace de type  $l_1^p$  car nous savons que le problème de séparation sur le cône formé par ce type de métrique est  $\mathcal{NP}$ -complet. Par contre rechercher le meilleur plongement dans un espace de type  $l_2^p$  peut se faire polynomialement.

**Théorème 8.3** Soit  $(V_n, d)$  un espace métrique. Alors  $d$  est  $l_2$  plongeable si et seulement si la matrice  $P = [p_{ij}] = [\frac{1}{2}(d_{i0}^2 + d_{j0}^2 - 2d_{ij})]$  est semi-définie positive.

Preuve :

1. Nécessité de la condition.

Supposons qu'il existe  $n$  vecteurs  $v_1, \dots, v_n \in \mathbb{R}^p$  tels que :

$$d_{ij}^2 = (v_i - v_j)^t (v_i - v_j) = v_i^t v_i + v_j^t v_j - 2v_i^t v_j$$

On peut alors dire que :

$$p_{ij} = \frac{1}{2}(v_i - v_0)^t (v_i - v_0) + (v_j - v_0)^t (v_j - v_0) - 2(v_i - v_0)^t (v_j - v_0) = (v_i - v_0)^t (v_j - v_0)$$

Ce qui montre que  $P$  est semi-définie positive.

2. Suffisance de la condition.

Si  $P$  est semi-définie positive alors il existe  $n$  vecteurs  $v_1, \dots, v_n \in \mathbb{R}^p$  tels que  $p_{ij} = v_i^t v_j$ . On peut alors écrire que  $(v_i - v_j)^t (v_i - v_j) = p_{ii} + p_{jj} - 2p_{ij} = d_{ij}^2$ . Ainsi les vecteurs  $v_1, \dots, v_n$  donne un plongement de  $(V_n, d)$ . ■



On peut remarquer que la matrice  $P$  est l'image de  $d$  par l'application covariance  $\xi$  vue précédemment.

Le problème du meilleur  $l_2$  plongement se formule alors comme suit :

$$\left\{ \begin{array}{l} \text{Min } \lambda \\ d_{ij} \leq p_{ij} \leq \lambda d_{ij} \\ [\frac{1}{2}(d_{i0}^2 + d_{j0}^2 - 2d_{ji})] = [p_{ij}] \succeq 0 \end{array} \right.$$

C'est-à-dire comme un programme semi-défini positif. La factorisation en matrice de Gram donne le plongement. Le plongement d'un espace  $(V_n, d)$  où  $d \in Cut_n$  dans un espace  $l_2^m$  s'apparente à une relaxation d'un programme en variables binaires. La relaxation semi-définie positive est obtenue par plongement dans un autre cône de métrique : celui des métriques de type négatif noté  $Neg_n$ .

**Définition 8.1** Une métrique  $d$  est de type négatif si  $\sqrt{d}$  est  $l_2$  plongeable.

**Remarque :**

On peut aisément établir que  $d$  est de type négatif si et seulement si la matrice  $P$  de terme générique  $p_{ij} = d_{i0} + d_{j0} - d_{ij}$  est semi-définie positive, le cône des métriques de type négatif est noté  $Neg_n$ .

Pour revenir à notre préoccupation initiale qui consiste à résoudre un problème en variables binaires il faut réaliser un plongement de l'espace métrique figurant la relaxation dans l'hypercube. Ce qui est équivalent à réaliser un plongement dans un espace muni de la norme  $l_1$ . Le théorème de J. Bourgain précise qu'il existe un espace muni de la norme  $l_1$  dans lequel on peut réaliser un plongement dont la distorsion est endéans un facteur  $\log_2 n$ , [12].

**Théorème 8.4 (Bourgain)** Soit  $(V_n, d)$  un espace métrique. Alors il existe  $n$  vecteurs  $v_1, \dots, v_n \in \mathbb{R}^N$  tels que :

$$\|v_i - v_j\|_1 \leq d_{ij} \leq \mathcal{O}(\log_2 n) \|v_i - v_j\|_1$$

$$\text{où } N = \sum_{k=1}^{\log_2 n} C_k^n.$$

On peut remarquer que la dimension de l'espace est a priori très grande et que le théorème ne garantit qu'un facteur  $\log_2 n$ , ce qui peut s'avérer en pratique rédhibitoire. Toutefois on peut espérer des conditions de plongement beaucoup plus favorables pour des problèmes particuliers. Il nous intéresse maintenant de rechercher de petits espaces pour la factorisation des matrices semi-définies positives dont nous nous servons pour résoudre les problèmes en variables 0-1. Il serait particulièrement intéressant de pouvoir factoriser les matrices de  $M^+(K)$  et de déterminer des plongements a priori de ses matrices à coefficients 0-1. La caractérisation des matrices de la base de Hilbert de ce cône et leur détermination algorithmique pourrait être utile.

## 8.1 Un exemple d'utilisation des cônes $Cut_n$ et $Neg_n$

Pour illustrer le propos sur l'utilisation des cônes de métriques pour résoudre des problèmes d'optimisation combinatoire nous modélisons le problème du graphe partiel k-arête-connexe sur le cône  $Cut_n$  et nous plongeons ce modèle dans le cône  $Neg_n$ .

Considérons un graphe  $G = (V, E)$  et définissons une semi-métrique  $\gamma(S)$  de la manière suivante :

$$\gamma(S)_{ij} = \begin{cases} \sum_{(uv) \in E} \delta(S)_{uv} \text{ if } |\{ij\} \cap S| = 1 \\ 0 \text{ sinon} \end{cases}$$

**Lemme 8.3**  $\gamma(S) \in Cut_n$

Preuve :

Immédiate en remarquant que  $\gamma(S)$  n'est rien d'autre que  $\delta(S)$  multipliée par le cocycle de  $S$ . ■

**Lemme 8.4** On définit  $\lambda(S) = \gamma(S) - \delta(S)$ . Alors  $\forall S \subseteq V \lambda(S) \in Cut_n$  si et seulement si  $G$  est connexe.

Preuve ::

Supposons que  $G$  ne soit pas connexe alors il existe  $S$  tel que  $\gamma(S) = 0$  et deux sommets  $i$  et  $j$  tels que  $\delta(S)_{ij} = 1$  ainsi  $\lambda(S)_{ij} = -1$ .

Réciproquement si  $G$  est connexe alors la cardinalité de chaque cocycle est au moins égale à 1. ■

Remarquons que le lemme précédent n'est rien d'autre que le critère de coupes.

**Lemme 8.5** Soit  $\lambda_k(S) = \gamma(S) - k\delta(S)$ . Alors  $\forall S \subseteq V \lambda_k(S) \in Cut_n$  si et seulement si  $G$   $k$ -arête-connexe.

Preuve :

En utilisant les mêmes arguments que pour le lemme 8.4. ■

IL est connu que  $Cut_n \subseteq Neg_n$ , ce résultat peut notamment être vu dans [23] et [37]. Aussi nous définissons l'application  $\xi$  de la manière suivante :

$$\begin{aligned} \xi : Cut_n &\longrightarrow Neg_n \\ \delta(S) &\longrightarrow Q(S) = \frac{1}{2}[\delta(S)_{ii_0} + \delta(S)_{jj_0} - \delta(S)_{ij}] \end{aligned}$$

**Lemme 8.6** L'application  $\xi$  est linéaire, si on se restreint à des scalaires positifs ou nuls  $Cut_n$  et  $Neg_n$  étant des cônes, et injective.

Preuve :

La linéarité de  $\xi$  est immédiate.

Supposons que  $Q(S) = 0$ . Etant donné que  $S$  est un sous-ensemble propre de  $V$  on peut dire qu'il existe  $\{i, j\} \subseteq V$  tel que  $\delta(S)_{ii} = 1$  donc soit  $i$  soit  $j$  est du même côté que  $i_0$ . Supposons que  $\delta(S)_{jj_0} = 1$  alors  $Q(S)_{jj} = \delta(S)_{jj_0} + \delta(S)_{jj_0} - \delta(S)_{jj} = 2\delta(S)_{jj_0} = 2$ , ce qui contredit  $Q(S) = 0$ . On peut donc dire que  $Ker\xi = \{0\}$  et donc qu'elle est injective. ■

Nous allons voir certaines propriétés de la matrice  $Q(S)$ .

**Lemme 8.7** Soit  $G = (V, E)$  un graphe  $k$ -arête-connexe  $S \subseteq V$ . Soit  $Q(S) = [q_{ij}] = \frac{1}{2}[\lambda_k(S)_{ii_0} + \lambda_k(S)_{jj_0} - \lambda_k(S)_{ij}]$ . If  $q_{ij} \neq 0$  alors  $i$  et  $j$  appartiennent tous deux à  $V - S$ , de plus  $q_{ij} = \lambda_k(S)_{ii_0}$ .

Preuve :

Supposons que  $i$  et  $i_0$  appartiennent tous deux à  $S$  et considérons le terme diagonal  $q_{ii}$  de la matrice  $Q(S)$ , on a alors :

$$q_{ii} = \frac{1}{2}(\lambda_k(S)_{ii_0} + \lambda_k(S)_{i_0i} - \lambda_k(S)_{ii}) = 0$$

car par définition de  $\lambda_k(S)$  chacun des trois termes est nul. La semi-définie positivité de la matrice  $Q(S)$  entraîne  $q_{ij} \leq q_{ii} = 0$ , par la diagonale dominance des matrices semi-définies positives. De plus  $\lambda_k(S)$  est une semi-métrie et à ce titre elle vérifie l'inégalité triangulaire :

$$q_{ij} = \frac{1}{2}(\lambda_k(S)_{ii_0} + \lambda_k(S)_{j_0i} - \lambda_k(S)_{ij}) \geq 0$$

Ainsi peut-on conclure que si  $i \in S$  alors  $\forall j \ q_{ij} = 0$ . Maintenant supposons que  $i_0 \in S$  et que  $i \notin S$ , on suppose de plus que  $j \notin S$  car dans le cas contraire on est à nouveau dans les hypothèses du cas précédent. On a alors :

$$q_{ij} = \frac{1}{2}(\lambda_k(S)_{ii_0} + \lambda_k(S)_{j_0i} - \lambda_k(S)_{ij}) = \frac{1}{2}(\lambda_k(S)_{ii_0} + \lambda_k(S)_{j_0i})$$

car  $i$  et  $j$  sont du même côté et donc  $\lambda_k(S)_{ij} = 0$ , alors par définition de  $\lambda_k(S)$  il vient que  $q_{ij} = \lambda_k(S)_{ii_0}$ . ■

On peut aisément remarquer que  $\text{rang}(Q(S)) = 1$ . On note  $Q(S, S')$  la matrice  $Q(S) + Q(S')$  et remarquons que par la linéarité de  $\xi$  on a  $\xi(\lambda_k(S) + \lambda_k(S')) = Q(S, S')$ .

**Lemme 8.8** *Soit  $S$  et  $S'$  deux sous-ensembles propres de  $V$  tels que  $S \neq S'$ , alors  $\text{rang}(Q(S, S')) = \text{rang}(Q(S)) + \text{rang}(Q(S')) = 2$ .*

Preuve :

Immédiate si on remarque que les mineurs non nuls de  $Q(S)$  et  $Q(S')$  n'ont pas les mêmes ensembles d'indices. ■

**Lemme 8.9** *Soit  $S_1, \dots, S_p$   $p$  sous-ensembles propres de  $V$  chacun d'eux contenant  $i_0$ . Alors  $\text{rang}(Q(S_1, \dots, S_p)) \leq n - 1$ .*

Preuve :

$\forall S \subseteq V$  tel que  $i_0 \in S$  on a  $Q(S)_{ii_0} = Q(S)_{i_0i} = 0$ , aussi peut-on affirmer que  $Q(S_1, \dots, S_p)$  a au moins une ligne nulle. ■

Nous allons voir comment on peut choisir  $n - 1$  sous-ensembles de sorte à obtenir une matrice de rang  $n - 1$ .

**Lemme 8.10** *Si  $G = (V, E)$  est un graphe  $k$ -arête-connexe il existe  $n - 1$  sous-ensembles de sommets  $S_1^*, \dots, S_{n-1}^*$  tels que  $Q(S_1^*, \dots, S_{n-1}^*) \succeq 0$  et  $\text{rang}(Q(S_1^*, \dots, S_{n-1}^*)) = n - 1$  si et seulement si  $\forall S \subseteq V \ Q(S) \succeq 0$*

Preuve :

Considérons  $n - 1$  sous-ensembles  $S_1^*, \dots, S_{n-1}^*$  tels que  $V - S_i^* \ i \in \{1, \dots, n - 1\}$  sont les supports de coupes d'un arbre de Gomory-Hu de  $G$ .

Suffisance de la condition

trivialement si  $\forall S \subseteq V \ Q(S) \succeq 0$  on a  $Q(S_i^*) \succeq 0 \ i \in \{1, \dots, n - 1\}$ . Etant donné  $U_{psd}$  est un cône  $Q(S_1^*, \dots, S_{n-1}^*) \succeq 0$ . Le fait que  $\text{rang}(Q(S_1^*, \dots, S_{n-1}^*)) = n - 1$  provient du fait que  $V - S_i^* \ i \in \{1, \dots, n - 1\}$  sont laminaires et du lemme 8.8.

Nécessité de la condition

Supposons qu'il existe  $S \subseteq V$  tel que  $Q(S) \notin U_{psd}$  alors  $\lambda_k(S) \notin Cut_n$  et par définition de  $S_1^*, \dots, S_{n-1}^*$  on peut dire qu'il existe  $i_0 \in \{1, \dots, n-1\}$  tel que  $\gamma(S_{i_0}^*) \leq \gamma(S)$ . Ce qui implique qu'il existe un couple de sommets  $(u, v)$  pour lesquels une coupe minimum  $\omega(V - S_{i_0}^*)$  et pour ce couple de sommets on a  $\lambda_k(S_{i_0}^*)_{uv} < 0$  ainsi  $Q(S_{i_0}^*) \notin U_{psd}$ . Ceci provient du fait que les coefficients non nuls de  $Q(S_{i_0}^*)$  sont ceux du mineur indexé par  $V - S_{i_0}$  et leur valeur est  $\gamma(S_{i_0}^*) - k\delta(S_{i_0}^*) < 0$ , aussi un des termes diagonaux de  $Q(S_{i_0}^*)$  est strictement négatif.

Soit  $Q(S_1^*, \dots, S_{n-1}^*)_{i_0}$  la matrice  $\sum_{i=1, i \neq i_0}^{n-1} Q(S_i^*)$ . Le fait que  $rang(Q(S_1^*, \dots, S_{n-1}^*)) = n - 1$  et le fait que  $rang(Q(S_{i_0}^*) = 1)$  impliquent que  $rang(Q(S_1^*, \dots, S_{n-1}^*)_{i_0}) \geq n - 2$  étant donné que  $Q(S_1^*, \dots, S_{n-1}^*) = Q(S_1^*, \dots, S_{n-1}^*)_{i_0} + Q(S_{i_0}^*)$ . De plus  $Q(S_1^*, \dots, S_{n-1}^*)_{i_0}$  est la somme de  $n - 2$  matrices de rang 1, aussi

$$rang(Q(S_1^*, \dots, S_{n-1}^*)_{i_0}) \leq n - 2.$$

Par conséquent  $rang(Q(S_1^*, \dots, S_{n-1}^*)_{i_0}) = n - 2$ . Aussi peut-on considérer que la matrice  $Q(S_1^*, \dots, S_{n-1}^*)_{i_0}$  a deux lignes identiquement nulle. L'indice d'une de ces deux lignes appartient à l'ensemble d'indices lignes du mineurs non nul de  $Q(S_{i_0}^*)$ , ce qui implique que  $Q(S_1^*, \dots, S_{n-1}^*)$  a un terme diagonal strictement négatif, aussi  $Q(S_1^*, \dots, S_{n-1}^*) \notin U_{psd}$ . ■

Pour obtenir une matrice représentative d'un graphe k-arête-connexe il nous faut encore rajouter de l'information relativement au sommet  $i_0$ . Etant donné que si  $G$  est un graphe k-arête-connexe d'ordre  $n - 1$  auquel on rajoute un sommet qu'on relie à  $k$  sommets distincts on obtient un graphe d'ordre  $n$  k-arête-connexe nous allons ajouter à la matrice  $Q(S_1^*, \dots, S_{n-1}^*)$  la matrice  $Q(S_n^*)$  où  $S_n^* = V - \{i_0\}$  obtenant ainsi la matrice  $Q(S_1^*, \dots, S_n^*)$ . On définit la matrice  $\Gamma(S)$  par  $\Gamma(S) = [\gamma(S)_{ii_0} + \gamma(S)_{ji_0} - \gamma(S)_{ij}]$  et d'une manière analogue la matrice  $\Delta(S) = [\delta(S)_{ii_0} + \delta(S)_{ji_0} - \delta(S)_{ij}]$ , on peut aisément voir que  $\Gamma(S) \succeq 0$  et  $\Delta(S) \succeq 0$ . De même la relation  $\Gamma(S) = Q(S) + k\Delta(S)$  peut être immédiatement établie depuis la définition des semi-métriques  $\delta(S)$ ,  $\gamma(S)$  et  $\lambda_k(S)$ . On définit les matrices  $\Gamma(S_1^*, \dots, S_n^*)$  et  $\Delta(S_1^*, \dots, S_n^*)$  d'une manière analogue à  $Q(S_1^*, \dots, S_n^*)$  :

$$\Gamma(S_1^*, \dots, S_n^*) = \sum_{i=1}^n \Gamma(S_i^*)$$

et

$$\Delta(S_1^*, \dots, S_n^*) = \sum_{i=1}^n \Delta(S_i^*)$$

Il apparaît clairement que les matrices  $\Gamma(S_1^*, \dots, S_n^*)$  et  $\Delta(S_1^*, \dots, S_n^*)$  sont semi-définies positives, elles peuvent donc se décomposer comme matrice de Gram. On peut alors définir une relaxation semi-définie positive du problème du graphe partiel k-arête-connexe de la

manière suivante,  $w_{ij}$  étant le coût d'une arête  $(ij) \in E$  :

$$\begin{aligned}
 \text{Min} \quad & \frac{1}{2} \sum_{(ij) \in E} w_{ij} v_i^t v_j \\
 & v_i^t v_j = 0 \quad \forall (ij) \notin E \\
 & [v_i^t v_j] = \Gamma(S_1^*, \dots, S_n^*) \succeq k \Delta(S_1^*, \dots, S_n^*) \\
 & \mathbf{0} \leq v_i \leq \mathbf{1}
 \end{aligned}$$

Avant de conclure remarquons que la matrice  $Q(S_1^*, \dots, S_n^*)$  définit des sous-graphes de niveau, chacun étant un mineur de la matrice. Il nous semble possible qu'il existe une relation entre ces sous-graphes de niveau et ceux définis par A. Sebő dans [80] pour les paires conservatives. La vérification de ce fait permettrait une généralisation de son théorème de structure sur les multiflots planaires car nous ne requerrons pas cette dernière hypothèse, cela nous demandera encore du travail.

## Chapitre 9

# Conclusion

La programmation mathématique en variables binaires est un domaine central de l'optimisation combinatoire dans ce sens qu'elle permet d'aborder tous les problèmes de cette discipline. Au cours de ce mémoire nous avons abordé une grande partie des résultats importants relatifs à la programmation linéaire en variables bivalentes, toutefois il nous manque d'avoir abordé l'important travail de M. Conforti, G. Cornuéjols et A. Kapoor sur les matrices idéales, parfaites et équilibrées.

Nous avons pu voir que les approches qui consistent à décrire l'enveloppe convexe des points à coordonnées entières d'un domaine polyédral ont toutes des fondements communs, aussi nous ne sommes pas surpris que leurs efficacités soient comparables. Ces approches peuvent être qualifiées de duales et nous les opposerons à des approches primales qui sont illustrées par l'algorithme de réduction de base de Lovász. Nous pensons que ces approches primales qui concourent à déterminer une base de Hilbert, une base de Gröbner, un test set méritent d'être mieux prise en compte dans la recherche de méthode efficace pour la programmation linéaire en nombres entiers ; les travaux de R. Wiesmantel en attestent.

Par ailleurs la programmation semi-définie positive a permis une amélioration des méthodes duales en définissant des relaxations plus fortes que la simple relaxation continue. de plus elle a permis de résoudre exactement certains problèmes d'optimisation combinatoire comme le problème du stable dans les graphes parfaits. La programmation semi-définie positive se couple naturellement avec la définition de péri-isométrie entre espaces métriques, elle permet alors l'élaboration d'algorithmes d'approximation. nous n'avons abordé ce domaine qu'au travers de l'exposé des travaux de M. Goemans et D. Williamson et par une esquisse pour le problème du graphe partiel  $k$ -arête-connexe mais nous pensons que ces investigations méritent d'être poursuivies.

Pour conclure disons deux mots des travaux présentés en annexe. Si nous avons pu déterminer le nombre minimum d'arêtes d'un graphe 2-connexe et de diamètre  $d$ , prouvant le caractère polynomial du problème lorsque le poids de chaque arête vaut 1, il demeure que si la pondération des arêtes est quelconque le problème est  $\mathcal{NP}$ -difficile. Aussi pensons-nous poursuivre notre travail en recherchant un algorithme d'approximation, c'est ce qui motive le travail exposé à la fin du chapitre sept.

Concernant les problèmes de flot avec multiplicateurs dépendant du chemin parcouru préalablement ils constituent un champ d'investigation à part entière. L'importance pour France Télécom, qui est notre employeur, de ces problèmes nous conduit à rechercher des méthodes de résolution plus efficaces, ceci passe par l'élaboration de techniques de recherche de chemins beaucoup plus fine que celles que nous mettons en œuvre actuellement.

# Bibliographie

- [1] K. Aardal, R. Weismantel, and L. Wolsey. Non-Standard Approaches to Integer Programming. *Preprint*, 2000.
- [2] F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, 5:13–51, 1995.
- [3] D. Avis and M. Deza. The cut cone,  $L^1$  embeddability, complexity and multicommodity flows. *Networks*, 21:595–617, 1991.
- [4] E. Balas. Disjunctive programming : Properties of the convex hull of feasible points. *MSRR Carnegie Mellon University*, 348, 1974.
- [5] E. Balas. Disjunctive programming and a hierarchy of relaxations for discrete optimisation problems. *SIAM Journal on Algebraic and Discrete Methods*, 6:466–486, 1985.
- [6] E. Balas, S. Ceria, and G. Cornuéjols. A lift-and-project cutting plane algorithm for mixed 0-1 programs. *Mathematical Programming*, 58:295–324, 1993.
- [7] E. Balas, S. Ceria, and G. Cornuéjols. Mixed 0-1 programming by lift-and-project in a branch-and-cut framework. *Management Science*, 42, number 9:1229–1246, 1996.
- [8] F. Barahona. Network design using cut inequalities. *SIAM Journal on optimization*, 6:823–837, 1996.
- [9] J.E Basley and N. Christofides. An Algorithm for the Resource Constrained Shortest Path Problem. *NETWORKS*, 19:379–394, 1989.
- [10] S. Bertrand. Cônes de matrices semi-définies positives pour la programmation en variables booléennes. *Centre National d'Études des Télécommunications, Rapport RP/CNET/5914*, 1998.
- [11] S. Bertrand and A. Laugier. L'algorithme du martin-pêcheur pour les problèmes de multifiot en variables mixtes. *ROADEF 99*, 1999.
- [12] J. Bourgain. On Lipschitz embedding of finite metric spaces in Hilbert space. *Israel Journal of Mathematics*, vol 52:46–52, 1985.
- [13] F. Boyer. Conception et routage dans les réseaux de télécommunication ; application de la méthode de Benders généralisée. *Thèse de doctorat Université Blaise Pascal*, 1997.

- 
- [14] M. Burlet and C. Lemaréchal. *Programmation Semi-définie positive et Applications à l'Optimisation Combinatoire. Tutorial*, 1999.
- [15] A. Caprara and M. Fischetti.  $0\text{-}\frac{1}{2}$  Chvátal-Gomory cuts. *Mathematical Programming*, vol 74:221–236, 1996.
- [16] C. Carathéodory. Über den Variabilitätsbereich der Fourierschen Konstanten von positive harmonischen Funktionen. *Rendiconto del Circolo Matematico di Palermo*, vol 7:193–217, 1911.
- [17] S. Ceria and J. Soares. Disjunctive cut generation for mixed 0-1 programs: duality and lifting. *submitted to Mathematical Programming*, 1997.
- [18] V. Chvátal. Edmonds polytopes and a hierarchy of combinatorial problems. *Discrete Mathematics*, 4:305–337, 1973.
- [19] V. Chvátal. *Linear Programming*. W.H. Freeman and Company, 1983.
- [20] W. Cook, J. Fonlupt, and A. Schrijver. An integer analogue of Carathéodory's theorem. *Journal of Combinatorial Theory (B)*, 40:63–70, 1986.
- [21] G.B. Dantzig. Maximization of a linear function of variables subject to linear inequalities. *Proceedings of Linear Programming Conference, June 20-24, 1949*, pages 359–373, 1951.
- [22] C de la Vallée Poussin. Sur la méthode de l'approximation minimum. *Annales de la société scientifique de Bruxelles*, 1910.
- [23] M.M. Deza and M. Laurent. *Geometry of Cuts and Metrics*. Springer, 1997.
- [24] J. Edmonds. Paths, Trees, and flowers. *Canadian Journal of Mathematics*, 17:449–467, 1965.
- [25] J. Edmonds and R. Giles. A min-max relation for submodular functions on graphs. *Annals of Discrete Mathematics*, 1:185–204, 1977.
- [26] J. Edmonds and R.M. Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the association for computing machinery*, 19:248–264, 1972.
- [27] L. Engebretsen. Approximate Constraint Satisfaction. *Doctoral Dissertation*, Kungl Tekniska Högskolan, 2000.
- [28] J. Farkas. On the applications of the mechanical principle of Fourier. *Mathematikai és természettudományi Értesítő*, 12:457–472, 1894.
- [29] J. Farkas. Paraméteres módszer Fourier mechanikai elvéhez. *Mathematikai és Fizikai Lapok*, 7:63–71, 1898.
- [30] J.B.J. Fourier. Analyse des travaux de l'Académie Royale des Sciences pendant l'année 1824. *Histoire de l'Académie Royale des Sciences de l'Institut de France*, 7, 1827.



- 
- [31] A. Frieze and M. Jerrum. Improved approximation algorithms for max  $k$ -cut and max bisection. 1994.
- [32] D.R. Fulkerson. Blocking polyhedra. *Graph theory and its applications*, pages 93–114, 1970.
- [33] D.R. Fulkerson. Blocking and antiblocking pairs of polyhedra. *Mathematical programming*, 1:168–194, 1971.
- [34] D.R. Fulkerson. Antiblocking polyhedra. *Journal of combinatorial theory (B)*, 12:50–71, 1972.
- [35] M.R. Garey and D.S. Johnson. *Computers and Intractability*. W.H. Freeman and Company, 1979.
- [36] F.R. Giles and W.R. Pulleyblank. Total dual integrality and integer polyhedra. *Linear Algebra and its Applications*, 25:191–196, 1979.
- [37] M.X. Goemans. Embeddings of metrics and multicommodity flow. *Lecture notes in l'Alpe d'Huez, march 2000*.
- [38] M.X. Goemans. Semidefinite programming in combinatorial optimization. *Mathematical Programming*, 79:143–161, 1997.
- [39] M.X. Goemans and L. Tunçel. When does the positive semidefiniteness constraint help in lifting procedures. *Manuscript*, 2000.
- [40] M.X. Goemans and D.P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J.ACM*, 42:1115–1145, 1995.
- [41] M.X. Goemans and D.P. Williamson. Approximation algorithms for max-3-cut and other problems via complex semidefinite programming. *Working paper*, 2001.
- [42] R. Gomory. An algorithm for the mixed integer problem. *The rand Corporation*, RM-2597, 1960.
- [43] R. Gomory. Solving linear programming problems in integers. *Combinatorial Analysis*, American Mathematical Society:211–216, 1960.
- [44] P. Gordan. Ueber die Auflösung linearer Gleichungen mit reellen Coefficienten. *Mathematische Annalen*, 6:23–28, 1873.
- [45] M. Grötschel, L. Lovász, and A. Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1:169–197, 1981.
- [46] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer Verlag, 1988.
- [47] C. Helmberg, S. Poljak, F. Rendl, and H. Wolkowicz. Combining semidefinite and polyhedral relaxations for integer programs. *IPCO IV Proceedings*, Springer, pages 124–134, 1995.

- 
- [48] Ch. Hermite. Extrait de lettres de M Ch. Hermite à M Jacobi sur différents objets de la théorie des nombres. *Journal für die reine und ungewandte Mathematik*, 40:261–315, 1850.
- [49] L.V. Kantorovitch. *Mathematical Methods of Organizing and Planning Production*. Publication House of the Leningrad State University, Leningrad, 1939.
- [50] R.M. Karp and C.H. Papadimitriou. On linear characterizations of combinatorial optimization problems. *SIAM Journal on Computing*, 11:620–632, 1982.
- [51] L. Kleinrock. *Queuing systems*. Wiley Interscience, 1975.
- [52] D.E. Knuth. The sandwich theorem. *The Electronic Journal of Combinatorics*, vol 1, 1994.
- [53] H.W. Kuhn. Solvability and consistency for linear equations and inequalities. *The American Mathematical Monthly*, 63:217–232, 1956.
- [54] H.W. Kuhn and A.W. Tucker. Nonlinear programming. *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, pages 481–492, 1950.
- [55] A. Laugier. A polyhedral cutting plane algorithm using matrix cuts. *ECCO99*, 1999.
- [56] M. Laurent and S. Poljak. On a positive semi-definite relaxation of the cut polytope. *Linear Algebra and its Applications*, 224:439–461, 1995.
- [57] A.K. Lenstra, H.W. Lenstra, and L. Lovász. Factoring polynomial with rational coefficients. *Mathematische Annalen*, 261:515–534, 1982.
- [58] H.W. Lenstra. Integer programming with a fixed number of variables. *Mathematics of Operations Research*, 8:538–548, 1983.
- [59] N. Linial, E. London, and Y. Rabinovitch. The geometry of graphs and some of its algorithmic applications. *35<sup>th</sup> Annual Symposium on Foundations of Computer Science*, pages 577–591, 1994.
- [60] L. Lovász. On the Shannon capacity of a graph. *IEEE transaction on Information Theory*, 25:1–7, 1979.
- [61] L. Lovász. Submodular functions and convexity. *Mathematical Programming, The State of Art*, Springer Verlag, Edited by A. Bachem, M. Grötschel, B. Korte, 1982.
- [62] L. Lovász. Semidefinite programs and combinatorial optimization. *Lectures notes*, 1995.
- [63] L. Lovász and H.E. Scarf. The generalized basis reduction algorithm. *Mathematics of Operations Research*, Vol 17:751–764, 1992.
- [64] L. Lovász and A. Schrijver. Cones of matrices and set-functions and 0-1 optimization. *SIAM Journal on Optimization*, Vol 1, number 2:166–190, 1991.
- [65] P. Mahey, J. Chifflet, and Sun Xiao Chao. Optimal design packet-switched networks under grade of service constraints. *Informs*, 1996.

- 
- [66] P. Mahey and A. Ouorou. Méthode de décomposition proximale et problèmes monotropiques. *Rapport de recherche Imag*, 1994.
- [67] P. Mahey and A. Ouorou. A minimum mean cycles canceling method for nonlinear multicommodity Flow Problems. *EJOR*, 121:532–548, 2000.
- [68] J.F Maurras. *Complexité et LP-Réduction*. Manuscript, 1997.
- [69] H. Minkowski. Geometrie der Zahlen (Erste Lieferung). *Teubner, Leipzig*, 1896.
- [70] G. Monge. Mémoire sur la théorie des déblais et des remblais. *Histoire de l'Académie Royale des Sciences, Mémoires de l'Académie Royale de Mathématique et de Physique*, pages Histoire :34–38, Mémoire :666–704, 1784.
- [71] T.S. Motzkin. Beiträge zur Theorie der linearen Ungleichungen. 1936.
- [72] G.L. Nemhauser and L.A. Wolsey. *Integer and Combinatorial Optimization*. Wiley Interscience, Series in Discrete Mathematics and Optimization, 1988.
- [73] A. Nemirovski. What can be expressed via conic quadratic and semidefinite programming. *Lecture notes*.
- [74] A. Ouorou. Décomposition proximale des problèmes de multiflot à critère convexe. application au problème de routage dans les réseaux de télécommunications. *PhD thesis, Université Blaise Pascal*, 1995.
- [75] A. Ouorou, P. Mahey, and L.-Ph Vial. A Survey of Algorithms for Convex Multicommodity Flow Problems. *Management Science*, Vol 46-1:126–147, 2000.
- [76] L. Porkolab and L. Khachiyan. On the complexity of semi-definite programs. *Rutcor Research Report*, RRR 40-95, 1995.
- [77] M.V. Ramana. An exact duality theory for semi-definite programming and its complexity implications. *Math Programming*, 1996.
- [78] M.V. Ramana, L. Tunçel, and H. Wolkowicz. Strong duality for semi-definite programming. *University of Waterloo Technical report CORR 95-12*, 1995.
- [79] A. Schrijver. *Theory of linear and integer programming*. Wiley Interscience, Series in Discrete Mathematics and Optimization, 1986.
- [80] A. Sebő. Potentials in undirected graphs and planar multiflows. *SIAM, J. Comput.*, vol 26:582–603, 1997.
- [81] H.D. Sherali and W.P. Adams. A hierarchy of relaxations between the continuous and convex hull representation for zero-one programming problems. *SIAM Journal on Discrete Mathematics*, vol 3:411–430, 1990.
- [82] N.Z. Shor. Utilization of the operation of space dilation in the minimization of convex functions. *Cybernetics*, vol 6:7–15, 1970.
- [83] N.Z. Shor. Cut-off method with space extension in convex programming problems. *Cybernetics*, vol 13:94–96, 1977.

- [84] J.G. van der Corput. Ueber diophantische systeme von linear-homogenen gleichungen und ungleichungen. *Proceedings Koninklijke Akademie van Wetenschappen te Amsterdam*, 34:372–382, 1931.
- [85] L. Vandenberghe and S. Boyd. Semidefinite programming. *Siam Review*, 1996.
- [86] J. von Neuman. Discussion of maximum problem. *Unpublished working paper*, 1947.
- [87] G. Voronoï. Nouvelles applications des paramètres continus à la théorie des formes quadratiques-Premier Mémoire. Sur quelques propriétés des formes quadratiques positives parfaites. *Journal für die reine und angewandte Mathematik*, 133:97–178, 1908.
- [88] R. Weismantel. Knapsack problems, test sets and polyhedra. *Habilitations-Schrift*, 1995.
- [89] H. Weyl. Elementare theorie der konvexen polyeder. *Commentarii Mathematici Helvetici*, 7:290–306, 1935.

# Two-connected graphs with given diameter

Aubin Jarry<sup>†</sup>  
Mascotte – CNRS/INRIA/UNSA  
2004, route des Lucioles  
F-06902 Sophia-Antipolis Cedex

Alexandre Laugier<sup>‡</sup>  
FTR&D/DMI/ISE  
905, rue Albert Einstein  
F-06921 Sophia-Antipolis Cedex

March 4, 2002

## Abstract

The problem of interest is to determine the minimum number edges of a 2-connected graph having a diameter equal to  $D$ . This problem deals with telecommunication survivable networks design with constraint of grade of service. In this article we prove the bounds of the number edges for 2-connected graphs or for 2-edges-connected graphs with a diameter  $D$ .

## 1 Introduction

One way to protect networks when some failure arises is to provide a sufficient connectivity to the network. In view of the robustness of the equipments and links 2-connectivity or 2-edge-connectivity is sufficient. Note also that  $k$ -connectivity,  $k \geq 3$  will lead to build too expensive networks. Another important parameter of a telecommunication network is the delay of transmission (or crossing delay). In first approximation, keeping the crossing delay under reasonable values corresponds to bound the diameter of the network, and the cost of the network depends on the number of links. So we are interested in finding the minimum number of edges of a 2-connected (2-edge-connected) graph of given diameter  $D$ . In this paper, we determine exactly these bounds, answering an old conjecture of the litterature due to L. Cacetta, see [Cac76].

**Theorem 1** *Let  $G = (V, E)$  be an undirected 2-connected graph of diameter  $D \geq 2$ . Then  $|E| \geq \lceil \frac{|V|D - (2D+1)}{D-1} \rceil$ .*

**Theorem 2** *Let  $G = (V, E)$  be an undirected 2-edge-connected graph (or multigraph) of diameter  $D \geq 2$ . If  $D$  is odd, then  $|E| \geq \lceil \frac{|V|D - (2D+1)}{D-1} \rceil$ . If  $D$  is even, then  $|E| \geq \min(\lceil \frac{|V|D - (2D+1)}{D-1} \rceil, \lceil \frac{(|V|-1)(D+1)}{D} \rceil)$ .*

In section 2, we first recall useful definitions and some results known on this kind of problem ; then we will prove our theorems (sections 3, 4 and 5).

## 2 Definition and previous results

$G = (V, E)$  will denote a graph with vertex set  $V$  and edge set  $E$ .  $n = |V|$  will denote the number of vertices (the order of  $G$ ). A graph is said to be  $k$ -connected (resp.  $k$ -edge-connected) if it remains connected after the deletion of any set of  $k - 1$  vertices (resp.  $k - 1$  edges). The distance  $d(x, y)$  between two vertices is the length of a shortest path between these two vertices.

The diameter  $D$  of the graph is the largest distance between any pair of vertices. As we said in the introduction we are interested in determining the minimum number of edges of a 2-connected (2-edge-connected) graph of diameter  $D$ . That is a particular case of the more general problem used in [MV64] of determining  $f(n, D, D', k - 1)$ , the minimum number of edges of a graph of order  $n$ , diameter  $D$  and such that if we remove any set of  $k - 1$  vertices, the resulting graph  $G'$  has diameter at most  $D'$ . If  $D' = n - 1$  that corresponds to ask that  $G$  remains connected after deletion of any  $k - 1$  vertices, and so  $f(n, D, n - 1, k - 1)$  is the minimum number of edges of a  $k$ -connected graph of diameter  $D$ . The case of  $k = 2$ , that is 2-connected, has received particular attention. For  $k = 2$  and  $D = 2$ , Murty [Mur68, Mur69] has shown that for  $D' \geq 3$   $f(n, 2, D', 1) = 2n - 5$ . The problem has also been solved for  $D = 3$  and  $D = 4$  at least for  $n$  not too small [Cac76, Cac79] and for  $D' \leq 6$  [Usa85]. All these results are explain [BBPP83]

Let  $G$  be the graph consisting of  $p$  internally disjoint paths joining 2 vertices  $x$  and  $y$ , one of length  $D+1$ ,  $p-2$  of length  $D$  and the last of length  $h \leq D$ . It has  $n = 2D+1+(p-2)(D-1)+(h-1)$  vertices and  $|E| = 2D+1+(p-2)D+h$  edges, so  $|E| \geq \lceil \frac{nD-(2D+1)}{D-1} \rceil$ . It is 2-connected of diameter  $D$  and its diameter increases to at most  $2D - 1$ . So for  $n > D' \geq 2D - 1$ ,  $f(n, D, D', 1) \leq \lceil \frac{nD-(2D+1)}{D-1} \rceil$ .

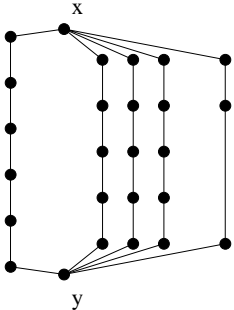


Figure 1:  $|E| = \lceil \frac{nD-(2D+1)}{D-1} \rceil$

Cacetta [Cac76] conjectured that  $f(n, D, D', 1) = \lceil \frac{nD-(2D+1)}{D-1} \rceil$  for  $D' \geq 2D - 1$ . Bollobás [Bol68] proved that  $\lim_{n \rightarrow \infty} f(n, D, D', 1) = \frac{D}{D-1}$  for  $D' \geq 2D - 2$  ( $D \geq 1$ ). Here we give a proof to Cacetta's conjecture. We recently discover another (more longer) proof has been published [EU99].

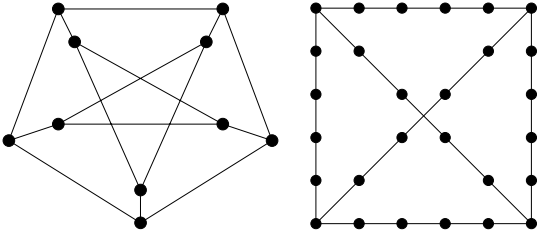


Figure 2: Other graphs with  $|E| = \lceil \frac{nD-(2D+1)}{D-1} \rceil$

In section 3, we introduce the notations and we give a way to decompose a 2-edge-connected graph. In section 4, we give the proof for 2-connected graphs. In section 5, we give the proof for 2-edge-connected graphs.

### 3 Preliminaries

Let  $G = (V, E)$  be a 2-connected graph of diameter  $D \geq 2$ . We call  $\ell_G = \max(0, |V| - (2D + 1))$  and  $k_G = |E| - |V|$ . Note that  $|E| \geq \lceil \frac{|V|D - (2D + 1)}{D - 1} \rceil$  is equivalent to  $k_G \geq \lceil \frac{\ell_G}{D - 1} \rceil$ . We will prove the bound  $k_G \geq \lceil \frac{\ell_G}{D - 1} \rceil$  by induction. It holds true for the cycles  $C_k$ . We assume that every 2-connected graph  $G'$  of diameter  $D' \leq D$  containing strictly less than  $|E|$  edges verifies  $k_{G'} \geq \lceil \frac{\ell_{G'}}{D' - 1} \rceil$ . We call  $C$  a smallest cycle of  $G$ , that is, a cycle such that every cycle in  $G$  contains at least  $|V_C|$  vertices. We will study the cases when  $G - C$  is disconnected, when  $G - C$  is connected and  $|V_C| \leq 2D$ , and eventually when  $|V_C| = 2D + 1$ . Note that if we add to graph  $G$  satisfying the hypothesis a path of length  $k$ ,  $k \leq D - 1$  we obtain a new graph satisfying the hypothesis. Such a path will be called an ear.

#### 3.1 Notation

Let  $G = (V, E)$  be an undirected graph. For all  $S \subset V$  we call  $d(x, S) = \min_{y \in S} d(x, y)$ . For all subgraph  $G' = (V', E')$  of  $G$ , we call  $G - G'$  or  $G - V'$  the subgraph generated by  $V - V'$ .

##### Definition 1 (path)

Let  $x_0, \dots, x_k \in V$  and  $e_1, \dots, e_k \in E$  such that  $\forall i \in \{1, \dots, k\}$ ,  $e_i = (x_{i-1}, x_i)$  and  $\forall j \neq i$ ,  $e_j \neq e_i$ . Then  $\mu = e_1 e_2 \dots e_k$  is called path of length  $k$  linking  $x_0$  to  $x_k$ . We define the two following sets  $V_\mu^{ext} = \{x_0, x_k\}$ ,  $V_\mu^{int} = \{x_1, \dots, x_{k-1}\}$  and  $E_\mu = \{e_1, \dots, e_k\}$ .

##### Definition 2 (ear)

Let  $\mu$  be a path.  $\mu$  is an ear if  $\forall x \in V_\mu^{ext}$ ,  $\deg(x) \neq 2$  and  $\forall x \in V_\mu^{int}$ ,  $\deg(x) = 2$ . For instance in the graph of figure 1 each path linking  $x$  to  $y$  is an ear.

##### Definition 3 (decomposition)

Let  $S \subset V$  and let  $V_1, V_2, \dots$  be the connected components of  $G - S$ . Let  $P_i$  be the subgraph generated by  $S \cup V_i$ .  $P_1, P_2, \dots$  is called a decomposition of  $G$  through  $S$ .

**Remark :** the decomposition is empty if  $S = V$ .

#### 3.2 Ear-partition

In this subsection, we show that a 2-edge-connected graph  $G$  can be partitionned into a given subgraph  $(S, E(S))$  and paths of bounded length  $\beta$ , these paths will be called ears. Intuitively we may say that starting from a subgraph generated as seen previously and if we add ears length at most  $k$  and repeat this operation we obtain the whole graph. We also give a property over  $\beta$  (claim 2).

##### Definition 4 ( $\overline{V_{CS \supset}(G, k)}$ )

Let  $G = (V, E)$  be an undirected 2-edge-connected graph. Let  $S$  be a non-empty subset of  $V$ . We call  $P_{CS \supset}(G, k)$  the set of all the paths of length  $k$  or less with their extremities in  $S$ . We call  $V_{CS \supset}(G, k)$  the vertex set of all these paths :  $V_{CS \supset}(G, k) = S \cup (\bigcup_{\mu \in P_{CS \supset}(G, k)} V_\mu^{int})$ . We call  $\overline{V_{CS \supset}(G, k)}$  the smallest set  $S'$  containing  $S$  such that  $V_{CS' \supset}(G, k) = S'$ .

##### Definition 5 ( $\beta(G, S)$ )

Let  $G = (V, E)$  be an undirected 2-edge-connected graph. Let  $S$  be a non-empty subset of  $V$ . We define  $\beta(G, S)$  as the smallest integer  $k \geq 0$  such that  $\overline{V_{CS \supset}(G, k)} = V$ .

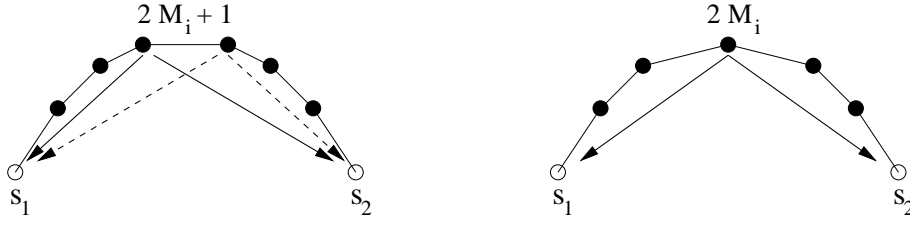


Figure 3:  $\beta$

**Claim 1 (ear-partition)** Let  $G = (V, E)$  be an undirected 2-edge-connected graph. Let  $G' = (V', E')$  be a non-null subgraph of  $G$  with  $V' \neq V$ . Let  $\beta = \beta(G, V')$ . Then  $G$  contains  $(|V'| + \ell)$  vertices and  $(|E'| + \ell + k)$  edges, with  $k \geq \lceil \frac{\ell}{\beta-1} \rceil$ .

**Proof**

We will prove this claim by induction on  $|V| - |V'|$ . Let  $\mu$  be a non-empty path of length smaller than  $\beta$  such that  $V_\mu^{ext} \subset S$  and  $V_\mu^{int} \cap S = \emptyset$ .  $\beta(G, S \cup V_\mu^{int}) \leq \beta$ , so  $G$  contains  $(|V'| + |V_\mu^{int}| + \ell')$  vertices and  $(|E'| + |E_\mu| + \ell' + k')$  edges, with  $k' \geq \lceil \frac{\ell'}{\beta-1} \rceil$ . Since  $|E_\mu| \geq \lceil \frac{|V_\mu^{int}|}{\beta-1} \rceil$ ,  $G$  contains  $(|V'| + \ell)$  vertices and  $(|E'| + \ell + k)$  edges, with  $k \geq \lceil \frac{\ell}{\beta-1} \rceil$ .

**Lemma 1** Let  $G = (V, E)$  be an undirected 2-edge-connected graph. Let  $S$  be a non-empty subset of  $V$ . For all  $0 \leq k \leq \lfloor \frac{\beta(G, S)}{2} \rfloor$ , there is  $x \in V$  such that  $d(x, S) = k$ .

**Proof**

Let  $\beta = \beta(G, S)$  and  $S' = \overline{V_{CS \supset}(G, \beta - 1)}$ . Let  $\mu$  be a shortest path with  $V_\mu^{ext} \subset S'$ ,  $V_\mu^{int} \neq \emptyset$  and  $V_\mu^{int} \cap S' = \emptyset$ . The length of  $\mu$  is greater or equal to  $\beta$ , so there is a vertex  $y \in V$  such that  $d(y, S') \geq \lfloor \frac{\beta}{2} \rfloor$ . Thus  $d(y, S) \geq \lfloor \frac{\beta}{2} \rfloor$ . Let  $\mu'$  be a shortest path from  $y$  to  $S$ . For every  $0 < k < d(y, S)$ , we have  $x \in V_\mu^{int}$  such that  $d(x, S) = k$ .

**Claim 2 (diameter)** Let  $G = (V, E)$  be an undirected 2-connected graph and let  $S \subset V$  such that  $|S|$  disconnects  $G$ . Let  $P_1, P_2, \dots$  be a decomposition of  $G$  through  $S$ . Then  $\forall i \neq j$ ,  $\beta(P_i, S) + \beta(P_j, S) \leq 2D + 1$ .

**Proof**

Let  $\beta_i = \beta(P_i, S)$  and  $\beta_j = \beta(P_j, S)$ .

- if  $\beta_i$  or  $\beta_j$  is even, then through lemma 1 we have  $\lfloor \frac{\beta_i}{2} \rfloor + \lfloor \frac{\beta_j}{2} \rfloor \leq D$ , so  $\beta_i + \beta_j \leq 2D + 1$ .
- if both  $\beta_i$  and  $\beta_j$  are odd :

According to lemma 1, we have  $x_i \in V_{P_i}$  such that  $d(x_i, S) = \lfloor \frac{\beta_i}{2} \rfloor$ . Let  $s_i \in S$  such that  $d(x_i, s_i) = d(x_i, S)$ . Since  $\beta_i$  is odd,  $d(x_i, S \setminus \{s_i\}) \geq \lfloor \frac{\beta_i}{2} \rfloor + 1$ . Let  $S' = \overline{V_{CS \supset}(P_j, \beta_j - 1)}$ . Since  $G$  is 2-connected,  $P_j$  is 2-connected so there is a non-empty set  $V'$  of vertices  $x$  in  $V_{P_j}$  such that  $d(x, S' \setminus \{s_i\}) = d(x, S') > 0$ . Since  $\beta(P_j, S') = \beta_j$ , there is  $x_j \in V'$  such that  $d(x_j, S') = \lfloor \frac{\beta_j}{2} \rfloor$ . Since  $\beta_j$  is odd,  $d(x_j, s_i) \geq \lfloor \frac{\beta_j}{2} \rfloor + 1$ . Thus,  $d(x_i, x_j) \geq \lfloor \frac{\beta_i}{2} \rfloor + \lfloor \frac{\beta_j}{2} \rfloor + 1$ , so  $\lfloor \frac{\beta_i}{2} \rfloor + \lfloor \frac{\beta_j}{2} \rfloor + 1 \leq D$ , which implies  $\beta_i + \beta_j \leq 2D + 1$ .



## 4 Proof of the bound for 2-connected graphs

In this section we prove that the bound  $\lceil \frac{|V|D-(2D+1)}{D-1} \rceil$  is best possible for 2-connected graphs. We will distinguish three cases. These cases are related to the length of a smaller cycle disconnecting the graph  $G$ . We will see that the number of connected components obtained by removing such a cycle is still important.

### 4.1 Smallest cycle disconnecting $G$

**Lemma 2** *Let  $C$  be a smallest cycle in  $G$ . Let  $P_1, P_2, P_3 \dots$  be a decomposition of  $G$  through  $V_C$ . Then  $P_i$  is 2-connected and its diameter is smaller or equal to  $D$ .*

**Proof**

Let  $x \in V_{P_i}$ . If  $x \in V_C$ ,  $(P_i - C)$  is connected to  $C$  by at least two vertices, so  $P_i - \{x\}$  remains connected. If  $x \notin V_C$ , each connected component of  $(P_i - C) - \{x\}$  is connected to  $C$ , so  $P_i - \{x\}$  is connected. This proves that  $P_i$  is 2-connected.

Let  $y \in V_{P_i} \setminus \{x\}$ . If a shortest path from  $x$  to  $y$  is included in  $P_i$ , then the distance between  $x$  and  $y$  is the same in  $P_i$  and in  $G$ . Otherwise, there is a shortest path  $\mu_1\mu_2\mu_3$  from  $x$  to  $y$  with  $\mu_1, \mu_3$  included in  $P_i$  and  $\mu_2$  with no edge in  $P_i$  linking  $a \in V_C$  to  $b \in V_C$ . Thus  $\mu_2$  is a shortest path from  $a$  to  $b$  strictly shorter than any path included in  $C$ . This contradicts the fact that  $C$  is a smallest cycle. This proves that the diameter of  $P_i$  is smaller or equal to  $D$ .

**Claim 3 (disconnect)** *Let  $C$  be a smallest cycle of  $G$ . If  $G - C$  is not connected, then  $k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$ .*

**Proof**

Let  $P_1, P_2, P_3 \dots$  be a decomposition of  $G$  through  $V_C$ . We call  $\beta_i = \beta(P_i, V_C)$ ,  $\ell_i = |V_{P_i}| - |V_C|$  and  $k_i = |E_{P_i}| - |V_{P_i}|$ . Claim 2 (diameter) shows that one  $P_i$  at most (say  $P_1$ ) satisfies  $\beta_i \geq D + 1$ .  $G$  contains  $|V_{P_1}| + \ell$  vertices and  $|E_{P_1}| + \ell + k$  edges, where  $\ell = \sum_{i \neq 1} \ell_i$  and  $k = \sum_{i \neq 1} k_i$ . According to claim 1 (ear-partition),  $\forall i \neq 1$   $k_i \geq \lceil \frac{\ell_i}{\beta_i - 1} \rceil$ , which implies  $k \geq \lceil \frac{\ell}{D-1} \rceil$ . According to lemma 2,  $P_1$  is 2-connected and its diameter is smaller or equal to  $D$ , and we have by induction  $k_{P_1} \geq \lceil \frac{\ell_{P_1}}{D-1} \rceil$  so  $k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$ .

### 4.2 Smallest cycle of size smaller than $2D$

**Lemma 3** *Let  $C$  be a smallest cycle of  $G$ . If the length of any ear of  $G$  included in  $C$  is greater than  $D + 1$ , then  $k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$ .*

**Proof**

Let  $\mu$  be an ear of length greater than  $D + 1$  included in  $C$ . Let  $P_1 = (V_\mu^{int} \cup V_\mu^{ext}, E_\mu)$ ,  $P_2, \dots$  be a decomposition of  $G$  through  $V_\mu^{ext}$ .  $\beta(P_1, V_\mu^{ext}) \geq D + 1$ , so according to claim 2,  $\forall i > 1$ ,  $\beta(P_i, V_\mu^{ext}) \leq D$ . This implies that  $\beta(G, V_C) \leq D$ . According to claim 1 (ear-partition),  $G$  contains  $|V_C| + \ell$  vertices and  $|E_C| + \ell + k$  edges, with  $k \geq \lceil \frac{\ell}{D-1} \rceil$ . Since  $C$  is 2-connected of diameter smaller than  $D$ , we have  $k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$ .

**Lemma 4** *Let  $C$  be a cycle in  $G$  such that  $G - C$  is connected. Let  $\mu$  be an ear of  $G$  included in  $C$ . Then  $G - V_\mu^{int}$  is 2-connected.*

**Proof**

Let  $\mu$  be an ear of  $G$  included in  $C$ . Let  $x \in V \setminus V_\mu^{int}$ . If  $x \in V_C$ ,  $(C - V_\mu^{int}) - \{x\}$  has at

most two connected components, each of them is connected to  $G - C$ . Since  $G - C$  is connected,  $(G - V_\mu^{int}) - \{x\}$  is connected as well. If  $x \notin V_C$ , since  $G$  is 2-connected, each connected component of  $(G - C) - \{x\}$  is connected to  $C - V_\mu^{int}$ . Therefore  $(G - V_\mu^{int}) - \{x\}$  is connected.

**Claim 4 (big ear)** *Let  $C$  be a smallest cycle of  $G$  such that  $G - C$  is connected. If an ear of  $G$  included in  $C$  contains more than  $\lceil \frac{|V_C|}{2} \rceil$  edges, then  $k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$ .*

**Proof**

Let  $\mu$  be a an ear of  $G$  included in  $C$ . If its length is greater than  $D + 1$  then lemma 3 tells that  $k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$ . Otherwise, according to lemma 4,  $G - V_\mu^{int}$  is a 2-connected graph. It contains  $(|V| - \ell)$  vertices and  $(|E| - \ell - k)$  edges, with  $k \geq \lceil \frac{\ell}{D-1} \rceil$ . If the length of  $\mu$  is greater than  $\lceil \frac{|V_C|}{2} \rceil$ , the diameter of  $G - V_\mu^{int}$  is smaller or equal to  $D$  so  $k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$ .

#### 4.2.1 Flatening

We introduce here a way to transform a cycle into a tree.

**Definition 6 (flatening)**

Let  $C$  be a cycle. A flatening of  $C$  is a pair  $(T, \sim)$  where  $T$  is a tree containing  $\lceil \frac{|V_C|+1}{2} \rceil$  vertices and where  $\sim$  is an equivalence relation such that  $T = \frac{C}{\sim}$ .

**Lemma 5 (legal flatening)** *Let  $C$  be a cycle and  $S \subset V_C$ . If the length of the longest ear of  $C$  with no vertex in  $S$  is smaller than  $\lfloor \frac{|V_C|}{2} \rfloor$  then there is a flatening  $(T, \sim)$  of  $C$  such that each leaf of  $T$  is a class containing an element of  $S$ .*

**Proof**

If  $|V_C|$  is even then let  $s_1$  and  $s_2$  be vertices of  $S$  such that  $d(s_1, s_2)$  is maximal.

- If  $d(s_1, s_2) = \frac{|V_C|}{2}$ , we set  $(x \sim y) \Leftrightarrow d(x, s_1) = d(y, s_1)$ .

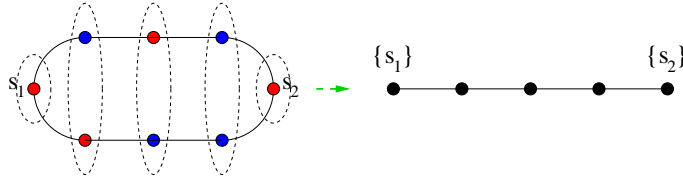


Figure 4: Flatening 1

- Otherwise  $S$  contains a vertex  $s_3$  such that each connected component of  $C - \{s_1, s_2, s_3\}$  contains at most  $(\frac{|V_C|}{2} - 2)$  vertices. We define  $D_1 = \frac{d(s_1, s_2) + d(s_1, s_3) - d(s_2, s_3)}{2}$ ,  $D_2 = \frac{d(s_1, s_2) + d(s_2, s_3) - d(s_1, s_3)}{2}$  and  $D_3 = \frac{d(s_1, s_3) + d(s_2, s_3) - d(s_1, s_2)}{2}$ . We have  $D_1 + D_2 + D_3 = \frac{|V_C|}{2}$ . We set  $(x \sim y) \Leftrightarrow \exists i$  such that  $d(x, s_i) = d(y, s_i) \leq D_i$

If  $|V_C|$  is odd we merge two consecutive vertices  $(x_1, x_2)$  of  $C$  ( $f(x_1) = f(x_2)$  and  $\forall x, y \in V_C \setminus \{x_1\}, x \neq y \Rightarrow f(x) \neq f(y)$ ). We obtain a cycle  $C'$  of even size  $(|V_C| - 1)$ . The length of any ear of  $C'$  with no vertex in  $f(S)$  is at most  $\frac{|V_C|-1}{2}$  so there is a flatening  $(T, \sim)$  of  $C'$  such that each leaf of  $T$  is a class containing an element of  $f(S)$ . We call  $\sim_f$  the equivalence relation such that  $x \sim_f y \Leftrightarrow f(x) \sim f(y)$ .  $(T, \sim_f)$  is a flatening of  $C$  such that each leaf of  $T$  is a class containing an element of  $S$ .

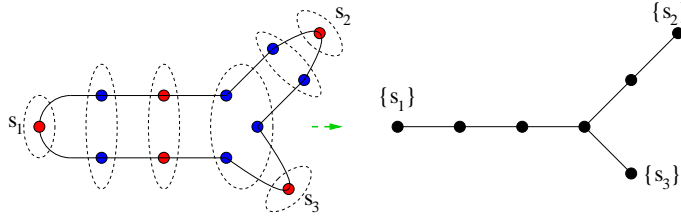


Figure 5: Flatening 2

#### 4.2.2 Application

We show that a certain flatening on a smallest cycle of  $G$  keeps the graph 2-connected. If the cycle is small enough, it boils down to remove an ear of length  $\beta \leq D$ .

**Lemma 6** *If there is a cycle  $C$  of  $G$  such that  $G - C$  is 2-connected and such that no ear of  $G$  included in  $C$  is longer than  $\lfloor \frac{|V_C|+1}{2} \rfloor$ , then there is an equivalence relation  $\sim$  on  $G$  such that  $\frac{G}{\sim}$  is a 2-connected graph containing  $|V| - \ell$  vertices and  $|E| - \ell - k$  edges, where  $\ell = \lceil \frac{|V_C|-1}{2} \rceil$  and  $k \geq 1$ .*

**Proof**

Let  $S = \{x \in V_C \mid \deg(x) \geq 3\}$ . According to lemma 5 (legal flatening), there is a flatening (claim 5)  $(T, \sim)$  of  $C$  such that each leaf of  $T$  is in  $\frac{S}{\sim}$ . We naturally extend  $\sim$  on  $G$  ( $\forall x \notin V_C, x \sim y \Leftrightarrow x = y$ ). Let  $x \in \frac{V}{\sim}$ . If  $x \notin V_T$ , then  $x$  is a class containing a single vertex of  $V$  so  $\frac{G}{\sim} - \{x\}$  is connected. If  $x \in V_T$ , then  $T - \{x\}$  is a reunion of trees. Each tree is connected to  $\frac{G-C}{\sim}$ . Since  $G - C$  is connected, so is  $\frac{G}{\sim} - \{x\}$ . So  $\frac{G}{\sim}$  is a 2-connected graph containing  $|V| - \lceil \frac{|V_C|}{2} \rceil + 1$  vertices and less than  $|E| - \lceil \frac{|V_C|}{2} \rceil$  edges.

**Claim 5 (small smallest cycle)** *Let  $C$  be a smallest cycle of  $G$  such that  $G - C$  is connected. If  $|V_C| \leq 2D$ , then  $k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$ .*

**Proof**

According to claim 4 (big ear), if the length of the biggest ear of  $C$  with no vertex in  $S$  is greater than  $\lfloor \frac{|V_C|+1}{2} \rfloor$ , then  $k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$ . Otherwise, according to lemma 6, there is an equivalence relation  $\sim$  on  $G$  such that  $\frac{G}{\sim}$  is a 2-connected graph containing  $|V| - \ell$  vertices and  $|E| - \ell - k$  edges, where  $\ell = \lceil \frac{|V_C|-1}{2} \rceil$  and  $k \geq 1$ . The diameter of  $\frac{G}{\sim}$  is smaller or equal to  $D$ . If  $|V_C| \leq 2D$ , then  $k \geq \lceil \frac{\ell}{D-1} \rceil$ . Therefore  $k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$ .

#### 4.3 Smallest cycle of size $2D + 1$

If the size of every cycle in  $G$  is greater than  $2D + 1$ , we have strong indications on  $G$ .

**Lemma 7** *If the size of a smallest cycle of  $G$  equals  $2D + 1$ , then each vertex in  $V$  belongs to a cycle of size  $2D + 1$ .*

**Proof**

Let  $\{x, y\} \subset V$  such that  $d(x, y) = D$ . We choose a shortest path  $\mu_1$  from  $x$  to  $y$ . Since  $\deg(y) \geq 2$ , we choose a vertex  $z$  adjacent to  $y$  which is not on  $\mu_1$ . We choose a shortest path  $\mu_2$  from  $z$  to  $x$ . The path  $\mu = \mu_1(yz)\mu_2$  is a path linking  $x$  to itself which length is smaller than  $2D + 1$ . Since  $(yz)$  appears only once in this path,  $\mu$  contains a cycle which size is smaller than  $2D + 1$ . Since it can not be smaller than  $2D$ , the size of this cycle is exactly  $2D + 1$  and  $x$  belongs to this cycle.

**Lemma 8** *Let  $C$  be a smallest cycle of  $G$  of size  $2D + 1$ . If  $G \neq C$ , then  $\forall x \in V_C$ ,  $\deg(x) \geq 3$ .*

**Proof**

Let  $a_0, a_2 \dots a_{2D}$  be the vertices of  $C$  such that  $E_C = \{(a_i a_{i+1}), i \in \frac{\mathbb{Z}}{(2D+1)\mathbb{Z}}\}$ . Since  $G \neq C$  we assume that  $\deg(a_D) \geq 3$ .  $C$  is a smallest cycle of  $G$ , so  $d(a_0, a_D) = D$ . Let  $x \in V \setminus V_C$  be a neighbour of  $a_D$  and let  $\mu$  be a shortest path from  $x$  to  $a_0$ . Since  $d(x, a_0) \leq D$ ,  $E_\mu$  does not contain  $(x a_D)$ . Since there is no cycle of size smaller than  $2D$ ,  $V_C \cap V_\mu^{int} = \emptyset$ . Therefore,  $\deg(a_0) \geq 3$ . By symmetry,  $\deg(a_{2D}) \geq 3$ . By symmetry every vertex in the cycle satisfies  $\deg(a_i) \geq 3$ .

**Claim 6 (big smallest cycle)** *If the size of a smallest cycle of  $G$  equals  $2D + 1$ ,  $k_G \geq \lceil \frac{\ell}{D-1} \rceil$ .*

**Proof**

According to the two former lemmas, each vertex in  $G$  belongs to a cycle of size  $(2D + 1)$ , and if  $G$  is not a cycle, each vertex is of degree greater or equal to 3. If  $G$  is a cycle, then  $k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$ . Otherwise,

- If  $D \geq 3$ , we have  $2|E| \geq 3|V|$
- If  $D = 2$ , We choose a vertex  $s$  of highest degree in  $G$ , and we divide the other vertices into two sets :  $D_1 = \{x \in G \mid d(x, s) = 1\}$  and  $D_2 = \{y \in G \mid d(y, s) = 2\}$ .  $G$  does not contain any cycle of size smaller or equal to 4. This implies

$$\begin{aligned} \forall \{x_1, x_2\} \subset D_1, x_1 \neq x_2, & \quad (x_1 x_2) \notin E \\ \forall y \in D_2 \exists! x \in D_1, & \quad (xy) \in E \end{aligned}$$

So we divide  $D_2$  into the disjoint union :

$$D_2 = \bigcup_{x \in D_1} D_2^x \quad \text{with} \quad D_2^x = \{y \in D_2 \mid d(x, y) = 1\}$$

If  $\deg(s) = 3$ ,  $G$  contains exactly 10 vertices and 15 edges ( $G$  is the Petersen graph), so  $k_G \geq \ell_G$ . If  $\deg(s) \geq 4$ , we define :  $\forall x \in D_1, r_x = \deg(x) - 3$  and  $r = \sum_{x \in D_1} r_x$ . We have  $|V| = 1 + \deg(s) + (2\deg(s) + r)$  which is  $|V| - 1 = 3\deg(s) + r$ . Since

$$\forall x_1 \in D_1, \forall y_1 \in D_2^{x_1}, \forall x_2 \in D_1 \setminus \{x_1\}, \quad d(y_1, x_2) \leq 2$$

the degree of each vertex in  $D_2$  is greater or equal to  $|D_1| \geq 4$ . We add the degrees of all the vertices in  $G$  :

$$\begin{aligned} 2|E| &= \deg(s) + \sum_{x \in D_1} \deg(x) + \sum_{y \in D_2} \deg(y) \\ 2|E| &\geq \deg(s) + \deg(s) \times 3 + (\deg(s) \times 2 + r) \times 4 \\ 2|E| &\geq 4(\deg(s) \times 3 + r) \\ 2|E| &\geq 4(|V| - 1) \\ |E| - |V| &\geq |V| - 2 \end{aligned}$$

So  $k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$ .

#### 4.4 Recapitulation of the proof

**Theorem 1** *Let  $G = (V, E)$  be an undirected 2-connected graph of diameter  $D \geq 2$ .*

*Then  $|E| \geq \lceil \frac{|V|D - (2D+1)}{D-1} \rceil$ .*

**Proof**

If  $G$  is a cycle then  $k_G = \ell_G = 0$ . Otherwise, we assume than each 2-connected graph  $G'$  of diameter  $D' \leq D$  and containing  $|E'| < |E|$  edges satisfies  $k_{G'} \geq \lceil \frac{\ell_{G'}}{D-1} \rceil$ . Let  $C$  be a smallest cycle of  $G$  :

- if  $G - C$  is not connected, claim 3 (disconnect) tells that  $k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$ .
- if  $G - C$  is connected and  $|V_C| \leq 2D$ , claim 5 (small smallest cycle) tells that  $k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$ .
- if  $|V_C| = 2D + 1$ , claim 6 (big smallest cycle) tells that  $k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$ .

$k_G \geq \lceil \frac{\ell_G}{D-1} \rceil$  is equivalent to  $|E| \geq \lceil \frac{|V|D - (2D+1)}{D-1} \rceil$ .

### 5 Proof of the bound for 2-edge-connected graphs

Let  $G = (V, E)$  be a 2-edge-connected graph of diameter  $D \geq 2$ . We will prove the bounds by induction. Those bounds hold true for 2-connected graphs, and graphs with less than  $2D + 1$  vertices. We assume that for every 2-edge-connected graph  $G' = (V', E')$  of diameter  $D' \leq D$  with  $|E'| < |E|$ , the bounds hold true. We also assume that there is a vertex  $s$  such that  $G - \{s\}$  is not connected.

**Theorem 2** *Let  $G = (V, E)$  be an undirected 2-edge-connected graph (or multigraph) of diameter  $D \geq 2$ . If  $D$  is odd, then  $|E| \geq \lceil \frac{|V|D - (2D+1)}{D-1} \rceil$ . If  $D$  is even, then  $|E| \geq \min(\lceil \frac{|V|D - (2D+1)}{D-1} \rceil, \lceil \frac{(|V|-1)(D+1)}{D} \rceil)$ .*

**Proof**

Let  $P_1, P_2, P_3, \dots$  be a decomposition of  $G$  through  $\{s\}$ ,  $\beta_i = \beta(P_i, \{s\})$ ,  $\ell_i = |V_{P_i}| - 1$  and  $k_i = |E_{P_i}| - |V_{P_i}|$ . We set  $P_1$  such that  $\beta_1$  is maximal.

- if  $\lfloor \frac{\beta_1}{2} \rfloor < \frac{D}{2}$ , according to claim 1 (ear-partition),  $\forall i k_i \geq \lceil \frac{\ell_i}{D-1} \rceil$  so  $|E| \geq \lceil \frac{|V|D - (2D+1)}{D-1} \rceil$ .
- if  $\lfloor \frac{\beta_1}{2} \rfloor = \frac{D}{2}$  ( $D$  must be even), then  $\forall i \beta_i \leq (D + 1)$  and according to claim 1 (ear-partition),  $\forall i k_i \geq \lceil \frac{\ell_i}{D} \rceil$  so  $|E| \geq \lceil \frac{(|V|-1)(D+1)}{D} \rceil$ .
- if  $\lfloor \frac{\beta_1}{2} \rfloor > \frac{D}{2}$ , then  $\forall i \neq 1, \beta_i \leq D$ .  $G$  contains  $|V_{P_1}| + \ell$  vertices and  $|E_{P_1}| + \ell + k$  edges, where  $\ell = \sum_{i \neq 1} \ell_i$  and  $k = \sum_{i \neq 1} k_i$ . According to claim 1 (ear-partition),  $\forall i \neq 1 k_i \geq \lceil \frac{\ell_i}{D-1} \rceil$ , which implies  $k \geq \lceil \frac{\ell}{D-1} \rceil$ . Observe that  $P_1$  is 2-(edge)-connected and that its diameter is smaller or equal to  $D$ , so the bounds hold true for  $P_1$  :

- if  $D$  is odd, then  $|E_{P_1}| \geq \lceil \frac{|V_{P_1}|D - (2D+1)}{D-1} \rceil$  so  $|E| \geq \lceil \frac{|V|D - (2D+1)}{D-1} \rceil$ .
- if  $D$  is even, then  $|E_{P_1}| \geq \min(\lceil \frac{|V_{P_1}|D - (2D+1)}{D-1} \rceil, \lceil \frac{(|V_{P_1}|-1)(D+1)}{D} \rceil)$ ,  
so  $|E| \geq \min(\lceil \frac{|V|D - (2D+1)}{D-1} \rceil, \lceil \frac{(|V|-1)(D+1)}{D} \rceil)$ .

## 6 Further researchs

Since we have solved the problem for the case  $k = 2$  we will look after the minimum number of edges of graphs satisfying higher connectivity requirements. One can check that the graphs we propose contain some vertices with very large degree, so in order to provide more robust networks it may be interesting to introduce a bound on the maximum degree of the graph.

The weighted case, i.e. case the weight is not one, is  $\mathcal{NP}$ -hard. Indeed one can check it contains the minimum weight Hamiltonian tour. Thus one of our goal is to derive from the construction explained in this article an approximation algorithm for the 2-connected subgraph with diameter  $D$  of minimum weight. We showed that the bounds are valid for subgraphs of the complete graph, it seems interesting to determine the classes of graphs for which the subgraphs satisfying the  $(n, 2, D)$  property verify the bounds. In addition we want to study the cases for which the inequality  $\sum_{e \in E} x_e \geq \lceil \frac{nD - (2D+1)}{d-1} \rceil$  is facets defining of the polytope of the 2-connected subgraphs having diameter no greater than  $D$ .

## References

- [BBPP83] J.C. Bermond, J. Bond, M. Paoli, and C. Peyrat. Graphs and interconnection networks : Diameter and vulnerability. *London Mathematical Society Lectures Notes, Cambridge University*, pages 1–30, 1983.
- [Bol68] B. Bollobás. A problem of the theory of communication networks. *Acta Math. Acad. Sci. Hungar*, 19:75–80, 1968.
- [Cac76] L. Cacetta. Extremal graphs of diameter 4. *Journal of Combinatoric theory*, 21:104–115, 1976.
- [Cac79] L. Cacetta. Extremal graphs of diameter 3. *J.Austral.Math.Soc.Ser.A*, 28:76–94, 1979.
- [EU99] H. Enomoto and Y. Usami. Etremal 2-connected graphs of given diameter. *Tokyo J.Math*, 1:1–16, 1999.
- [Mur68] U.S.R. Murty. On some extremal graphs. *Acta.Math.Acad.Sci.Hungar*, 19:69–74, 1968.
- [Mur69] U.S.R. Murty. *Proof techniques in graph theory*, chapter Extremal nonseparable graphs of diameter 2. Academic press, 1969.
- [MV64] U.S.R. Murty and K. Vijayan. On accessiblility in graphs. *Sankhya (A)*, 26:221–234, 1964.
- [Usa85] Y. Usami. Etremal graph of diameter at most 6 after vertex deletion. *J. Graph Theory*, 9:221–234, 1985.

# Routing flows in networks with heterogeneous protocols and path-dependent edge costs

Sébastien Bertrand\*      Alexandre Laugier†      Philippe Mahey‡

March 4, 2002

## Abstract

The links which compose telecommunication networks are issued from various technologies and information has to be carried on each link with respect to a link protocol, inducing many encapsulation processes and extra traffic. In this paper we give a formulation of the Minimum Cost Routing Problem for networks with nested connections and heterogeneous protocols. This formulation relies on the notion of Non-Conservative Flows and exhibits edge costs which depend on the paths they belong. We prove that in this context, the Minimum Cost Routing Problem is *NP*-complete. We then develop a column-generation model of the Routing Problem and provide the reader with a heuristic algorithm to identify new columns.

## 1 Introduction

The heterogeneous nature of Interconnection Networks is traditionally regarded as not being of key importance when a Network Designer has to establish the routing of multiple demands, or design networks *ex nihilo*. Yet it is undeniable that the links of a network are issued from various technologies which range from basic copper wires to leased lines and optical fibres, and as interconnection networks have been designed and improved for the past twenty years, various protocols can coexist in the same network. For instance, communications issued from local centers of a company can be federated on X.25 trunks, and the information can then be carried to other centers via an ATM or a Frame Relay network. We believe that in order to devise new models closer to the reality of modern day networks we have to take this variety of links into account as early as possible in the process of optimisation.

If we investigate more profoundly the nature of telecommunication networks, we see that information is carried on each link with respect to a *link protocol*. Each data flow which crosses a link is encapsulated according to the link protocol, and a certain amount of overhead is then generated. For the case of an IP data flow carried on an ATM trunk, the overhead corresponds to the addition of 5 bytes to each packet obtained after segmentation of the IP packets into smaller packets of size 48 bytes each (see Figure 1). If the IP packet to be encapsulated into ATM is 1024 bytes long, the encapsulation results in a bandwidth increase equal to 14%, whereas encapsulating an IP packet of size 49 bytes results in an increase greater than 100%.

---

\*sebastien.bertrand@francetelecom.com, France Télécom R&D – LIMOS.

†alexandre.laugier@francetelecom.com, France Télécom R&D.

‡philippe.mahey@isima.fr, LIMOS. Research supported by France Télécom R&D under grant 99-1B-281.

The amount of overhead generated can therefore represent a significant part of the traffic which the users of the network are not supposed to pay for.

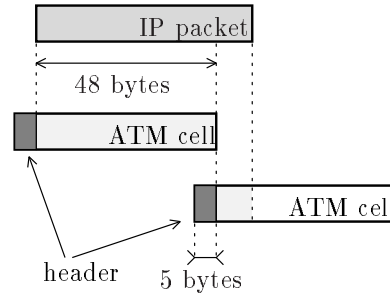


Figure 1: encapsulating one IP packet into two ATM cells

The networks we are going to study are organized in layers, each layer being composed of connections. A connection can be seen as the reservation of a certain amount of bandwidth between two communicating nodes, for a fixed duration. Each connection from a given layer possesses its own protocol and can be supported by a connection issued from a deeper layer, provided that these connections are correctly nested, i.e. the end nodes of any supporting connection can not be outside the scope of the supported connection (see Figure 2 for the correct nesting of any layer  $n$  and layer  $n + 1$  connections). The reader can find a detailed description of the encapsulation and decapsulation mechanisms in [LM00]. In the remainder of this article we will use the term *encapsulation* to indicate that a data flow carried on a connection belonging to any layer  $n$  is supported by a connection of layer  $n + 1$ .

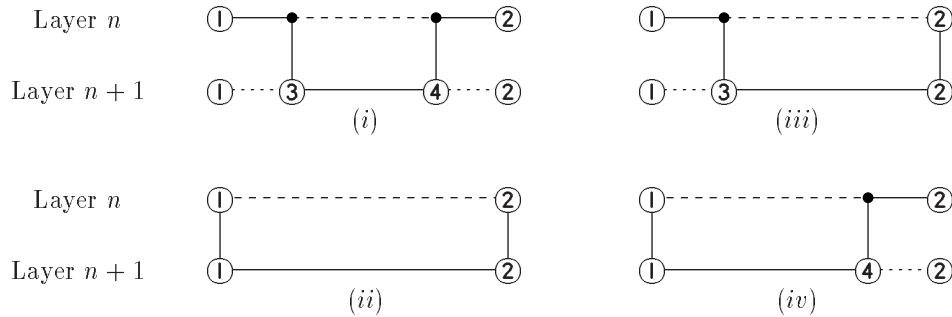


Figure 2: connections correctly nested

It is important to notice that we do not limit the number of times a data flow can be encapsulated. Indeed, it is not uncommon to encounter X.25 flows encapsulated into Frame Relay and then carried on ATM trunks at the core of large scale networks. Any data flow which is carried on a connection possesses an Encapsulation Level which reflects the number of overheads which have been added to this data flow in its native form. For example, consider a native flow of type IP crossing a connection whose protocol is X.25. Over this connection, the encapsulation level of the data flow will be equal to 1. Suppose the X.25 connection is supported by a Frame Relay connection: the encapsulation level of the IP data flow over this Frame Relay connection will then be 2. The corresponding flow, named “IP encapsulated into



X.25 encapsulated into Frame Relay”, will be denoted by  $IP \succ X25 \succ FR$ . We give in Figures 3 and 4 the example of a network composed of 3 layers, layer 0 representing the requests of connections between the nodes of the network. The nodes in black are the end nodes of the demands. These graphs should be understood as follows: there is a request of type IP between nodes 1 and 2. This connection can be supported by a X.25 connection between nodes 1 and 2. This X.25 connection can be supported between nodes 1 and 4 by a Frame Relay connection, and between nodes 4 and 3 by an ATM connection. The Frame Relay connection between nodes 1 and 4 can in turn be supported by an ATM connection.



Figure 3: sample Layer 0 and Layer 1 Connections



Figure 4: sample Layer 2 and Layer 3 Connections

In Figure 5 we give a recapitulation of the routing of the IP data flow between nodes 1 and 2 along a path composed of connections with various protocols which are supported by deeper connections. We represent the packets by triangles, and the overheads by smaller triangles. We give the Encapsulation Level of the data flow for each connection it crosses. The level entitled “Physical Layer” gives the type of the data flow which effectively circulates in the network between nodes 1, 4, 3 and 2.

We leave as an exercise for the reader the recapitulation of the routing of the data flow between nodes 1 and 3. Notice that in most cases there is more than one connection available to route a demand between any two given nodes, and the routing of the layer  $n$  connections over the connections of layer  $n + 1$  is not always known *a priori*.

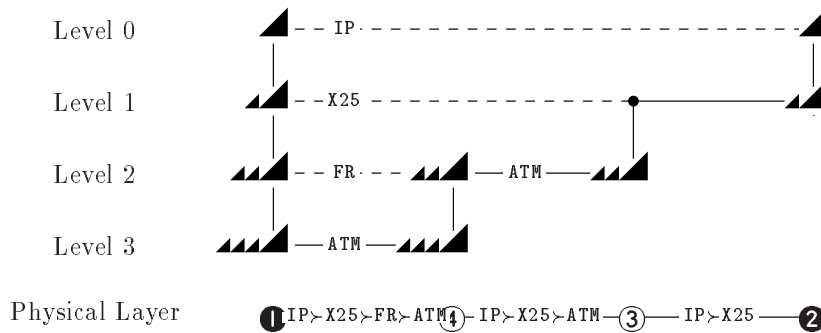


Figure 5: routing a data flow of type IP between nodes 1 and 2 over nested connections

## Contributions

In our view, the contributions of this paper include:

- New and up-to-date formulations of the Routing Problem for flows in networks comprising nested connections with various protocols. With these formulations we take the encapsulation of data flows over many connections into account and introduce the notion of *Non-Conservative Flows*.
- Study of the complexity of the Routing Problem both when the routing of the connections over deeper layers has to be decided as part of the problem (this problem can be seen as a Multi-Layer Routing Problem), and when the routing of the connections over deeper layers is known *a priori*, leaving to the optimiser the decision of routing flows over pre-established nested connections at a minimum cost.
- Development of a column-generation algorithm which takes into account the dependence of reduced edge costs to the paths they belong.

A first study of the problem of routing flows in a heterogeneous context has been carried out by the authors in [BBC<sup>+</sup>99]. In this communication, the networks comprised only one layer and at each node implying a change of protocol a decapsulation was carried out and an encapsulation immediately followed. The problem of interest was the choice of routing paths which had the least impact on the number of encapsulations and decapsulations, no matter the quantity of overhead the encapsulation would generate. The authors proved that finding routings which exhibit the smallest number of protocol changes in such networks can be achieved in polynomial time if the number of protocol changes for any path of the routing is ponderated by the percentage of information carried.

## Outline

The remainder of this paper is organized as follows. In the next Section we give the models of the Routing Problem that we wish to study. In Section 3 we examine the complexity of the Flow Problem which lies at the core of the Routing Problem. We then give a column-generation model of the Routing Problem and provide a greedy algorithm to identify new columns in Section 4. We finally provide an insight into further developments in Section 5.

## 2 Routing Problem formulation

We model a network  $G$  by a collection of undirected capacitated multigraphs:  $G = \{G^\ell, \ell = 0..N\}$ . Each graph  $G^\ell = (V^\ell, E^\ell)$  comprises vertice set  $V^\ell$  and edge set  $E^\ell$ . Let graph  $G^0$  represent the demands; each edge  $e \in E^0$  has protocol  $p_e$  and quantity of information  $d_e$  to be routed between its extremities. For  $\ell = 1..N$ , let edge set  $E^\ell$  represent the set of connections available at layer  $\ell$ ; each edge  $e \in E^\ell$  represents a connection of protocol  $p_e$  and available bandwidth  $C_e$ . In some particular cases, edge  $e$  can represent a “ghost connection”, with no specified protocol ( $p_e = \text{NUL}$ ) and unlimited bandwidth ( $C_e = \infty$ ). These particular edges are represented in Figures 3 and 4 by dotted lines.

We now model the encapsulation phenomenon. Each Protocol Data Unit  $D$  is composed of a Header  $H$  and a Body  $B$ . The size of the header is part of the definition of the protocol. When a connection with protocol  $p$  is supported by a connection with protocol  $q$ , the body  $B_q$

is composed of the whole Protocol Data Unit of protocol  $p$ . Let  $p \succ q$  denote the encapsulation of protocol  $p$  into protocol  $q$ . This phenomenon can be rendered by the following set of operations:

$$B_q = H_p + B_p \quad (1)$$

$$D_q = H_q + B_q \quad (2)$$

$$D_q = H_q + H_p + B_p \quad (3)$$

Let  $\alpha_{p \succ q}$  be the increase in the quantity of information generated by the encapsulation. The value of this multiplier is immediately given by:

$$D_q = \alpha_{p \succ q} D_p \quad (4)$$

$$\alpha_{p \succ q} = \frac{H_q + D_p}{D_q} \quad (5)$$

The multipliers  $\alpha$  obey the following rules:

- If protocols  $p$  and  $q$  are identical, we set  $\alpha_{p \succ q} = 1$ .
- If  $q = \text{NUL}$  then  $\alpha_{p \succ q} = 1$  for any protocol  $p$ .
- The multiplier for cascading encapsulations of protocols  $p$ ,  $q$  and  $r$  is  $\alpha_{p \succ q \succ r} = \alpha_{p \succ q} \alpha_{q \succ r}$ .

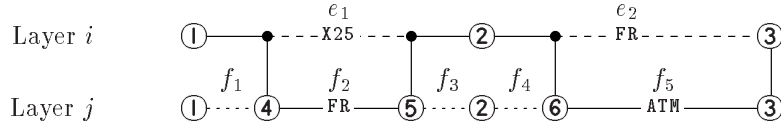


Figure 6: sample connections of layer  $i$  supported by some connections of layer  $j$

Consider any two nodes  $s$  and  $t$  in vertex set  $V^0$ . For any layer  $\ell$ , let  $\Pi_{st}^\ell$  be the set of  $st$ -paths in graph  $G^\ell$ . Consider two layers  $i$  and  $j$  such that  $0 \leq i < j \leq N$ . Notice that because of the nesting of the connections, we have  $V^i \subseteq V^j$ . Let  $\pi$  be a path in  $\Pi_{st}^i$  and  $\rho$  be a path in  $\Pi_{st}^j$ . We say that  $\rho \in \Pi_\pi^j$  if all nodes in path  $\pi$  are contained in path  $\rho$  and the connections of path  $\rho$  support the connections of path  $\pi$ .

We denote by  $\phi_\pi$  the fraction of demand  $d_{st}$  which is carried on path  $\pi$ . Each edge  $e$  in path  $\pi$  carries a flow of type  $\tau_{\pi e}$ . The amount of flow carried on this edge is denoted by  $\psi_{\pi e}$ .

Because the connections composing path  $\pi$  are supported by the connections of path  $\rho$ , there exists a subpath  $\rho'$  of path  $\rho$  corresponding to each edge  $e$  of path  $\pi$ . The type of flow circulating on each edge  $f$  of subpath  $\rho'$  is given by

$$\tau_{\rho' f} = \tau_{\rho f} = \tau_{\pi e} \succ p f \quad (6)$$

If  $\phi_\rho$  is the fraction of demand  $d_{st}$  carried on path  $\rho$ , the amount of flow circulating on each edge  $f$  of subpath  $\rho'$  is given by

$$\psi_{\rho' f} = \psi_{\rho f} = \alpha_{\tau_{\rho f}} \phi_\rho \quad (7)$$

Let us apply these calculations to the sample connections given in Figure 6. We have

$$\pi = \{e_1, e_2\}, \quad \rho = \{f_1, f_2, f_3, f_4, f_5\} \quad (8)$$

Suppose that the connections of path  $\pi$  have to support 1000 units of flow of type IP while the connections of path  $\rho$  have to support 500 units of flow. Then we obtain

$$\tau_{\pi e_1} = \text{IP} \succ \text{X25}, \quad \psi_{\pi e_1} = \alpha_{\text{IP} \succ \text{X25}} \times 1000 \quad (9)$$

$$\tau_{\rho f_1} = \text{IP} \succ \text{X25} \succ \text{NUL} = \text{IP} \succ \text{X25}, \quad \psi_{\rho f_1} = \alpha_{\text{IP} \succ \text{X25}} \times 500 \quad (10)$$

$$\tau_{\rho f_2} = \text{IP} \succ \text{X25} \succ \text{FR}, \quad \psi_{\rho f_2} = \alpha_{\text{IP} \succ \text{X25} \succ \text{FR}} \times 500 \quad (11)$$

We are now ready to give the formulation of the Multilayer Routing Problem ( $\mathbf{R}_N$ ). The networks we consider are composed of one Demand Layer  $G^0$  and  $N$  Support Layers  $G^\ell$ ,  $\ell = 1..N$ . For any layer  $\ell$ , each edge  $e \in E^\ell$  has unit flow cost  $\beta_e$ . For any layer  $\ell$  and edge  $e \in E^\ell$ , we denote by  $\Pi_{ste}^\ell$  the set of  $st$ -paths in layer  $\ell$  which contain edge  $e$ :  $\Pi_{ste}^\ell = \{\pi \in \Pi_{st}^\ell \mid e \in \pi\}$ .

$$(\mathbf{R}_N) \left\{ \begin{array}{l} \min \sum_{st \in E^0} \sum_{\pi \in \Pi_{st}^N} \sum_{e \in \pi} \beta_e \alpha_{\tau_{\pi e}} \phi_\pi \\ \text{subject to} \\ (i) \quad \sum_{\pi \in \Pi_{st}^1} \phi_\pi = d_{st} \quad \forall st \in E^0 \\ (ii) \quad \sum_{\rho \in \Pi_\pi^\ell} \phi_\rho = \phi_\pi \quad \forall st \in E^0, \forall \ell = 2..N, \forall \pi \in \Pi_{st}^{(\ell-1)} \\ (iii) \quad \sum_{st \in E^0} \sum_{\pi \in \Pi_{ste}^\ell} \alpha_{\tau_{\pi e}} \phi_\pi \leq C_e \quad \forall \ell = 1..N, \forall e \in E^\ell \\ (iv) \quad \phi_\pi \geq 0 \quad \forall st \in E^0, \forall \ell = 1..N, \forall \pi \in \Pi_{st}^\ell \end{array} \right.$$

Constraint (i) requires that all demands be satisfied on the first layer, constraint (ii) requires that any fraction of the demand routed over any layer be satisfied on the next supporting layer. Constraint (iii) requires that for any layer, the total amount of flow circulating on any edge be less than the capacity of this edge. Constraint (iv) requires that all flow variables be non-negative. The objective function requires that the flow circulating on layer  $N$  be routed at a minimum cost.

It is clear that for any layer  $\ell$  and for any nodes  $s$  and  $t$  in  $V^0$ , the number of  $st$ -paths in  $G^\ell$  can not be expressed as a polynomial of the number of nodes or edges of the network. Therefore, problem ( $\mathbf{R}_N$ ) exhibits a non-polynomial number of variables.

### 3 Complexity issues

For the sake of simplicity, let us examine the problem of routing only one commodity between nodes  $s$  and  $t$  with requested bandwidth  $d_{st}$  and protocol  $p_{st}$  over  $N$  layers of connections.

The Routing Problem can now be seen as a Minimum Cost Flow Problem in the special context of heterogeneous networks. This problem will be referred to as the *Minimum Cost Non-Conservative Flow Problem* and denoted  $(F_N)$ .

**Lemma 3.1** *The Minimum Cost Non-Conservative Flow Problem is NP-hard.*

Proof: For any layer  $\ell$  and any edge  $e \in E^\ell$ , a fixed size header  $\omega_e^\ell$  is added to the data flow which crosses edge  $e$ . Let  $\pi$  be a path in  $\Pi_{st}^\ell$  such that the data flow it carries is strictly positive. Let  $x_{\pi e}$  equal 1 if edge  $e$  is in path  $\pi$  and 0 otherwise, thus

$$\alpha_{\tau_{\pi e}} \phi_\pi = \phi_\pi + \sum_{i=1}^{\ell} \omega_e^i x_{\pi e} \quad (12)$$

Problem  $(F_N)$  can be written as below:

$$(F_N) \left\{ \begin{array}{l} \min \sum_{e \in E^N} \beta_e \sum_{\pi \in \Pi_{st}^N} \left( \phi_\pi + \sum_{i=1}^N \omega_e^i x_{\pi e} \right) \\ \text{subject to} \\ (i) \quad \sum_{\pi \in \Pi_{st}^1} \phi_\pi = d_{st} \\ (ii) \quad \sum_{\rho \in \Pi_\pi^\ell} \phi_\rho = \phi_\pi \quad \forall \ell = 2..N, \forall \pi \in \Pi_{st}^{(\ell-1)} \\ (iii) \quad \sum_{\pi \in \Pi_{ste}^\ell} \left( \phi_\pi + \sum_{i=1}^{\ell} \omega_e^i x_{\pi e} \right) \leq C_e \quad \forall \ell = 1..N, \forall e \in E^\ell \\ (iv) \quad \phi_\pi \geq 0 \quad \forall \ell = 1..N, \forall \pi \in \Pi_{st}^\ell \\ (v) \quad x_{\pi e} \geq \frac{1}{d_{st}} \phi_\pi \quad \forall \ell = 1..N, \forall \pi \in \Pi_{st}^\ell, \forall e \in \pi \\ (vi) \quad x_{\pi e} \in \{0, 1\} \quad \forall \ell = 1..N, \forall \pi \in \Pi_{st}^\ell, \forall e \in \pi \end{array} \right.$$

If we consider that demand  $d_{st}$  is extremely small, we can polynomially reduce the Minimum Edge-Cost Flow Problem to  $(F_N)$ . Since the decision problem associated to the Minimum Edge-Cost Flow Problem is known to be *NP*-complete (see [GJ79], problem ND32), we may conclude to the *NP*-hardness of  $(F_N)$ .  $\blacksquare$

We now investigate the complexity of the Column-Generation Problem for  $(F_N)$ .

**Corollary 3.1** *The Column-Generation Problem for the Minimum Cost Non-Conservative Flow Problem is NP-complete.*

Proof: Consider the Dual Problem (D) of the Minimum Cost Non-Conservative Flow Problem. Problem (D) is also *NP*-hard, otherwise  $(F_N)$  could be solved in polynomial time.

Since problem (D) has a polynomial number of variables and an exponentially large number of constraints, by the Separation-Optimisation Theorem stated in [GLS81], we may affirm that the separation problem for (D) is *NP*-complete. Since generating a constraint in the dual is the same problem as generating a column in the primal, we are done. ■

According to the *NP*-hardness of the Minimum Cost Non-Conservative Flow Problem we can conclude that problem (R<sub>N</sub>) is *NP*-hard.

### Note on Generalized Flows

We draw the reader's attention to the fact that problem (F<sub>N</sub>) can not, in most cases, be seen as a member of the family of Generalized Minimum Cost Flow Problems (see [AMO93] for a description of these problems). For such flow problems a positive multiplier  $\alpha_e$  is associated with each edge  $e$ , and there exist polynomial-time algorithms to find minimum cost flows in networks with such gains, based on general linear programming techniques. Recently, Wayne developed the first polynomial combinatorial algorithm for these problems (see [Way99]). Problem (F<sub>N</sub>) could nonetheless be seen as a Generalized Minimum Cost Flow Problem if for each demand  $st$  and for each  $st$ -path  $\pi$ , the multiplier  $\alpha_{\tau_{\pi e}}$  associated with any edge  $e$  was to be equal to a fixed value  $\alpha_e$ .

## 4 Column-generation model of the Routing Problem

The general idea of column-generation is that optimal solutions to large linear programs can be obtained without explicitly including all variables (i.e. columns) in the constraint matrix. In fact, only a small subset of all columns will be in an optimal solution, and all other columns can be ignored. In a minimisation problem, this implies that all columns with positive reduced cost need not be computed.

In order to identify the columns of program (R<sub>N</sub>) which need to be computed, we need the following dual variables. For all  $st \in E^0$ , let  $\lambda_{st}$  be the unrestricted dual variables associated with constraints (i). For all  $st \in E^0$ , for all  $\ell = 2..N$  and for all  $\pi \in \Pi_{st}^{(\ell-1)}$ , let  $\nu_{\pi}^{\ell}$  be the unrestricted dual variables associated with constraints (ii). Finally, for all  $\ell = 1..N$  and for all  $e \in E^{\ell}$ , let  $\mu_e^{\ell}$  be the nonnegative dual variables associated with constraints (iii).

For any demand  $st \in E^0$ , a column  $\phi_{st}$  of linear program (R<sub>N</sub>) comprises  $N$  flow variables defined over  $N$  different  $st$ -paths  $\pi_{\ell}$ , each path  $\pi_{\ell}$  belonging to graph  $G^{\ell}$ :  $\phi_{st} = (\phi_{\pi_1}, \phi_{\pi_2}, \dots, \phi_{\pi_N})$ . An example of such a column is given in Figure 7.

The reduced cost of any column is given by:

$$\bar{c}_{\phi_{st}} = -\lambda_{st} + \nu_{\pi_1}^2 - \nu_{\pi_1}^2 + \nu_{\pi_2}^3 \dots + \nu_{\pi_{N-1}}^N - \nu_{\pi_{N-1}}^N + \sum_{\ell=1}^N \sum_{e \in \pi_{\ell}} \alpha_{\tau_{\pi_{\ell} e}} \mu_e^{\ell} + \sum_{e \in \pi_N} \alpha_{\tau_{\pi_{\ell} e}} \beta_e \quad (13)$$

$$\bar{c}_{\phi_{st}} = -\lambda_{st} + \sum_{\ell=1}^N \sum_{e \in \pi_{\ell}} \alpha_{\tau_{\pi_{\ell} e}} \mu_e^{\ell} + \sum_{e \in \pi_N} \alpha_{\tau_{\pi_{\ell} e}} \beta_e \quad (14)$$

Before we can identify columns that price out, we need to overcome the difficulty posed by the dependency of edge costs to the paths they belong. Indeed, we wish to find new columns of program (R<sub>N</sub>) by solving one or many shortest paths problems in graphs whose edge costs are conveniently chosen, as is usually done in the field of column-generation algorithms. But

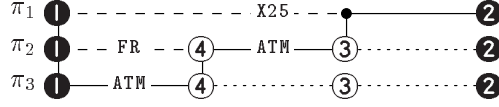


Figure 7: structure of a sample column of program  $(R_N)$

first we have the following interesting lemma for the special case where the routing of each layer of connections over the next layer is known in advance.

**Lemma 4.1** *If the routing of each layer of connections over the next layer is known, the Minimum Cost Non-Conservative Flow Problem can be solved in a polynomial time of the size of the data.*

Proof: In this case, all the columns of program  $(R_N)$  can be obtained in polynomial time as fully developed branches of “supporting trees” for each demand (see Figure 8). Problem  $(R_N)$  can then be rewritten as a much simpler linear program and thus be solved in polynomial time.  $\blacksquare$

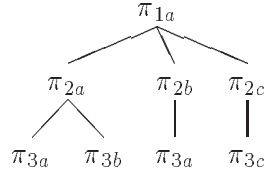


Figure 8: sample supporting tree for path  $\pi_{1a}$  of layer 1

## Greedy algorithm for column-generation

In the remainder of this section we give a way of finding new columns of program  $(R_N)$  as part of a usual column-generation process. At this point we know the values of the dual variables  $\lambda_{st}$  for each demand  $st$ , as well as the values of the dual variables  $\mu_e^\ell$  for each layer  $\ell$  and for each edge  $e \in E^\ell$ .

For each demand  $st \in E^0$ :

- Let  $\pi_0 = \{(s, t)\}$  be the  $st$ -path composed of the only one edge between nodes  $s$  and  $t$  in graph  $G^0$ .
- Let  $L_{st}$  be the length associated with the column we are building, and set  $L_{st} = 0$ .
- For each layer  $\ell = 1..N$ :
  - Set  $\pi_\ell = \{\emptyset\}$ .
  - For each edge  $e = (u, v) \in \pi_{\ell-1}$ :
    - Let  $\tau_e$  be the type of flow which circulates on edge  $e$ .
    - For each edge  $f \in G^\ell$ :
      - Let  $\alpha_f = \alpha_{\tau_e \succ p_f}$  be the multiplier associated with edge  $f$ .

- If  $f$  is a ghost connection then update its cost  $\beta_f$ .
  - If  $\ell < N$  set edge weight  $w_f = \alpha_f \mu_f^\ell$ , else set edge weight  $w_f = \alpha_f (\mu_f^\ell + \beta_f)$ .
  - Let  $\pi_{uv}$  be the shortest  $uv$ -path in graph  $G^\ell$  whose edges are ponderated by weights  $w$ . Let  $L_{uv}$  be the length of path  $\pi_{uv}$ .
  - Do  $L_{st} = L_{st} + L_{uv}$ .
  - If  $L_{st} \geq \lambda_{st}$  then STOP, else concatenate path  $\pi_{uv}$  to path  $\pi_\ell$ .
- If  $L_{st} < \lambda_{st}$  then column  $\{\pi_1, \dots, \pi_N\}$  must be included in program ( $R_N$ ).

## 5 Further developments

Some numerical results are currently under way, and we should be able to examine the effect of edge multipliers on the cost of routings in heterogeneous networks. For the time being our column-generation procedure has been tested on small networks, achieving satisfactory results. Yet we still have to find column-generation procedures which exploit the combinatorial structure of our problem. Our formulation of the Routing Problem will serve as a good starting point for the formulation of a Capacitated Network Design Problem with heterogeneous protocols and path-dependent edge costs.

## References

- [AMO93] Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. *Network Flows: Theory, Algorithms and Applications*. Prentice Hall, Englewood Cliffs, New Jersey, February 1993.
- [BBC<sup>+</sup>99] Sébastien Bertrand, Florence Boyer, Pascal Chanas, Alexandre Laugier, and Jean-Pierre Paris. Routing Heterogeneous Traffic on Heterogeneous Networks. In *Proceedings of the Twelfth European Chapter on Combinatorial Optimization (ECCO XII)*, Island of Bendor, France, May 27-29 1999.
- [GJ79] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co, 1979.
- [GLS81] Martin Grötschel, László Lovász, and Alexander Schrijver. The Ellipsoid Method and its Consequences in Combinatorial Optimization. *Combinatorica*, 1:169–197, 1981.
- [LM00] Alexandre Laugier and Philippe Mahey. Design of Multi-Level-Protocol Data Networks. In *Actes des deuxièmes rencontres francophones sur les aspects algorithmiques des télécommunications (AlgoTel2000)*, La Rochelle, France, May 10-12 2000.
- [Way99] Kevin D. Wayne. A Polynomial Combinatorial Algorithm for Generalized Minimum Cost Flow. In *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing (STOC'99)*, Atlanta, Georgia, May 1-4 1999.