



HAL
open science

Synthèse de réseaux à composantes connexes unicycliques

Makhlouf Hadji

► **To cite this version:**

Makhlouf Hadji. Synthèse de réseaux à composantes connexes unicycliques. Réseaux et télécommunications [cs.NI]. Université Pierre et Marie Curie - Paris VI, 2009. Français. NNT: . tel-00459860

HAL Id: tel-00459860

<https://theses.hal.science/tel-00459860>

Submitted on 25 Feb 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.

**Thèse de Doctorat de Télécom & Management SudParis
conjointement à l'université Paris VI
Pierre et Marie Curie**

Spécialité
INFORMATIQUE

présentée par
Makhlouf Hadji

pour obtenir le grade de
**Docteur de Télécom & Management SudParis conjointement
à l'université Pierre et Marie Curie**

**Synthèse de réseaux à composantes connexes
unicycliques**
(version préliminaire)

Soutenance prévue le ... Septembre 2009 devant le jury composé de

Mohamed DIDI-BIHA	Rapporteur	HDR. à l'Université d'Avignon
Dritan NACE	Rapporteur	Prof. à l'Université de Technologie de Compiègne
Philippe CHRETIENNE	Examineur	Prof. au LIP6, Université Pierre et Marie Curie - Paris VI
Léo LIBERTI	Examineur	HDR. au LIX, École Polytechnique
Adam OUOROU	Examineur	HDR. à France Télécom/Orange Labs
Walid BEN-AMEUR	Directeur	Prof. à l'Institut Télécom et Management SudParis

À la mémoire de ma mère

Résumé

Cette thèse s'inscrit dans le domaine de l'optimisation combinatoire. Elle utilise l'approche polyédrale pour résoudre des problèmes combinatoires qui se posent dans le contexte des réseaux de télécommunications.

Nous introduisons et étudions le problème de synthèse de réseaux à composantes connexes unicycliques.

Après avoir rappelé que le problème est facile à résoudre en absence d'autres contraintes, nous étudions de nouvelles variantes en intégrant de nouvelles contraintes techniques.

Nous commençons par une contrainte portant sur la taille des cycles. Nous souhaitons interdire tous les cycles contenant au plus p sommets. Le problème est alors NP-Difficile. Des inégalités valides sont alors proposées pour ce problème. On montre sous des conditions bien précises que ces inégalités peuvent être des facettes. Plusieurs algorithmes polynomiaux ont été proposés pour la séparation des inégalités valides. Ces algorithmes sont mis en oeuvre et des résultats numériques sont donnés.

Nous nous focalisons par la suite sur un nouveau problème dit de Steiner consistant à partitionner un réseau en composantes unicycliques tout en imposant que certains sommets soient sur les cycles. On montre alors que ce problème est facile au sens de la complexité algorithmique en proposant un algorithme polynomial et une formulation étendue du problème. On présente également une description partielle de l'enveloppe convexe des vecteurs d'incidence de ces réseaux. La séparation des inégalités est également étudiée. Nous proposons notamment une généralisation de l'algorithme de Padberg-Rao pour séparer les inégalités Blossom.

D'autres contraintes techniques sont prises en compte : contraintes de degrés, contrainte sur le nombre de composantes connexes, appartenance de certains sommets à une même composante connexe et enfin la séparation de certains sommets qui doivent être sur des composantes différentes.

Enfin, nous faisons une étude spectrale de deux classes spécifiques de graphes unicycliques.

Mots clés : Optimisation Combinatoire, Polyèdres, Algorithme à Plans Coupants, Graphes Unicycliques, Matroïdes, Spectre.

Abstract

In this thesis, we use the polyhedral approach to solve combinatorial problems in telecommunications context.

First, we introduce the problem of network design with unicyclic connected components.

We recall that without other constraints, our problem is easy to solve, and we propose a study with new technical constraints.

We start our study by adding constraints on the size of cycles. We aim to obtain unicyclic components such that the size of each cycle is not lower than a certain p . This problem is NP-Hard. We describe some valid inequalities for the design of unicyclic graphs with girth constraints. The faces induced by these valid inequalities are also studied. Some of them can be separated in polynomial time. A cutting plane algorithm based on these inequalities is implemented to solve the problem.

Furthermore, we focus on a Steiner type problem, which consists in partitioning the graph to unicyclic components, such that some given vertices belong to a cycle. We prove then that our problem is easy to solve, and we propose an exact extended formulation and a partial description of the convex hull of the incidence vectors of our Steiner network problem. Polynomial time separation algorithms are described. One of them is a generalization of the Padberg&Rao algorithm to separate blossom inequalities.

Other technical constraints are proposed such as degree constraints, a bound of the number of unicyclic components, constraints related to whether some given pairs of vertices belong to the same component or to different components. Finally, we study the spectra of two specified classes of unicyclic graphs.

Keywords : Combinatorial Optimization, Polyhedral Study, Cutting Plane Algorithm, Unicyclic Graphs, Matroids, Spectra.

Table des matières

Introduction Générale	1
1 Notions préliminaires	3
1.1 Optimisation des réseaux	3
1.2 Approche polyédrale	4
1.2.1 Introduction	4
1.2.2 Les ensembles et combinaisons convexes	4
1.2.3 Programmation Linéaire et Optimisation Combinatoire	5
1.2.4 Polyèdres, Faces et Facettes	6
1.2.5 Méthode de Coupes	7
1.2.6 Méthode de Branch&Cut	7
1.2.7 Séparation et Optimisation	8
1.3 Problèmes combinatoires importants	8
1.3.1 Les problèmes de plus court chemin	8
1.3.1.1 L'algorithme de Dijkstra	9
1.3.1.2 L'algorithme de Bellman	9
1.3.1.3 L'algorithme de Bellman-Ford	10
1.3.2 Problème de l'arbre couvrant de poids minimum	10
1.3.2.1 Algorithme de Kruskal	10
1.3.2.2 Algorithme de Prim	10
1.3.3 Le problème de flot maximum-coupe minimum	11
1.3.3.1 La méthode de Ford-Fulckerson	11
1.3.3.2 La Méthode des préflots- Algorithme de Goldberg-Tarjan	12
1.3.3.3 L'algorithme de Gomory-Hu	13
1.3.4 Les T-join et T-coupe	14
1.3.4.1 Le problème de T-coupe de poids minimum	14
1.3.4.2 Le problème du T-join de poids minimum	14
1.3.5 Le problème de couplage	15
1.3.5.1 Le problème de b-couplage	15
1.3.6 Le problème du voyageur de commerce	16
1.4 Réseaux à composantes unicycliques	18
1.4.1 Les matroïdes	18
1.4.1.1 Les matroïdes bi-circulaires	18
1.4.2 Partitionnement des réseaux en composantes unicycliques	19

1.4.2.1	L'algorithme glouton	19
1.4.2.2	L'algorithme de couplage maximum de poids minimum	19
1.5	Spectre des graphes	20
1.5.1	Quelques liens avec l'optimisation combinatoire	21
1.5.1.1	Lien avec le diamètre d'un graphe	21
1.5.1.2	Lien avec le nombre isopérimétrique	22
1.5.1.3	Lien avec le problème de coupe maximum (Max-Cut)	22
2	Réseaux à composantes unicycliques avec contrainte sur la taille des cycles	23
2.1	Introduction	23
2.2	Notations et Formulation Mathématique	24
2.3	Inégalités Valides	26
2.4	Sur la structure faciale de $P(G, p)$	30
2.5	Un Algorithme à Plans Coupants	40
2.5.1	Séparation des inégalités (2.5)	40
2.5.2	Séparation des inégalités (2.2)	42
2.5.3	Séparation des inégalités (2.4)	45
2.5.4	Séparation des inégalités (2.6)	46
2.5.5	Séparation des inégalités (2.7), (2.8) et (2.9)	47
2.6	Résultats Expérimentaux	48
2.7	Conclusion	50
3	Réseaux à composantes unicycliques avec contraintes de type Steiner	53
3.1	Introduction	53
3.2	Une première formulation mathématique	54
3.3	Complexité et formulation étendue	56
3.4	Les inégalités valides	58
3.5	Les algorithmes de séparation	60
3.5.1	Séparation des contraintes (3.3)	60
3.5.2	Séparation des contraintes (3.7)	61
3.6	Extensions et liens avec d'autres problèmes	65
3.7	Résultats numériques	66
3.8	Conclusion	67
4	Réseaux à composantes unicycliques avec de nouvelles contraintes techniques	69
4.1	Introduction	69
4.2	Contraintes portant sur les degrés des sommets	70
4.3	Contrainte portant sur le nombre de composantes unicycliques	70
4.4	Séparation de certains couples de sommets	72
4.5	Joindre certains couples de sommets	72
4.6	Conclusion et perspectives	75
5	Spectre de quelques graphes unicycliques	77
5.1	Introduction	77
5.2	Rappels sur les bornes des valeurs propres extrêmes des matrices A et L	78
5.2.1	Bornes de λ_1 et de λ_n	78
5.2.2	Bornes de μ_1 et de μ_{n-1}	79

5.3	Spectre des graphes unicycliques $G(q, n_1, 0)$	79
5.4	Spectre des graphes unicycliques $G(q, n_1, n_2)$	87
5.5	Conclusion	98
Conclusion générale et perspectives		100
Bibliographie		102
Liste des figures		113
Liste des tables		115

Introduction Générale

Cette thèse s'inscrit dans le domaine de l'optimisation combinatoire. Elle a pour objet d'étudier la synthèse de réseaux à composantes connexes unicycliques. Elle a été financée par un contrat de recherche avec les laboratoires d'Orange.

L'optimisation combinatoire est la science qui consiste à trouver un élément optimal au sens d'un coût donné parmi un ensemble fini d'éléments. L'approche polyédrale est l'une des techniques permettant de résoudre des problèmes d'optimisation combinatoire. Elle est particulièrement utile pour résoudre des problèmes NP-difficiles. Chaque solution réalisable est généralement représentée par un vecteur. L'idée fondamentale des méthodes polyédrales est d'approcher au mieux l'enveloppe convexe de ces vecteurs et d'optimiser sur cette enveloppe approchée en utilisant la programmation linéaire.

Les travaux de cette thèse portent globalement sur des problèmes qui se posent dans le cadre des réseaux de télécommunications. En effet, il s'agit de construire un réseau dont toutes les composantes connexes sont unicycliques. On rappelle qu'un graphe est dit unicyclique si et seulement si il est connexe et contient un seul cycle. La structure du cycle permet d'assurer une certaine sûreté de fonctionnement.

Nous détaillons dans la suite l'ensemble des travaux présentés dans le manuscrit.

Nous introduisons dans le premier chapitre les concepts de base et nous exposons les outils mathématiques que nous utiliserons dans la suite. Nous présentons également deux algorithmes pour résoudre le problème de synthèse de réseaux à composantes connexes unicycliques. En effet, si on ne considère que la contrainte d'unicyclicité, le problème est facile à résoudre.

Dans le chapitre 2, on introduit une contrainte supplémentaire portant sur la taille des cycles : tous les cycles doivent être de taille supérieure ou égale à $p + 1$. Nous faisons une étude polyédrale du problème d'optimisation sous-jacent. Un algorithme à plans coupants suivi d'un Branch&Cut est proposé. On termine par donner des résultats numériques en considérant plusieurs classes de graphes et des instances de tailles différentes.

Le chapitre 3 se consacre à l'étude du problème de partitionnement d'un réseau en composantes unicycliques en imposant que certains sommets appartiennent à l'un des cycles. Ce problème est en effet une généralisation du problème de 2-couplage. Ce problème s'est avéré facile à résoudre, et une formulation étendue sera alors proposée. Par la suite et dans le but de caractériser l'enveloppe convexe de ce problème, une étude polyédrale est alors proposée en mettant en oeuvre de nouvelles inégalités valides caractérisant partiellement l'enveloppe

convexe. Des algorithmes polynomiaux ont été proposés pour la séparation de ces inégalités valides.

L'objet du chapitre 4 est d'intégrer de nouvelles contraintes techniques qui répondent à différents besoins. Quatre contraintes ont été proposées. La première concerne les degrés des sommets. Le deuxième type de contraintes consiste à considérer une borne supérieure sur le nombre de composantes unicycliques à obtenir en fin du partitionnement. Nous intégrons également des contraintes qui consistent à séparer un ou plusieurs couples de sommets sur différentes composantes unicycliques. Enfin, on peut exiger dans certains cas que certains sommets appartiennent à une même composante connexe.

Le chapitre 5 fait l'objet d'une étude de deux classes spécifiques de graphes unicycliques, pour lesquels on donne des résultats sur les valeurs propres extrêmes des matrices d'adjacence et du Laplacien de ces graphes. Outre l'intérêt théorique de ces résultats, ils peuvent être utiles dans le contexte de synthèse de réseaux appartenant à ces classes de graphes.

Notions préliminaires

1.1 Optimisation des réseaux

Nous traitons dans cette thèse plusieurs problèmes d'optimisation combinatoire qui se posent dans le cadre de l'optimisation des réseaux. Il convient donc de décrire brièvement l'optimisation des réseaux.

En réalité la tâche est ardue pour plusieurs raisons. En effet, d'une part les réseaux sont multiples (plusieurs technologies, plusieurs architectures). D'autre part, les problèmes sont différents. Enfin, les outils mathématiques qui sont utilisés sont également très divers.

Pour ne pas nous perdre dans les détails, nous allons nous contenter de présenter très brièvement les types de problèmes qu'on cherche généralement à résoudre pour optimiser des réseaux.

Un réseau a une structure. Le problème de base consiste à définir la structure ou la topologie du réseau. Ceci revient à imposer des contraintes de connexité, voire de k -connexité. On peut également considérer des contraintes de diamètre pour imposer que les communications se fassent en un temps limité (voir [1], [5], [34], [35], [55], [77]). D'autres problèmes de topologie consistent à localiser des équipements d'un type particulier (concentrateurs, noeuds d'accès, etc) [81], [126]. Construire un réseau sous la forme d'un ensemble de réseaux d'accès connectés par une dorsale est un autre exemple de problème de topologie : comment faire ce découpage d'une manière optimale ?. Construire donc la topologie d'un réseau donne lieu à un ensemble de problèmes combinatoires très divers et souvent difficiles à résoudre. Tout devient plus complexe si on intègre un aspect dynamique de configuration des réseaux : des noeuds qui se déplacent ou qui changent de comportement induisant un changement de la topologie. Ré-optimiser la topologie en respectant un certain nombre de contraintes techniques est un challenge d'actualité.

Une fois la topologie fixée, il faut relayer du trafic dans le réseau. En d'autres termes, il faut résoudre des problèmes de routage. Ceci se ramène souvent à des problèmes de flot dans un graphe. Plusieurs types de contraintes peuvent être prises en compte : routage de chaque demande sur un seul chemin [7], routage sur plus court chemin [8], [41], routage avec contraintes de délai [6], routage avec contrainte d'équité [105]. On peut également considérer des contraintes de sécurisation se traduisant par l'obligation de rerouter le trafic en cas de panne de certaines composantes du réseau [10], [91], [99]. Tous ces problèmes de routage peuvent

aussi être étudiés dans un contexte incertain, c'est à dire lorsque la matrice de trafic qu'on cherche à écouler dans le réseau n'est pas définie d'une manière précise. On peut par exemple supposer que la matrice de trafic varie dans un polytope et chercher un routage compatible avec toutes ces matrices [9], [107], [114].

Pour router du trafic sur une topologie déterminée, il faut installer des capacités de transmission. Ces capacités sont souvent choisies parmi un ensemble discret. L'optimisation du coût des capacités installées se ramène souvent à un problème de programmation linéaire en nombres entiers. Ces problèmes sont généralement appelés problèmes de dimensionnement. Ils sont en réalité souvent résolus en même temps que les problèmes de routage.

Un autre type de problèmes d'optimisation qui se posent dans le contexte des réseaux est le problème de tarification. En effet, un réseau offre des services qui doivent être facturés aux clients. Trouver le bon mode de tarification qui garantit par exemple une certaine équité ou un certain niveau de bénéfice à l'opérateur, se ramène à des problèmes d'optimisation et de théorie des jeux [16].

Les types de problèmes qu'on vient d'exposer peuvent être étudiés séparément ou d'une manière combinée. L'utilisation d'une technologie donnée (Internet, ATM, GSM, UMTS, SDH, etc.) induit des contraintes techniques particulières et donc des problèmes d'optimisation particuliers.

Les problèmes d'optimisation de réseaux que nous avons étudiés dans cette thèse sont des problèmes du type topologie.

1.2 Approche polyédrale

Dans cette section on donne des notions générales sur la théorie des polyèdres, et pour plus de détails, le lecteur peut se référer à [117], [124].

1.2.1 Introduction

Soit E un ensemble fini tel que $|E| = n$. Soit $(w_e)_{e \in E}$ un vecteur de poids associé aux éléments de E . Soit $\mathbb{F} \subseteq 2^E$ une famille de sous ensembles de E . Si $F \in \mathbb{F}$ alors on note le poids de F par $w(F) = \sum_{e \in F} w(e)$. Le problème qu'on peut poser ainsi, est de chercher un ensemble noté par F^* dans \mathbb{F} de telle sorte que $w(F^*)$ soit optimum. Ce problème est appelé un **problème d'optimisation combinatoire**.

Pour illustrer notre propos, appuyons-nous sur l'exemple du problème de recherche d'un plus court chemin dans un graphe, où E est l'ensemble des arcs d'un graphe et w définit la fonction des poids sur les arcs : $w : E \rightarrow \mathbb{R}^+$, et \mathbb{F} sera l'ensemble de tous les chemins entre l'origine et la destination qu'on souhaite.

D'autres exemples d'optimisation combinatoire peuvent être consultés dans [17], [20].

1.2.2 Les ensembles et combinaisons convexes

Étant donné un graphe $G = (V, E)$, et soit $F \subseteq E$ un sous ensemble d'arêtes. On note par x^E le vecteur en 0 – 1 dit d'incidence tel que $x^F \in \mathbb{R}^E$, et $x^F(e) = 1$ si $e \in F$, sinon 0.

Un ensemble $\mathcal{C} \in \mathbb{R}^n$ est dit **convexe** s'il contient tous les segments en ligne entre leurs deux

extrémités. Autrement dit, $\lambda_1 x_1 + \lambda_2 x_2 \in \mathcal{C}$, pour tout $x_1, x_2 \in \mathcal{C}$, $\lambda_1, \lambda_2 \geq 0$ et que $\lambda_1 + \lambda_2 = 1$. La figure (1.1) représente quelques ensembles convexes et non convexes dans \mathbb{R}^2 .

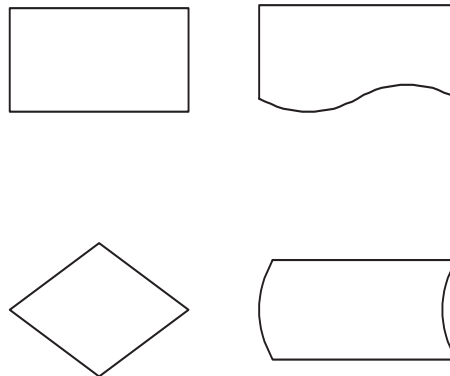


FIGURE 1.1 – Exemple d'ensembles convexes et d'ensembles non convexes

Une *combinaison linéaire* x des vecteurs $x_1, x_2, \dots, x_k \in \mathbb{R}^m$ est exprimée par $\sum_{i=1}^k \lambda_i x_i$. Si de plus on a $\sum_{i=1}^k \lambda_i = 1$ on parlera alors de combinaison *affine* des x_i . Si au même temps on a $\forall i = 1, \dots, k : \lambda_i \geq 0$ alors on a une combinaison *convexe*. On dit alors que des vecteurs sont linéairement indépendants (respectivement. affinement indépendants) si on ne trouve pas de vecteur qui s'exprime en combinaison linéaire (combinaison affine) des autres vecteurs. Si S est un ensemble de vecteurs d'incidence dans \mathbb{R}^m , alors on note par $Conv(S)$ l'*enveloppe convexe* de S qui représente l'ensemble des combinaisons convexes des vecteurs de S .

1.2.3 Programmation Linéaire et Optimisation Combinatoire

Comme on l'a déjà défini ci-dessus, un problème d'optimisation combinatoire est de la forme suivante :

$$POC = \max \{ w(F) = \sum_{e \in F} w(e), F \in \mathbb{F} \} \quad (1.1)$$

Afin de bien pouvoir poser les problèmes à traiter sous forme mathématique, on propose la notation suivante : pour $F \in \mathbb{F}$, on associe un vecteur $x^F \in \{0, 1\}^{|E|}$, appelé *vecteur d'incidence*, et dont les valeurs sont données par

$$x_i^F = \begin{cases} 1, & i \in F; \\ 0, & i \in E \setminus F. \end{cases}$$

En utilisant cette notation, on peut alors formuler un problème d'optimisation combinatoire, en un programme en 0-1. Considérons maintenant l'enveloppe convexe de tous les vecteurs d'incidence. Si on dispose d'une description polyédrale de cette enveloppe convexe, alors maximiser la fonction linéaire $w x$ sur cet ensemble est strictement équivalent à la maximiser sur l'ensemble des vecteurs d'incidence. On sait, en effet, que tout programme linéaire atteint son optimum en au moins un point extrême.

La difficulté de cette approche réside dans la caractérisation polyédrale de l'enveloppe convexe. Notons qu'en pratique, une description partielle pourrait suffir lorsqu'on applique une méthode du type Branch&Cut qu'on décrira dans la suite.

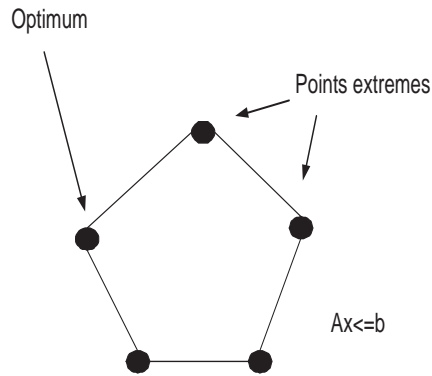


FIGURE 1.2 – Représentation d’un polyèdre

1.2.4 Polyèdres, Faces et Facettes

Étant donné une matrice $A_{m \times n} \in \mathbb{R}^{m \times n}$ et un vecteur $b \in \mathbb{R}^m$. On définit un **polyèdre** P par $\{x \in \mathbb{R}^n; Ax \leq b\}$. P représente alors l’intersection d’un nombre fini de demi-espaces dans \mathbb{R}^n . On dit que le polyèdre P est un **polytope** s’il est borné. La dimension d’un polyèdre P notée par $\dim(P)$ est égale au nombre maximum de vecteurs affinement indépendants moins un. En fonction du rang de la matrice A , on peut d’abord distinguer la notation $A^=$ qui représente une sous matrice de A des inégalités satisfaites à l’égalité par tous les vecteurs de P , alors on peut redéfinir la dimension de P , par $\dim(P) = m - \text{rang}(A^=)$. Si $\dim(P) = n$ alors on dit que le polyèdre est de *pleine dimension*.

On dit qu’une inégalité $ax \leq \alpha$ est *valide* si elle est satisfaite par tous les vecteurs de P . Ainsi, l’ensemble des inégalités valides de P satisfaites à l’égalité est défini par une **facette** F , et tel que $F = \{x \in P; ax = \alpha\}$. La dimension d’une facette F est $\dim(F) = \dim(P) - 1$.

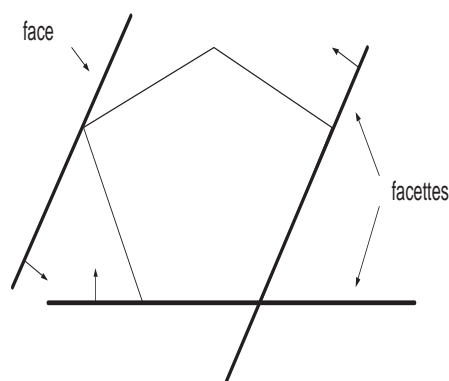


FIGURE 1.3 – Polyèdre , face et facettes

Il est difficile de décrire l’ensemble des facettes de l’enveloppe convexe des solutions réalisables d’un problème combinatoire NP-Difficile. Ceci est une conséquence de l’équivalence

entre la séparation et l'optimisation. Cependant, même si le problème combinatoire est facile à résoudre, il peut être difficile d'énumérer toutes les facettes. En effet, dans certains cas, elles sont en nombre exponentiel. Dans d'autres cas, on ne connaît même pas la forme de ces facettes.

1.2.5 Méthode de Coupes

Pour un problème d'optimisation combinatoire donné, il est en général difficile de caractériser le polyèdre associé par un système d'inégalités linéaires. De plus, même si ce dernier est caractérisé, le système décrivant le polyèdre peut contenir un nombre exponentiel d'inégalités et donc reste inexploitable pour être résolu comme un programme linéaire. Cependant, une description partielle du polyèdre garantie par la *méthode de coupes* peut être suffisante pour résoudre le problème jusqu'à l'optimum.

Soit le problème d'optimisation combinatoire suivant

$$\mathbb{P} = \max\{wx : Ax \leq b, x \text{ entier}\} \quad (1.2)$$

et soit P l'enveloppe convexe des solutions de (1.2). On rappelle que ce problème est équivalent au programme $\max\{wx : x \in P\}$, et que si les inégalités du système $Ax \leq b$ sont suffisantes pour décrire complètement P , alors tous les points extrêmes du polyèdre $\{x \in \mathbb{R}^n : Ax \leq b\}$ sont des entiers, et par conséquent le problème (1.2) est équivalent à la relaxation $\max\{wx : Ax \leq b\}$. Néanmoins, ce n'est pas toujours le cas, et le polyèdre $\{x \in \mathbb{R}^n : Ax \leq b\}$ peut bien avoir des points extrêmes fractionnaires. En d'autres mots, il existe une contrainte valide violée par cette solution optimale fractionnaire. Cette dernière peut être ajoutée au système $Ax \leq b$, afin de trouver une nouvelle relaxation linéaire plus forte.

Dans la méthode de coupes, l'étape de la génération des contraintes valides est la plus difficile. L'une des premières techniques dont la finalité est d'identifier les contraintes valides pour des programmes en variables entières, a été introduite par Gomory. Ensuite, Chvátal a généralisé cette technique, pour obtenir la méthode bien connue de Chvátal-Gomory.

1.2.6 Méthode de Branch&Cut

La méthode qu'on décrit ici, permet de construire un arbre de résolution (arbre de branch&cut), où chaque sommet de l'arbre correspond à un sous-problème, et le problème initial est associé à la racine de l'arbre.

Le principe de cette méthode commence par résoudre initialement la relaxation du problème (1.2). Si la solution trouvée est optimale et contient une variable fractionnaire, qui est sensée être entière, alors un algorithme à plans coupants est appliqué afin de trouver de nouvelles inégalités valides (quelques unes peuvent suffire). Ces dernières sont satisfaites par tous les points entiers, mais violées par la solution courante. Ces inégalités seront ajoutées au programme courant.

Par la suite, on utilise une méthode de branchement qui consiste à obtenir de nouveaux noeuds. On résout chaque programme associé à un noeud en y ajoutant un certain ensemble de coupes

qui peuvent être valides localement au niveau du noeud courant ainsi que du sous-arbre issu de lui, ou globalement au niveau de tous les noeuds de l'arbre de branch&cut.

Lors de la résolution d'un programme associé à un noeud de l'arbre, on calcule une évaluation. Si cette évaluation est supérieure (pour un problème de minimisation) à la borne supérieure calculée avant, alors on coupe ce noeud. Dans le cas contraire, il est exploité à son tour.

On arrête cet algorithme lorsqu'on ne peut plus exploiter un noeud de l'arbre. On prend alors la meilleure solution trouvée.

Remarque : Les coupes qu'on ajoute aux programmes associés aux noeuds de l'arbre de branch&cut peuvent être générées jusqu'à épuisement.

1.2.7 Séparation et Optimisation

Considérons un problème d'optimisation combinatoire du type

$$\mathbb{P} = \max\{wx : Ax \leq b, x \text{ entier}\} \quad (1.3)$$

Le problème de séparation qui lui est associé est défini comme suit :

Pour une solution donnée x^ , vérifier si x^* appartient à l'enveloppe convexe des solutions entières du système $Ax \leq b$, et sinon déterminer une inégalité valide pour cette enveloppe convexe qui soit violée par x^* .*

Ce problème est appelé **problème de séparation** associé à l'enveloppe convexe des solutions entières du système $Ax \leq b$.

Si x^* ne vérifie pas le système $Ax \leq b$ alors il existe un hyperplan séparant x^* et le polyèdre $Ax \leq b$.

Un théorème fondamental prouvé dans [60] stipule que la séparation et l'optimisation sont équivalentes au niveau de la complexité. En d'autres termes, si on dispose d'un algorithme de séparation, alors on peut optimiser en faisant appel à cet algorithme un nombre polynomial de fois. Ici la notion de polynomialité s'entend en fonction de la taille maximale de codage d'une facette (ou d'un sommet) de l'enveloppe convexe.

1.3 Problèmes combinatoires importants

1.3.1 Les problèmes de plus court chemin

Les problèmes du plus court chemin dans un graphe sont sans doute parmi les problèmes les plus importants dans l'optimisation combinatoire. Ces problèmes ont plusieurs champs d'application. Selon différents cas, ils peuvent être difficiles à résoudre si les poids des arêtes sont arbitraires. Par exemple, si les poids des arêtes d'un graphe $G = (V, E)$ sont tous égaux à -1 , alors la recherche des $s-t$ chemins de poids $1 - |V(G)|$ est simplement la recherche des $s-t$ chemins Hamiltoniens, dont le problème de décision est difficile au sens de la théorie de la complexité.

On distingue plusieurs cas de recherche de plus court chemin :

- Étant donnés deux sommets u et v , on s'intéresse à la recherche d'un plus court chemin de u à v .
- Étant donné un sommet u , on s'intéresse à la recherche d'un plus court chemin de u vers tous les autres sommets.
- Pour tout couple de sommets (u, v) , on cherche le plus court chemin entre u et v .

Pour tout cas de ces problèmes, il existe différents algorithmes de résolution selon les pondérations des arêtes.

Soit $G = (V, E)$ un graphe orienté, d'ordre n et de taille m , et soit $w : E \rightarrow \mathbb{R}$ une application représentant les poids (coûts) sur les arêtes de G . Sans perte de généralités, on suppose ici que le graphe G est simple et connexe.

1.3.1.1 L'algorithme de Dijkstra

Soient u et v deux sommets quelconques de V , et on s'intéresse dans cette section à la recherche d'un plus court chemin entre u et v . Au départ, on va supposer que ce chemin existe, et que les pondérations sur les arêtes sont toutes positives : $w : E \rightarrow \mathbb{R}^+$, afin d'éviter les circuits absorbants.

La résolution du problème de plus court chemin par l'algorithme de Dijkstra est alors détaillée ci-dessous :

À chaque étape, on sélectionne un sommet t et on fixe la valeur du plus court chemin de u à t .

On dispose alors de deux ensembles de sommets :

1. *Sommets examinés* : sera l'ensemble des sommets pour lesquels on connaît la valeur du plus court chemin à partir de u .
2. *Sommets à-traiter* : sera l'ensemble des sommets qui ont au moins un prédécesseur dans l'ensemble des sommets examinés.

Le critère de sélection d'un sommet dans l'ensemble *à-traiter* est de choisir le sommet t qui minimise le coût du chemin de u à t .

La sélection d'un sommet dans l'ensemble *à-traiter* pénalise à chaque fois l'algorithme de Dijkstra au sens de la complexité. En effet, cette dernière est de $O(n^2)$.

1.3.1.2 L'algorithme de Bellman

On a vu précédemment que l'algorithme de Dijkstra fonctionne sur des graphes avec circuits mais de coûts positifs. Dans ce qui suit, on considère des graphes sans circuits, mais de coûts réels et de signes quelconques. On note par $\mu(u, t)$ le plus court chemin de u à t .

L'algorithme de Bellman calcule le plus court chemin d'un sommet u vers tous les autres sommets t , et ses détails sont donnés ci-dessous :

- On numérote les sommets de 1 à n : tri topologique (tous les antécédents du sommet j auront des numéros inférieurs à j).
- Pour tout sommet y prédécesseur de t , on calcule $\mu(u, t) = \min\{\mu(u, y) + w(y, t)\}$

L'algorithme de Bellman est aussi polynomial et de complexité de $O(n + m)$.

1.3.1.3 L'algorithme de Bellman-Ford

Dans cette partie, on s'intéresse à la recherche d'un plus court chemin d'un sommet aux autres sommets, sur un graphe dont les pondérations peuvent être quelconques, et l'existence de circuits dans ce graphe n'est pas exclue.

On présente alors l'algorithme de Bellman-Ford, qu'on décrit ci-dessous :

C'est un algorithme qui fonctionne sur plusieurs étapes. À l'étape k on cherche de u vers tout sommet t un plus court chemin dans G ayant au plus k arcs, et la longueur d'un tel chemin. Dans le graphe G (d'ordre n) les chemins élémentaires ont tous $n - 1$ arcs, et l'ensemble des longueurs des plus courts chemins trouvés à l'étape $n - 1$ doit être identique à celui de l'étape n , à l'exception du cas où le graphe contient un circuit absorbant. Pour savoir si on est dans un tel cas, on utilise une variable booléenne qui nous informe si le passage de l'étape $k - 1$ à l'étape k a fait varier la longueur des chemins. Si k vaut n et que aucune distance n'a varié par rapport à l'étape précédente, alors le graphe ne contient pas de circuit absorbant, et les distances trouvées donnent la longueur des plus courts chemins de u vers les autres sommets. Dans le cas contraire dans lequel on constate un changement et que k est égal à n , alors il existe un circuit absorbant accessible à partir de u .

Cet algorithme est de complexité de $O(nm)$.

1.3.2 Problème de l'arbre couvrant de poids minimum

Dans cette partie, on suppose qu'on a un graphe $G = (V, E)$ non orienté et connexe. Les arêtes de G sont pondérées (coûts, poids,...).

Le problème de l'arbre couvrant de poids minimum du graphe G consiste à chercher un graphe partiel de G qui soit un arbre et de coût minimum.

Pour résoudre ce problème, on propose alors deux algorithmes de complexités différentes : l'algorithme de Kruskal, et l'algorithme de Prim.

1.3.2.1 Algorithme de Kruskal

Le principe de fonctionnement de cet algorithme est simple. En effet, il suffit de prendre les arêtes de poids minimum une à une, et vérifier à chaque fois que l'arête en cours ne constitue pas un cycle avec les arêtes prises avant.

On peut alors le détailler ainsi :

1. Trier les arêtes par ordre des poids croissants.
2. Procéder dans l'ordre du tri ainsi : prendre la première arête non examinée, et vérifier si elle forme un cycle avec les arêtes précédemment choisies, si oui la rejeter, sinon la garder, et continuer cette deuxième étape jusqu'à retenir $n - 1$ arêtes.

La complexité de l'algorithme de Kruskal est de l'ordre de $O(n^2 + m \log_2 m)$.

1.3.2.2 Algorithme de Prim

Dans cette partie, on va donner un deuxième algorithme pour la recherche d'un arbre couvrant de coût minimum, sans passer par le tri des poids des arêtes. En effet l'algorithme de Prim fonctionne ainsi : on pose T l'arbre en cours de construction. À chaque étape on ajoute à T un

sommet et une arête. À une étape donnée, on pose R l'ensemble des sommets qui ne sont pas dans T . À chaque sommet u de R on associe $proche(u)$ qui est le sommet de T tel que le poids de l'arête $(u, proche(u))$ soit minimum sur les autres sommets de T . On pose $d(u)$ (distance) le poids de cette arête. On choisit ainsi de faire entrer dans T le sommet u de distance minimum. Ce sommet u va servir comme *pivot* pour l'étape suivante. Ensuite, on met à jour les attributs des sommets r qui restent dans R et qui sont voisins du *pivot*, en comparant l'ancienne distance $d(r)$ à $w(pivot, r)$. Dans le cas où $d(r) > w(pivot, r)$ alors on met à jours $proche(r) = pivot$ et $d(r) = w(pivot, r)$.

La complexité de l'algorithme de Prim est de l'ordre de $O(n^2)$.

1.3.3 Le problème de flot maximum-coupe minimum

Dans cette sous section on considère un graphe $G = (V, E)$ orienté, et soient s et t deux sommets fixés de G . Soit $c : E \rightarrow \mathbb{R}^+$ une fonction dite de *capacité* sur les arcs du graphe. On distingue deux sommets s et t qui seront appelés *source* et *puits*.

Définition 1.1: Soit $R = (V, E, c)$ un réseau, s et t les sommets source et puits. Un **flot** f est une application de $V \times V$ dans \mathbb{R} vérifiant :

1. $f(x, y) = -f(y, x), \forall (x, y) \in E$.
2. $f(e) \leq c(e), \forall e \in E$.
3. $f(v, V) = 0, \forall v \in V \setminus \{s, t\}$.

D'une manière concrète, on dispose d'un graphe orienté dont les arêtes sont étiquetées par des nombres. On cherche à étiqueter par des nombres inférieurs ou égaux à C en assurant la nullité de leur somme orientée en tout point n'étant pas la source ou le puits.

La valeur du flot f est alors $|f| = f(s, V)$. On va s'intéresser à la recherche d'un flot f maximisant $|f|$.

1.3.3.1 La méthode de Ford-Fulckerson

Pour décrire cette méthode, on va utiliser le graphe des écarts, les chemins améliorants et les coupes, qu'on définit d'abord.

Soit $R = (V, E, c)$ un réseau, s et t sont la source et le puits respectivement, f un flot sur le réseau. La capacité résiduelle d'un arc u est notée par $c_f = c(u) - f(u)$.

Définition 1.2: (Le réseau des écarts). On appelle par **le réseau des écarts** le graphe $R_f = (V, E_f, c_f)$ où $E_f = \{(x, y) \in V \times V, c_f(xy) > 0\}$.

En d'autres mots, pour un arc donné u , si $c_f(u) = 0$, alors on remplace u par un arc étiqueté par $c_f(u)$, sinon on le supprime.

Définition 1.3: (Chemin améliorant). Un **chemin améliorant** noté μ , est un chemin de s à t dans R_f .

Définition 1.4: (Capacité résiduelle). La **capacité résiduelle** de μ est notée par $c_f(\mu) = \min\{c_f u \mid u \in \mu\}$.

Définition 1.5: (Coupe). Une **coupe** dans $R = (V, E, c)$ est une partition de V en (S, \bar{S}) avec $s \in S$ et $t \in \bar{S}$. La capacité de cette coupe est $c(S, \bar{S}) = \sum_{i \in S, j \in \bar{S}} c(i, j)$.

Le théorème suivant établit le lien entre la coupe et le flot dans un réseau.

Théorème 1.1: (Ford et Fulckerson). Soit f un flot dans un réseau $R = (V, E, c)$. Il y a équivalence entre les trois propositions suivantes :

1. f est un flot maximum.
2. Il n'existe pas de chemin améliorant.
3. Il existe une coupe (S, \bar{S}) telle que $|f| = c(S, \bar{S})$.

La méthode de Ford-Fulckerson consiste à démarrer d'un flot nul, et à l'améliorer en lui ajoutant f_μ tant qu'il existe un chemin améliorant μ . Elle est d'une complexité algorithmique de $O(f * |E|)$

Il existe un autre algorithme dit Algorithme d'Edmonds-Karp, qui consiste à choisir dans R_f à chaque étape de la méthode de Ford-Fulckerson, un plus court chemin en nombre d'arcs. Cet algorithme se base sur un graphe non orienté, et ne calcule pas les graphes des écarts. En d'autres mots, il ne parcourt que les arcs de capacité résiduelle strictement positive. Il est d'une complexité de $O(nm^2)$.

1.3.3.2 La Méthode des préflots- Algorithme de Goldberg-Tarjan

Définition 1.6: (Préflot). Soit $R = (V, E, c)$ un réseau de source s et de puits t . Un **préflot** est une application $f : E \rightarrow \mathbb{R}$, et qui vérifie les trois propriétés suivantes :

- $f(x, y) = -f(y, x)$;
- $f(u) \leq c(u), \forall u \in E$;
- $f(V, v) \geq 0, \forall v \in V \setminus \{s\}$

Le flot en **excès** en un sommet v est défini par $e(v) = f(V, v)$. Si $e(v) > 0$, alors v est dit un sommet **excédentaire**.

Définition 1.7: (Fonction de hauteur). Soit le réseau $R = (V, E, c)$ de source s et de puits t . Soit f un préflot.

Une fonction $h : V \rightarrow \mathbb{N}$ est dite une **Fonction de hauteur** si :

- $h(s) = |V|$
- $h(t) = 0$
- $h(u) \leq h(v) + 1, \forall (u, v) \in E_f$

L'algorithme de Goldberg-Tarjan se base sur deux opérations importantes :

1. Opération "*pousser*".
2. Opération "*élever*".

L'opération **pousser** qui s'applique sur un arc (u, v) peut être appliquée si :

- $e(u) > 0$ (u est excédentaire)
- $c_f(u, v) > 0$ ((u, v) n'est pas saturé par le flot)
- $h(u) = h(v) + 1$ (on pousse le flot de haut en bas)

L'algorithme 1 résume la fonction $pousser(u, v)$:

On dit que la poussée est **saturante** si après cette opération, l'arc (u, v) est saturé, donc $c_f(u, v) = 0$.

La deuxième opération dite **élever**, s'applique si :

- $e(u) > 0$ (u est excédentaire)
- $\forall (u, v) \in E_f, h(u) \leq h(v)$
- $u \neq t$

```

1 Fonction : pousser(u, v);
2  $d_f = \min(e(u), c_f((u, v)))$ ;
3  $f(u, v) = f(u, v) + d_f$ ;
4  $f(v, u) = -f(u, v)$ ;
5  $e(u) = e(u) - d_f$ ;
6  $e(v) = e(v) + d_f$ ;

```

Algorithm 1: Algorithme de l'opération **pousser**

```

1 Fonction : elever(u, v);
2  $h(u) = 1 + \min\{h(v) \mid (u, v) \in E_f\}$ ;

```

Algorithm 2: Algorithme de l'opération **élever**

L'algorithme 2 résume la fonction Élever.

Ensuite, on définit la fonction *init-préflot* (algorithme 3) qui va saturer tous les arcs issus de la source s .

```

1 Fonction : init – preflot(R, s, t);
2 for  $u \in V$  do
3    $h(u) = 0$ ;  $e(u) = 0$ ;
4 end
5 for  $(u, v) \in E$  do
6    $f(u, v) = 0$ ;  $f(v, u) = 0$ ;
7 end
8  $h(s) = |V|$ ;
9 for  $v \in Successeur(s)$  do
10   $f(u, v) = c(s, v)$ ;  $f(v, s) = -c(s, v)$ ;  $e(v) = f(s, v)$ ;
11 end

```

Algorithm 3: Algorithme d'initialisation du préflot

L'algorithme 4 est le corps principal de l'algorithme de préflot, dit aussi l'algorithme de *Push/Relabel*. On peut trouver différentes façons pour l'implémenter, et chaque façon nous donne une complexité différente de l'algorithme :

- FIFO : qui donne une complexité de $O(n^3)$.
- Plus haut degré : une complexité de $O(n^2\sqrt{m})$.

1.3.3.3 L'algorithme de Gomory-Hu

L'algorithme de Gomory-Hu est aussi une autre méthode de calcul de flot maximum-coupe minimum qui opère sur un graphe en le transformant en un arbre qui va représenter les flots maximums entre chaque deux sommets du graphe G .

Cet algorithme est d'une nature récursive. Il consiste à diviser l'ensemble des sommets du graphe G en deux sous ensembles R et S , et faire pareillement par la suite sur ces deux derniers ensembles. Initialement, il est appliqué sur tous les sommets du graphe.

Au départ, on sélectionne d'une manière aléatoire deux sommets du graphe, et on applique

- 1 *init* – *preflot* **while** \exists une opération pousser ou élever **do**
- 2 sélectionner une opération valide et l'appliquer

Algorithm 4: Algorithme de préflot

par exemple l'algorithme de Ford-Fulkerson pour calculer un flot maximum-coupe minimum entre ces deux sommets. Après ce calcul, on obtient deux sous-ensembles de sommets (R et S). Chacun des deux sommets sélectionnés au départ va appartenir à un des deux ensembles (R ou S). On continue encore cette procédure en choisissant à chaque fois deux sommets des deux dernières parties formées, et on calcule de nouveau un flot maximum entre ces deux sommets. Ainsi on aura calculé $n - 1$ flots maximums dans ce graphe, et le résultat est l'arbre de Gomory-Hu, qui nous donne le flot maximum (coupe minimum) entre chaque couple de sommets de G . La complexité de cet algorithme est de l'ordre de $O(n^2 \sqrt{m})$.

1.3.4 Les T-join et T-coupe

Étant donné un graphe non orienté $G = (V, E)$, et un sous-ensemble de sommets T de V , tel que T est de cardinalité paire. Un sous ensemble d'arêtes F est dit un **T-Join** s'il coïncide avec l'ensemble des sommets de degrés impairs dans le graphe $G' = (V, F)$. Une **T-coupe** est un ensemble d'arêtes de la forme de $\delta(W)$ où W est un ensemble de sommets tels que la cardinalité de $|T \cap W|$ est impaire.

1.3.4.1 Le problème de T-coupe de poids minimum

Il s'agit de chercher dans le graphe G la T-coupe de poids minimum. Grâce à l'algorithme de Gomory-Hu décrit ci-dessus, on construit un arbre de coupes, et on choisit parmi elles, celles qui représentent des T-coupes, ensuite on choisit la moins coûteuse [80].

1.3.4.2 Le problème du T-join de poids minimum

Sur les arêtes du graphe G , on définit la fonction des coûts $w : E \rightarrow \mathbb{R}$. Soit T un sous ensemble de V de cardinalité paire.

On s'intéresse dans cette partie à la recherche d'un T-join de poids minimum. Ce problème peut être la généralisation de plusieurs problèmes d'optimisation combinatoire. On cite par exemple :

- Si les poids des arêtes du graphe G sont positifs, et que T est l'ensemble de sommets de degrés impairs, alors on obtient le problème du postier chinois non orienté.
- Si $T = V$, alors le T-Join de cardinalité $\frac{|V|}{2}$ est en effet un couplage parfait du graphe G , et donc le problème du T-join de poids minimum peut se réduire au problème du couplage parfait de poids minimum.

Edmonds et Johnson [80] ont montré que, dans le cas où les poids des arêtes du graphe G sont positifs, le problème du T-join de poids minimum peut être résolu en $O(n^3)$.

1.3.5 Le problème de couplage

Dans cette section, on s'intéresse à l'existence d'un ensemble d'arêtes deux à deux disjointes recouvrant tous les sommets d'un graphe.

Un **couplage** de $G = (V, E)$ est un ensemble d'arêtes $M \subset E$ tel que deux arêtes de M n'ont pas d'extrémités communes. En d'autres mots, dans le graphe partiel $G' = (V, E')$, on a $d_{G'}(v) \leq 1$, $\forall v \in V$.

Les sommets v pour lesquels on a $d_{G'}(v) = 1$ sont dits **M-saturés**. Un couplage M est dit **Maximal** si $\forall e \in E \setminus M, M \cup \{e\}$ n'est pas un couplage. Un couplage M est **Maximum** si pour tout couplage M' , on a $|M'| \leq |M|$. On dit finalement qu'un couplage M est **Parfait** s'il sature tous les sommets du graphe G .

Une chaîne est **M-alternée** si ses arêtes sont alternativement dans M et dans $E \setminus M$.

Le problème de couplage est l'un des problèmes de base. Il consiste à trouver un couplage de cardinalité maximum, ce qui signifie qu'on cherche à couvrir un nombre maximum de sommets. Chaque couple de sommets seront reliés par une et une seule arête, et deux arêtes du couplage n'auront jamais la même extrémité. Tutte et Berge ont montré que pour un graphe non orienté, la cardinalité d'un couplage maximum est égale à $\min_{V' \subseteq V} \frac{|V| + |V'| - \text{odd}(V \setminus V')}{2}$ où $\text{odd}(V \setminus V')$ représente le nombre de composantes dans le sous graphe $G - V'$ qui ont un nombre impair de sommets (voir [60]). Edmonds a donné une caractérisation polyédrale du problème de couplage :

$$\begin{aligned} x(e) &\geq 0, & \forall e \in E \\ x(\delta(v)) &\leq 1, & \forall v \in V \\ x(E(W)) &\leq \lfloor \frac{|W|}{2} \rfloor & \forall W \subseteq V, |W| \text{ impair} \end{aligned}$$

Si on suppose que les arêtes du graphe G sont pondérées, on peut alors chercher un couplage maximum de poids minimum. Et si de plus on veut que notre couplage soit parfait, et que les arêtes sont de poids minimum, on s'intéressera à un couplage parfait de poids minimum.

Dans le cas où le graphe G est un graphe biparti, la recherche d'un couplage maximum peut se faire avec la technique du chemin augmentant. En effet, on suppose qu'on a un couplage de départ M , et que les deux parties des sommets de G sont U et W , et on oriente toute arête (u, w) avec $u \in U$ et $v \in W$, ainsi :

- si $e \in M$, alors on oriente e de w vers u .
- sinon, on l'oriente de u vers w .

Ainsi on obtient un graphe orienté noté D dont les sommets sont $U' = U \setminus \cup M$ et $W' = W \setminus \cup M$. On cherche par la suite dans D un chemin augmentant (s'il existe) à partir d'un sommet quelconque de U' vers un autre sommet de W' . De cette façon, on peut trouver un couplage plus large que M .

1.3.5.1 Le problème de b-couplage

Le problème de **b-couplage** est considéré comme une généralisation du problème de couplage. En effet, et étant donné $b : V \rightarrow \mathbb{Z}^+$, le b-couplage est alors une fonction $x : E \rightarrow \mathbb{Z}^+$ et tel que $x(\delta(v)) \leq b(v)$, $\forall v \in V$. Un b-couplage parfait consiste alors à remplacer cette inégalité par $x(\delta(v)) = b(v)$, $\forall v \in V$. Le polytope des b-couplages (avec des capacités de 1) est défini par [80] :

$$\begin{aligned} 0 \leq x(e) \leq 1, & \quad \forall e \in E \\ x(\delta(v)) \leq b(v), & \quad \forall v \in V \end{aligned}$$

$$x(E(W)) + x(F) \leq \lfloor \frac{b(W) + |F|}{2} \rfloor \quad \forall W \subseteq V, F \subseteq \delta(W), (b(W) + |F|) \text{ impair} \quad (1.4)$$

Les contraintes (1.4) (dites de blossom) sont importantes. En effet, on les trouve dans les enveloppes convexes de plusieurs problèmes d'optimisation NP-Difficiles comme le problème du voyageur de commerce (TSP). À cause de ce fait, on va s'intéresser à leur séparation.

Algorithme de séparation de Padberg&Rao

L'algorithme de Padberg&Rao consiste à séparer en temps polynomial les inégalités (1.4). Les étapes de cet algorithme sont les suivantes :

1. Construire un nouveau graphe \hat{G} avec de nouveaux poids \hat{x} sur les arêtes.
2. Affecter un ordre de parité (pair/impair) aux sommets de \hat{G} , et soit T l'ensemble des sommets impairs.
3. Calculer une coupe minimum impaire dans \hat{G} :
 - Calculer un arbre de coupes dans le graphe \hat{G} .
 - Vérifier sur toutes les $|T| - 1$ arêtes de l'arbre de coupes, si son poids < 1 , et que la coupe qui l'induit est impaire.

Pour mieux comprendre le fonctionnement de cet algorithme, on détaille ces étapes :

– Construire un nouveau graphe \hat{G} avec de nouveaux poids \hat{x} sur les arêtes :

Pour la construction du nouveau graphe \hat{G} , on prend chaque arête $e = (i, j)$ du graphe initial, et on la divise en deux, on y ajoutant un nouveau sommet k_e au milieu. On obtient deux arêtes dont les poids sont : $\hat{x}_{i,k_e} = x_e$ et $\hat{x}_{k_e,j} = 1 - x_e$. Le nouveau graphe $\hat{G} = (\hat{V}, \hat{E})$ aura $\hat{V} = |V| + |E|$, et $\hat{E} = 2|E|$.

– Affecter un ordre de parité (pair/impair) aux sommets de \hat{G} , et soit T l'ensemble des sommets impairs :

Soit $u \in \hat{V}$, alors u est impair si et seulement si :

- $u = k_e \in V$, ou bien
- $u \in V$, et que le nombre d'arêtes complémentaires incidentes à u est impair

dans le cas contraire u sera pair.

– Calculer une coupe minimum impaire dans \hat{G} :

Si la valeur d'une coupe minimum impaire est inférieure à 1 alors on a une violation des contraintes (1.4). Soit $T = \{\text{sommets impairs} \in \hat{V}\}$. Une coupe $\delta(U)$ est dite impaire si elle vérifie que la cardinalité $|U \cap T|$ est impaire. La recherche d'un arbre de coupes peut être effectuée grâce à l'algorithme de Gomory-Hu qui utilisera $|T| - 1$ flots maximums (coupes minimums) qu'on peut calculer avec l'algorithme de Goldberg-Tarjan.

L'algorithme de Padberg&Rao est alors d'une complexité de l'ordre de $O(|E|^3 \log(|V|))$.

1.3.6 Le problème du voyageur de commerce

Étant donné un graphe $G = (V, E)$ dont les arêtes sont pondérées par des poids $w : E \rightarrow \mathbb{R}$. Un circuit (cycle) Hamiltonien est un circuit (cycle) passant par tous les sommets de G une

et une seule fois. Le coût de ce circuit est la somme des coûts de ses arêtes. Le problème du voyageur de commerce (TSP pour Travelling Salesman Problem) consiste alors à trouver un circuit (cycle) Hamiltonien de G de coût minimum. C'est un problème typique d'optimisation combinatoire, avec quelques applications réelles, comme par exemple le problème de tournées de véhicules, ...

Le problème du voyageur de commerce est connu comme étant NP-Difficile [45].

Le modèle suivant présente l'une de ses formulations en programmation linéaire. À toute arête e on associe une variable x_e qui peut prendre deux valeurs : 1 si l'arête e fait partie de la solution, et 0 sinon.

$$\begin{aligned}
 & \min \sum_{e \in E} w_e x_e \\
 & \sum_{i=1}^n x_{ij} = 1, \forall j \in V \\
 & \sum_{j=1}^n x_{ij} = 1, \forall i \in V \\
 & \sum_{i,j \in S} x_{ij} \leq |S| - 1, \forall S \subseteq V, 2 \leq |S| \leq n - 1 \\
 & x_e \in \{0, 1\}, \forall e \in E.
 \end{aligned} \tag{1.5}$$

La fonction objectif consiste à minimiser le coût de la tournée. Les deux premières contraintes consistent à assurer que le voyageur arrive et repart d'une ville, puis les troisièmes contraintes sont dites contraintes des sous tours, et elles consistent à éviter de revenir au point de départ avant la réalisation de la tournée Hamiltonienne.

Le nombre d'inégalités d'élimination des sous tours est proche de 2^n , ceci rend la résolution du modèle impossible, mais avec la relaxation linéaire de cette dernière formulation, on peut arriver à résoudre plusieurs cas.

Parmi les méthodes utilisées pour résoudre ce problème, on cite les méthodes exactes. Ces dernières sont beaucoup utilisées, à cause de leur efficacité. En effet, il s'agit de construire un algorithme de Branch&Cut dans lequel on propose de nouvelles inégalités valides pour le problème du TSP, comme les inégalités dites de *blossom* proposées par Edmonds, ou alors d'autres inégalités valides dites de peigne (*comb*) utilisées dans une méthode de Branch&Cut.

D'une autre part, les heuristiques aussi ont fait leurs preuves. L'heuristique du plus proche voisin est l'une des premières et plus simples proposées pour la résolution du problème de TSP. La méthode construit un cycle Hamiltonien en ajoutant un par un les noeuds à la fin des tours partiels. Cette heuristique trouve une solution en $O(n^2)$.

L'heuristique de Christofides est très efficace pour ce problème. En effet, elle garantit un rapport d'approximation de $\frac{3}{2}$ dans le cas du TSP Euclidien. Elle consiste à construire un arbre de poids minimum sur le graphe G , puis ajoute aux arêtes de cet arbre, celles qui réalisent un couplage de poids minimum entre les noeuds de degré impair. L'ensemble des arêtes forme ainsi un graphe connexe dans lequel chaque noeud a un degré pair. De ce dernier graphe, on peut extraire un cycle Eulérien puis éliminer les visites multiples à un noeud pour obtenir un cycle Hamiltonien.

1.4 Réseaux à composantes unicycliques

1.4.1 Les matroïdes

On dit que $M = (E, \mathfrak{F})$ est un matroïde, si et seulement si on a les trois propriétés suivantes :

1. $\emptyset \in \mathfrak{F}$.
2. Si $A \in \mathfrak{F}$ et que $B \subseteq A$, alors $B \in \mathfrak{F}$.
3. Si $A, B \in \mathfrak{F}$, et que $|B| > |A|$ alors $\exists e \in B \setminus A$, tel que $A \cup \{e\} \in \mathfrak{F}$.

Si \mathfrak{F} ne satisfait que les deux premières propriétés précédentes, alors on parle d'un système d'indépendance. Une *base* de E est un ensemble maximal dans E , et toutes les bases d'un matroïde ont la même cardinalité. Une fonction *rang* notée par ρ est définie par $\rho : 2^E \rightarrow \mathbb{Z}$ telle que

$$\rho(A) = \max\{|X|, X \subseteq A; \text{ et } X \in \mathfrak{F}\}, A \subseteq E$$

On appelle un matroïde *dual* le matroïde $M^d = (E, \mathfrak{F}^d)$ défini par

$$\mathfrak{F}^d = \{J \subseteq E : E - J \text{ contient une } M\text{-base}\}$$

La fonction rang du matroïde dual est donnée par

$$\rho^d(A) = |A| + \rho(E - A) - \rho(E), A \subseteq E$$

Dans [85], [94], et [108], on peut distinguer plusieurs matroïdes. On en cite :

- **Le matroïde matriciel** : étant donnée une matrice réelle $A_{m \times n}$. Soient $E = \{a_1, a_2, \dots, a_n\}$ l'ensemble des colonnes de A et $\mathfrak{F} = \{J \subseteq E : \{a_j\}_{j \in J} \text{ est linéairement indépendant}\}$, alors on forme un matroïde matriciel.
- **Le matroïde graphique** : étant donné un graphe G . On pose E est l'ensemble d'arêtes de G , et si on pose : $\mathfrak{F} = \{J \subseteq E : G = (V, J) \text{ est une forêt}\}$ on obtient alors un matroïde graphique.
- **Le matroïde transversal** : étant donné un graphe $G = (V_1 \cup V_2, E)$, et on définit sur l'ensemble des sommets V_1 l'ensemble $\mathfrak{F} = \{J \subseteq V_1 : \text{il existe un couplage dans } G \text{ recouvrant } J\}$. Si V_2 est une collection des sous ensembles de V_1 , et que les arêtes du graphe G traduisent l'appartenance des éléments à ces ensembles, alors un ensemble indépendant est dit transversal.

1.4.1.1 Les matroïdes bi-circulaires

Étant donné un graphe $G = (V, E)$, et si p est le nombre de composantes connexes de G , alors le nombre cyclomatique de G serait de $|E| - |V| + p$.

$M = (E, \mathfrak{F})$ est dit un **matroïde bi-circulaire** d'un graphe G , avec E qui sera l'ensemble des arêtes du graphe G , et les circuits de M seront les ensemble minimaux d'arêtes connexes dont le nombre cyclomatique correspondant est de 2. De plus, un ensemble d'arête est dit *indépendant* dans M si et seulement si le sous graphe induit par ce dernier contient au plus un cycle dans chaque composante.

Plus de détails sur le matroïde bi-circulaire se trouvent dans [94], [108].

Nous prouvons dans ce qui suit le fait que le matroïde bi-circulaire est bien un matroïde.

Proposition 1.1: Soit $G = (V, E)$ un graphe simple, d'ordre n . Alors $M = (E, \mathfrak{F})$ est un matroïde, tel que $\mathfrak{F} = \{I \subseteq E, I \text{ a ses composantes connexes qui contiennent au plus un cycle}\}$.

Preuve:

- La première condition est évidente.
- Si on a $A \in \mathfrak{F}$, alors ceci veut dire que les composantes connexes de A contiennent au plus un cycle, et donc dire que $B \subseteq A$, ceci veut dire que B est soit un cycle, ou alors un arbre, mais dans tous les cas B reste une composante qui contient au plus un cycle, donc $B \in \mathfrak{F}$.
- Supposons que $A = \cup_{i=1}^k A_i$ qui sont les composantes connexes de A qui contiennent au plus un cycle. Alors, pour tout $i = 1, \dots, k$, on a $G_i = (V_i, A_i)$ tel que G_i est soit un arbre, ou une composante unicyclique. Ce qui nous amène à déduire que $n_A = \sum_{i=1}^k |V_i| = |A| + k'$, où $k' = 1, \dots, k$. On suppose aussi que $B = \cup_{j=1}^t B_j$, et donc $n_B = \sum_{j=1}^t |V_j| = |B| + t'$, où $t' = 1, \dots, t$. Sachant que $|B| > |A|$, on distingue alors deux cas :
 1. $t' > k'$: alors $|B| + t' > |A| + k'$ et donc $n_B > n_A$, ceci veut dire que B atteint plus de sommets que A , alors $\exists x$ atteint par B et non par A , et supposons que $e \in B$ qui contient x , donc $A \cup \{e\} \in \mathfrak{F}$.
 2. $n_B < n_A$: on suppose que les arêtes de B connectent chaque deux sommets de A de la même composante A_i , qui est un arbre ou une composante unicyclique. Raisonnons par l'absurde, et supposons qu'il n'y a pas de $e \in B \setminus A$, tel que $A \cup \{e\} \in \mathfrak{F}$, alors ceci veut dire que :
 - Soit $e \in B$, et alors e relie un sommet de la composante connexe unicyclique, avec un autre sommet de la même composante A_i .
 - Soit $e \in B$ et elle relie deux composantes unicycliques, d'où la violation de la troisième propriété du matroïde.
 Dans les deux cas, on trouve $|B| \leq |V_1| + |V_2| + \dots + |V_k|$, d'où $|B| \leq |A|$ qui est en contradiction avec $|B| > |A|$ \square

1.4.2 Partitionnement des réseaux en composantes unicycliques

Étant donné un graphe pondéré $G = (V, E)$ connexe et non orienté, et dont les poids des arêtes sont $w_e \geq 0, e \in E$. On suppose que $|V| = n$ et $|E| = m$. On s'intéresse à construire à partir du graphe G des composantes unicycliques de coût minimum.

1.4.2.1 L'algorithme glouton

Il est clair que nous sommes à la recherche d'une base optimale du matroïde bi-circulaire. On peut donc appliquer un algorithme glouton pour obtenir une telle base.

1.4.2.2 L'algorithme de couplage maximum de poids minimum

La deuxième solution, est d'utiliser un algorithme de recherche d'un couplage maximum de poids minimum, appliqué à un nouveau graphe construit à base de G . Ce nouveau graphe aura comme sommets ceux de G , ainsi que des nouveaux sommets qui sont toutes les arêtes de G . On construit donc un graphe biparti, d'une part on met les sommets de G , et de l'autre part,


```

1 Poser  $F = \emptyset$ ;
2  $w(e_1) \leq w(e_2) \leq \dots \leq w(e_m)$ ;
3 for  $i \leftarrow 1$  to  $m$  do
4   if  $F \cup \{e_i\} \in \mathfrak{F}$  then
5      $F := F \cup \{e_i\}$ ;
6   end
7 end

```

Algorithm 5: Algorithme glouton

on met les arêtes comme étant des sommets. Quant aux arêtes de ce nouveau graphe biparti, seront construites en reliant un sommet i de $V(G)$ avec un autre sommet de la forme ij , avec ij est une arête dans le graphe G . On garde les mêmes pondérations que celles existantes sur G (voir Figure (1.4)). Sur ce nouveau graphe, on cherche alors un couplage maximum de poids minimum. Cette méthode s'avère intéressante pour répondre à notre besoin, puisque le résultat est un partitionnement en composantes connexes unicycliques et de poids total minimum.

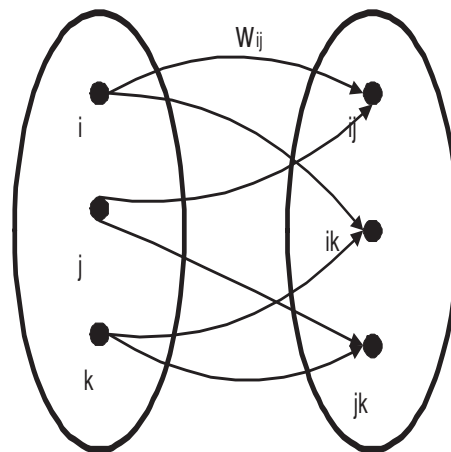


FIGURE 1.4 – Graphe biparti

Énoncé de l'algorithme

Soit U la partie gauche des sommets du nouveau graphe biparti, et W la partie droite.

1.5 Spectre des graphes

Étant donné un graphe G d'ordre n , et de taille m . On note par $A(G)$ ou simplement par A la matrice d'adjacence de G , et par L la matrice obtenue de la différence entre la matrice d'adjacence et la matrice des degrés D . L est connue sous le nom de *matrice d'admittance*, ou d'une manière générale *matrice du Laplacien*. On note aussi par d_1, d_2, \dots, d_n les degrés des

- 1 $M = \{e\}$, tel que e est l'arête de poids minimum orientée de U vers W ;
- 2 Orienter toutes les arêtes $e \in M$ de W vers U d'un poids $-w_e$;
- 3 Orienter toutes les arêtes e' qui ne sont pas dans M , de U vers W , d'un poids $+w_e$;
- 4 Poser $U_M = \{u \in U \setminus M\}$; et $W_M = \{u \in W \setminus M\}$;
- 5 Chercher tant que ça existe, le plus court chemin P de U_M vers W_M ;
- 6 Poser $M' = M \Delta P$ ($M \Delta P = (M \cup P) \setminus (M \cap P)$), et répéter les étapes 2, 3 et 4 ;
- 7 M' est le couplage recherché.

Algorithm 6: Algorithme de couplage sur un graphe biparti

sommets de G , ordonnés dans un ordre non-croissant $d_1 \geq d_2 \geq \dots \geq d_n$ en posant $d_1 = \Delta$ et $d_n = \delta$ qui sont les degrés maximum et minimum respectivement.

On note en général les valeurs propres associées à la matrice d'adjacence par $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$ qui sont ordonnées dans un ordre non-croissant, et par $\mu_1 \geq \mu_2 \geq \dots \mu_{n-1} \geq \mu_n = 0$ les valeurs propres associées à la matrice L et qui sont aussi ordonnées dans le même ordre.

On remarque aussi que la valeur propre $\mu_n = 0$, et cela est facilement démontrable, en prenant un vecteur propre x dont toutes ses coordonnées sont égales à 1.

Ce dernier résultat sur la plus petite valeur propre nulle du Laplacien est en lien direct avec le nombre de composantes connexes du graphe G . En effet, la multiplicité de cette valeur propre représente le nombre de composantes connexes de G .

1.5.1 Quelques liens avec l'optimisation combinatoire

Il existe des problèmes d'optimisation combinatoire NP-Difficiles, et dont la solution est loin d'être trouvée pour des instances moyennes, et pour des temps raisonnables. C'est à ce stade que les valeurs propres interviennent afin de donner des bornes significatives aux solutions des problèmes traités.

Dans ce qui suit, nous donnerons quelques liens importants qui existent entre les valeurs propres extrêmes et des problèmes d'optimisation combinatoire. Plus de détails et d'autres liens peuvent être consultés dans [102], [103].

1.5.1.1 Lien avec le diamètre d'un graphe

Dans un graphe G , la distance entre deux sommets u et v notée par $d(u, v)$ est définie comme étant la longueur du plus court chemin joignant u et v . Le diamètre d'un graphe G est alors la distance maximum entre toute paire de sommets dans G . Le calcul du diamètre d'un graphe n'a pas que des intérêts théoriques mais possède aussi différents champs d'application. Lorsque les graphes sont utilisés comme modèles dans les réseaux de communication, le diamètre correspond alors aux retards dans le passage des messages dans le réseau, et donc joue un rôle important dans l'analyse des performances et l'optimisation des coûts.

Le diamètre est aussi lié aux valeurs propres. La proposition suivante le met en évidence.

Proposition 1.2:
$$D(G) \leq \left\lceil \frac{\log(n-1)}{\log\left(\frac{\mu_1 + \mu_{n-1}}{\mu_1 - \mu_{n-1}}\right)} \right\rceil$$

1.5.1.2 Lien avec le nombre isopérimétrique

Le problème du nombre isopérimétrique est l'un des plus anciens. Il consiste à trouver parmi toutes les courbes d'une longueur donnée, celle qui renferme la plus grande surface. En théorie des graphes, ce problème reste essentiellement le même. En effet, il consiste à calculer le nombre de sommets d'un ensemble S , et la surface interprétée par le nombre d'arêtes $e(S, \bar{S})$ séparant les ensembles S de \bar{S} , et le problème du nombre isopérimétrique est de s'intéresser au rapport $\frac{e(S, \bar{S})}{|S|}$.

Le nombre isopérimétrique d'un graphe G est donné par

$$i(G) = \min\left\{\frac{e(S, \bar{S})}{|S|} \mid S \subset V, 0 \leq |S| \leq \frac{n}{2}\right\}$$

Le calcul du nombre isopérimétrique d'un graphe est un problème NP-Difficile [101], [102]. Le lien de ce problème avec les valeurs propres extrêmes du Laplacien d'un graphe nous permet de borner ce nombre.

Proposition 1.3: Soit G un graphe pondéré d'ordre n . Δ est le plus grand degré dans G . On a alors

$$\frac{\mu_{n-1}}{2} \leq i(G) \leq \sqrt{2\Delta\mu_{n-1}}$$

L'inégalité $i(G) \leq \sqrt{2\Delta\mu_{n-1}}$ est dite de *Cheeger*.

1.5.1.3 Lien avec le problème de coupe maximum (Max-Cut)

Pour rappel, le problème de coupe maximum (Max-Cut) consiste à trouver une coupe de poids maximum dans un graphe. On note :

$$mc(G) = \max\{e(S, \bar{S}) \mid S \subset V(G)\}$$

Ce problème est classé NP-Difficile au sens de la complexité algorithmique [102].

La proposition suivante met en évidence une borne supérieure de la solution du problème de coupe maximum en fonction du rayon spectral du Laplacien correspondant.

Proposition 1.4: Étant donné un graphe G pondéré et d'ordre n . Alors

$$mc(G) \leq \frac{n}{4}\mu_1$$

Réseaux à composantes unicycliques avec contrainte sur la taille des cycles

L'objet de ce chapitre est l'étude du problème de synthèse des réseaux à composantes connexes unicycliques respectant une contrainte sur la taille des cycles. Le réseau ne doit pas contenir des cycles d'une taille inférieure ou égale à p . L'étude proposée est polyédrale. Nous présentons donc un ensemble d'inégalités valides dont nous étudions la dimension des faces induites ainsi que leurs algorithmes de séparation. Des résultats numériques montrent que l'approche permet de résoudre des problèmes pratiques de dimensions raisonnables.

2.1 Introduction

Étant donné un graphe pondéré et non orienté désigné par $G = (V, E)$, on souhaite calculer un sous-graphe de poids total minimum $G' = (V, E')$ ($E' \subset E$) dont les composantes sont toutes unicycliques. En outre, le nombre d'arêtes de chacun des cycles obtenus dans G' ne doit pas être strictement inférieur à $p + 1$.

Ce problème a des applications dans les réseaux de télécommunications. En effet, on souhaite en général construire un réseau de coût total minimum avec un certain niveau de résistance aux pannes. La structure du cycle (ou l'anneau dans le contexte des réseaux) est une bonne alternative pour apporter une certaine résistance aux pannes simples de liens. Les demandes en trafic entre des noeuds d'un même cycle sont protégées contre ces éventuelles pannes alors que les autres demandes peuvent être interrompues. Cependant, des cycles de petites tailles n'offrent qu'une protection limitée, d'où la considération d'une contrainte sur la taille des cycles. D'autres contraintes peuvent aussi être importantes à considérer (la taille des composantes connexes, le nombre des composantes connexes, des contraintes sur les degrés des noeuds, etc).

Dans ce chapitre, on considère uniquement la contrainte portant sur la tailles des cycles des composantes connexes unicycliques obtenues.

D'importants travaux ont été réalisés dans le domaine de synthèse des réseaux sécurisés. Les références [5, 34, 52, 55, 77] sont quelques exemples. Une vaste littérature porte également sur

les problèmes de cycles. [83] est une bonne référence qui présente un état de l'art des travaux de localisation des cycles dans les réseaux de télécommunications (et transports).

Ce nouveau problème possède plusieurs liens avec d'anciens problèmes. Par exemple si $p = |V| - 1$, on obtient le problème du voyageur de commerce. Ceci implique clairement que notre problème est aussi NP-Difficile [45]. Si les contraintes sur la taille des cycles ne sont plus imposées ($p = 2$), le problème devient alors facile. En effet, on sait que l'ensemble des sous-ensembles d'arêtes qui induisent un graphe dont les composantes contiennent au plus un cycle est un matroïde. Ce matroïde est connu sous le nom de matroïde bi-circulaire (voir [94]). Un simple algorithme glouton peut donc être appliqué pour calculer un réseau de coût total minimum dont les composantes connexes sont unicycliques (voir Chapitre 1).

Il est important de signaler qu'il est aussi facile de construire un réseau (graphe) unicyclique (graphe connexe qui contient exactement un seul cycle). Ceci provient du fait que l'ensemble des sous-ensembles d'arêtes qui induisent un graphe qui contient au plus un cycle est aussi un matroïde. Ce matroïde est inclus dans le matroïde bi-circulaire (voir Chapitre 1).

Si les contraintes d'unicyclicité (au plus un cycle) vont être remplacées par des contraintes dites de cyclicité (chaque composante est exactement un cycle) on obtient alors le problème du 2-Couplage (2-factor) qui est un problème facile (voir [117]). Si des contraintes sur la taille des cycles sont ajoutées (pas de cycles de taille inférieure à 5), le problème devient alors difficile (voir [22]).

Notre problème a aussi des liens avec le problème de l'étoile-anneau qui consiste à construire un réseau qui contient un seul cycle autour de quelques noeuds, alors que les autres noeuds sont directement liés à ceux qui forment le cycle. Ce problème et quelques variations sont étudiés dans [2, 82].

Si les contraintes d'unicyclicité sont relaxées, on obtient alors un autre problème connu consistant à trouver un graphe de poids minimum (ou maximum) qui ne contient pas de cycle de taille inférieure à $p + 1$. Ce problème est équivalent au problème de couverture dans lequel on cherche un ensemble d'arêtes de poids minimum (maximum) intersectant tout cycle dont la taille ne dépasse pas p . Quand $p = 3$, on sait déjà que le problème est facile si le graphe n'est pas contractible à $K_5 \setminus e$ (le graphe complet avec 5 noeuds moins une arête) [39]. Cependant, le problème est en général NP-Difficile [128].

Ce chapitre est organisé comme suit : les notations et une formulation mathématique sont introduites dans la section suivante. Quelques inégalités valides sont décrites dans la section 3.4. Les faces induites par ces inégalités valides sont étudiées dans la section 2.4. Dans la section 2.5, on présente un algorithme à plans coupants pour résoudre le problème du réseau à composantes connexes et unicycliques de coût total minimum avec la contrainte sur la taille des cycles. Les problèmes de séparation relatifs aux inégalités valides sont étudiés. La section 2.6 est dédiée aux résultats numériques. On cloture ce chapitre par une conclusion qu'on peut trouver en section 2.7.

2.2 Notations et Formulation Mathématique

Soit \mathcal{H}_p l'ensemble des graphes tels que chaque composante connexe contient exactement un cycle ayant au moins $p + 1$ arêtes. En d'autres termes, toutes les composantes connexes sont

des graphes unicycliques et il n'y a pas de cycle de taille inférieure ou égale à p .

Étant donné un graphe $G = (V, E)$, on considère tous les sous-graphes $G' = (V, E')$ ($E' \subset E$) qui appartiennent à \mathcal{H}_p . Un sous-graphe est représenté par un vecteur d'incidence $(x_e)_{e \in E'}$ où $x_e = 1$ si e appartient au sous-graphe et 0 sinon. L'enveloppe convexe des vecteurs d'incidence x de ces sous-graphes qui appartiennent à \mathcal{H}_p est notée par $P(G, p)$. Calculer un sous-graphe de poids minimum qui appartient à \mathcal{H}_p est ainsi équivalent à calculer un vecteur de coût minimum dans $P(G, p)$.

Dans ce chapitre, on suppose que $G = (V, E)$ est un graphe simple, connexe et non orienté (si G n'est pas connexe, on peut s'intéresser à chacune de ses composantes connexes). Le nombre de sommets est noté par $n = |V|$. On suppose aussi que $n \geq 3$ et $p \leq n - 1$. Un coût w_e est associé à chaque arête $e \in E$.

Une arête $e \in E$ est parfois notée par (i, j) où i et j sont les deux extrémités de e . Étant donné un vecteur $(\alpha_e)_{e \in E}$, α_e peut aussi être noté par α_{ij} . Si A est un sous-ensemble de sommets, alors $\delta(A)$ est l'ensemble des arêtes qui ont exactement une extrémité dans A . Si $A = \{v\}$ où $v \in V$, alors on peut écrire $\delta(v)$ au lieu de $\delta(\{v\})$.

Soient A et B deux sous-ensembles de sommets disjoints, $\delta(A) \setminus \delta(B)$ représente l'ensemble des arêtes qui ont une extrémité dans A et l'autre extrémité dans $V \setminus \{A \cup B\}$. Soit $(x_e)_{e \in E}$ un quelconque vecteur indexé par E , alors $x(A, B)$ est utilisé pour noter la somme des poids des arêtes qui ont une extrémité dans A et une extrémités dans B : $x(A, B) = \sum_{(i,j) \in E, i \in A, j \in B} x_{ij}$. Quand

$B = \{v\}$, on écrit $x(A, v)$ au lieu de $x(A, \{v\})$. On écrit aussi $x(A)$ pour noter la somme des poids des arêtes qui ont les deux extrémités dans A . $x(A) = \sum_{(i,j) \in E, i \in A, j \in A} x_{ij}$. Si C est un ensemble

d'arêtes $C \subset E$, alors $x(C)$ représente naturellement la somme des poids des éléments de C : $x(C) = \sum_{e \in C} x_e$.

Dans ce qui suit, on va donner la formulation mathématique pour notre problème de synthèse de réseaux.

Pour commencer, on sait que le matroïde bi-circulaire est un matroïde transversal [94]. Autrement dit, les ensembles indépendants du matroïde bi-circulaire peuvent être représentés par un couplage. Pour être plus précis, on pose H le graphe biparti dont l'ensemble des sommets est donné par $V \cup E$, et il existe une arête entre $v \in V$ et $e \in E$ si et seulement si v est une extrémité de e . Pour tout couplage de H , l'ensemble des arêtes de E incluses dans ce couplage est un indépendant du matroïde bi-circulaire. Les bases de ce matroïde sont données par des couplages de taille n (en effet, notre problème n'a pas de solution si le rang du matroïde bi-circulaire associé à G est inférieur à n).

Les observations données ci-dessus nous aident alors à construire une formulation mathématique des contraintes d'unicyclicité. Pour toute arête $e = (i, j) \in E$, on a une variable x_e et deux variables y_{ie} et y_{je} où y_{ie} est égal à 1 si le sommet i est couvert par l'arête (i, j) . x_e est donné par $y_{ie} + y_{je}$. En utilisant la caractérisation polyédrale et classique des couplages bipartis (voir [117]), on déduit que notre problème peut être formulé ainsi (après relaxations des

contraintes sur la taille des cycles) :

$$\begin{aligned}
& \min \sum_{e \in E} w_e x_e \\
& x_e = y_{ie} + y_{je}, \forall e = (i, j) \in E \\
& \sum_{e=(i,j) \in E} y_{ie} = 1, \forall i \in V \\
& 0 \leq y_{ie}, \forall i \in V, e = (i, j) \in E \\
& x_e \leq 1, \forall e \in E.
\end{aligned} \tag{2.1}$$

Les contraintes sur la taille des cycles peuvent être intégrées comme suit. Il suffit d'écrire que le nombre d'arêtes dans tout ensemble de sommets A de cardinalité au plus p ne peut pas dépasser $|A| - 1$.

$$x(A) \leq |A| - 1, A \subset V, 2 \leq |A| \leq p \tag{2.2}$$

Ainsi, notre problème est bien formulé par (2.1), (2.2) avec les contraintes d'intégrité (2.3) :

$$x_e \in \{0, 1\}, \forall e \in E. \tag{2.3}$$

Quand les solutions extrêmes de (2.1) sont des vecteurs en 0-1, celles de (2.1)+(2.2) peuvent être fractionnaires. En outre, la séparation des inégalités (2.2) n'est pas facile. On essaiera alors de renforcer le système (2.1)+(2.2) en ajoutant quelques inégalités qui sont valides pour $P(G, p)$.

2.3 Inégalités Valides

Proposition 2.1: Soient A et B deux ensembles disjoints de V tel que $|A| \leq p - 1$. Alors l'inégalité suivante est valide pour l'enveloppe convexe $P(G, p)$:

$$2x(A) + x(A, B) \leq 2|A| + |B| - 2. \tag{2.4}$$

Preuve: Soit C l'ensemble complémentaire de $A \cup B$. Sachant que $n = x(A) + x(B) + x(C) + x(\delta(C)) + x(A, B)$, l'inégalité $2x(A) + x(A, B) \leq 2|A| + |B| - 2$ est équivalente à $x(A) \leq |A| - 2 + x(B) + x(C) + x(\delta(C)) - |C|$. On a alors à montrer la validité de cette dernière inégalité.

D'abord, on remarque qu'on a nécessairement $x(C) + x(\delta(C)) - |C| \geq 0$. Ceci provient du fait que chaque composante connexe de C qui ne contient pas de cycle est nécessairement connectée à au moins un sommet dans $A \cup B$.

On suppose ensuite que $x(C) + x(\delta(C)) - |C| \geq 1$. Du fait que $|A| \leq p - 1 \leq p$, on a alors $x(A) \leq |A| - 1 = |A| - 2 + 1 \leq |A| - 2 + x(B) + x(C) + x(\delta(C)) - |C|$.

On suppose maintenant que $x(C) + x(\delta(C)) - |C| = 0$. Si on a $x(B) \geq 1$ ou $x(A) \leq |A| - 2$, alors l'inégalité $x(A) \leq |A| - 2 + x(B) + x(C) + x(\delta(C)) - |C|$ est valide. Il reste à montrer qu'on ne peut pas avoir simultanément $x(B) = 0$, $x(A) = |A| - 1$ et $x(C) + x(\delta(C)) - |C| = 0$.

$x(C) + x(\delta(C)) - |C| = 0$ implique que chaque composante connexe de C et qui ne contient pas de cycle est connectée à $A \cup B$ par exactement une arête. Ceci implique aussi que toute composante unicyclique de C (composante qui contient un cycle) n'est pas connectée à $A \cup B$. Comme on sait que $|A| \leq p - 1$, alors il n'y a pas de sommet dans B qui est lié (connecté) à deux sommets de A . Mais, on a aussi $x(B) = 0$ qui implique qu'il n'est pas possible de construire

un cycle. Autrement dit, il n'existe aucune composante connexe unicyclique qui contient les sommets de A ou de B . La figure (2.1) illustre le fait qu'on ne peut pas avoir simultanément les cas $x(B) = 0$, $x(A) = |A| - 1$ et $x(C) + x(\delta(C)) - |C| = 0$. \square

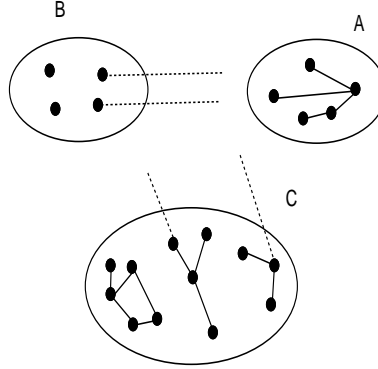


FIGURE 2.1 – Sur la validité de l'inégalité (2.4)

Proposition 2.2: Soient A et B deux ensembles disjoints de V . L'inégalité suivante est alors valide pour $P(G, p)$:

$$\sum_{v \in A} (2 - x(\delta(v))) + \sum_{v \in B} (x(\delta(v) \setminus \delta(A)) - 2) \leq n - (p + 1). \quad (2.5)$$

Preuve: Pour commencer, on s'intéresse au cas où $B = \emptyset$. Le nombre de sommets dont le degré est égal à 1 ne peut pas être supérieur à $n - (p + 1)$. En plus, $\sum_{v \in A} (2 - x(\delta(v)))$ est inférieur ou égal au nombre de sommets de A et dont le degré est de 1. Ceci implique alors que $\sum_{v \in A} (2 - x(\delta(v))) \leq n - (p + 1)$ pour tout sous-ensemble A .

Maintenant, et en utilisant le fait que $\sum_{v \in A} (2 - x(\delta(v))) + \sum_{v \in V \setminus A} (2 - x(\delta(v))) = 0$, on déduit que

$$\sum_{v \in V \setminus A} (x(\delta(v) - 2)) = \sum_{v \in A} (2 - x(\delta(v))) \leq n - (p + 1).$$

Autrement, si $A = \emptyset$ alors on a $\sum_{v \in B} (x(\delta(v) \setminus \delta(A)) - 2) \leq n - (p + 1)$ pour tout sous-ensemble B .

Intéressons nous maintenant au cas général dans lequel on a $A \neq \emptyset$ et $B \neq \emptyset$. Il est clair que $\sum_{v \in A} (2 - x(\delta(v))) + \sum_{v \in B} (x(\delta(v) \setminus \delta(A)) - 2)$ ne peut pas décroître si on élimine de A tous les sommets pour lesquels $2 - x(\delta(v)) \leq 0$. On peut alors supposer que $2 - x(\delta(v)) = 1$ pour tout sommet $v \in A$. Considérons le graphe obtenu par la suppression de A . Comme la suppression d'un sommet de degré 1 ne casse pas la propriété d'unicyclicité (toutes les composantes connexes restent unicycliques, et pas de nouvelles composantes connexes créées), alors le nouveau graphe satisfait les contraintes $\sum_{v \in B} (x(\delta(v)) - 2) \leq (n - |A|) - (p + 1)$ (on sait que les inégalités (2.5) sont valides quand A est vide). Notons aussi que dans le nouveau graphe $\sum_{v \in B} (x(\delta(v)) - 2)$ est égale à $\sum_{v \in B} (x(\delta(v) \setminus \delta(A)) - 2)$ dans l'ancien graphe. En utilisant le fait

que $2 - x(\delta(v)) = 1$ pour tout sommet $v \in A$, on déduit que $\sum_{v \in A} (2 - x(\delta(v))) + \sum_{v \in B} (x(\delta(v)) \setminus \delta(A) - 2) \leq n - (p + 1)$. \square

On verra après que les inégalités (2.5) peuvent être séparées en temps polynomial soit en minimisant une fonction sous-modulaire, ou en résolvant un problème classique de coupe minimum dans un graphe approprié.

Il est intéressant de noter que lorsque $A = \emptyset$ et $B = \{v\}$, les inégalités (2.5) se réduisent à

$$x(\delta(v)) \leq n - (p - 1).$$

La famille d'inégalités suivante est basée sur le fait que le nombre de composantes connexes ne peut pas excéder $\lfloor \frac{n}{p+1} \rfloor$ (Il y a au moins $1 + p$ sommets dans chaque composante).

Soit $(\alpha_e)_{e \in E}$ un ensemble de nombres réels. Pour tout nombre $0 \leq i \leq \lfloor \frac{n}{p+1} \rfloor - 1$, γ_{n+i} est le coût minimum (au sens des $(\alpha_e)_{e \in E}$) d'un graphe connexe avec $n + i$ arêtes. Soit T_α un arbre couvrant de poids minimum et β_i est le coût maximum des i arêtes de T_α . Autrement dit, $\beta_i = \max_{E' \subseteq T_\alpha, |E'|=i} \sum_{e \in E'} \alpha_e$.

Dans la sous-section 2.5.4, on verra que des algorithmes gloutons peuvent être utilisés pour calculer γ_{n+i} ainsi que β_i pour tout nombre i .

Proposition 2.3: L'inégalité suivante est valide pour $P(G, p)$:

$$\sum_{e \in E} \alpha_e x_e \geq \min_{0 \leq i \leq \lfloor \frac{n}{p+1} \rfloor - 1} (\gamma_{n+i} - \beta_i). \quad (2.6)$$

Preuve: Soit x un vecteur d'incidence représentant un graphe G de \mathcal{H}_p . Soit i le nombre de composantes connexes de G . On sait que $i \leq \lfloor \frac{n}{p+1} \rfloor$. Comme T_α est connexe, on peut alors ajouter à G quelques $i - 1$ arêtes de T_α pour le rendre connexe, et donc on obtient un graphe connexe avec $n + i - 1$ arêtes. Le coût de ce graphe n'est pas inférieur à γ_{n+i-1} . En outre, le coût de $i - 1$ arêtes ajoutées à G est clairement pas supérieur à β_{i-1} . En combinant ces deux observations, on en déduit alors que $\sum_{e \in E} \alpha_e x_e \geq \gamma_{n+i-1} - \beta_{i-1}$. En prenant le minimum de toutes les valeurs possibles de i , on trouve $\sum_{e \in E} \alpha_e x_e \geq \min_{0 \leq i \leq \lfloor \frac{n}{p+1} \rfloor - 1} (\gamma_{n+i} - \beta_i)$. \square

Plus de détails sur la séparation de cette famille d'inégalités seront donnés plus tard.

La borne supérieure $\lfloor \frac{n}{p+1} \rfloor$ du nombre de composantes connexes peut être exploitée pour découvrir de nouvelles inégalités valides.

Soit S un sous-ensemble de sommets. Pour tous les sommets i et j de S , $\delta(A_{ij})$ représente une coupe séparant i et j .

Proposition 2.4: L'inégalité suivante est valide pour $P(G, p)$:

$$\sum_{i,j \in S, i < j} \sum_{e \in \delta(A_{ij})} x_e \geq \max \left(0, |S| - \lfloor \frac{n}{p+1} \rfloor, \left\lceil \left(|S| - \lfloor \frac{n}{p+1} \rfloor \right) \frac{|S|}{2 \lfloor \frac{n}{p+1} \rfloor} \right\rceil \right). \quad (2.7)$$

Preuve: Soit x un vecteur d'incidence d'un sous-graphe couvrant de G qui appartient à \mathcal{H}_p . Considérons un réseau où la capacité de chaque arête $e \in E$ est x_e . On note par f_{ij} le flot maximum qu'on peut écouler entre i et j . On a $\sum_{i,j \in S, i < j} \sum_{e \in \delta(A_{ij})} x_e \geq \sum_{i,j \in S, i < j} f_{ij}$.

Il est aussi facile de voir que $\sum_{i,j \in S, i < j} f_{ij} \geq |S| - \lfloor \frac{n}{p+1} \rfloor$. Il reste alors à montrer que $\sum_{i,j \in S, i < j} f_{ij} \geq$

$(|S| - \lfloor \frac{n}{p+1} \rfloor) \frac{|S|}{2 \lfloor \frac{n}{p+1} \rfloor}$. Soit z le nombre de composantes connexes (donc $z \leq \lfloor \frac{n}{p+1} \rfloor$). Soit S_k le sous-ensemble de sommets de S qui appartient à la k^{eme} composante connexe. Comme $f_{ij} \geq 1$ quand i et j sont sur la même composante, on en déduit que $\sum_{i,j \in S, i < j} f_{ij} \geq |S_1|(|S_1| - 1)/2 + \dots + |S_z|(|S_z| - 1)/2$. Par la convexité de la fonction $\frac{x(x-1)}{2}$ on obtient $\sum_{i,j \in S, i < j} f_{ij} \geq \frac{z}{2} \frac{|S|}{z} (\frac{|S|}{z} - 1) = \frac{|S|}{2} (\frac{|S|}{z} - 1)$. En combinant cette inégalité avec $z \leq \lfloor \frac{n}{p+1} \rfloor$ et $\sum_{i,j \in S, i < j} \sum_{e \in \delta(A_{ij})} x_e \geq \sum_{i,j \in S, i < j} f_{ij}$ on arrive alors au résultat souhaité. \square

Il est intéressant de noter que lorsque $\lfloor \frac{n}{p+1} \rfloor = 1$, les inégalités (2.7) se réduisent à

$$\sum_{i,j \in S, i < j} \sum_{e \in \delta(A_{ij})} x_e \geq |S|(|S| - 1)/2$$

qui est une combinaison des inégalités $\sum_{e \in \delta(A_{ij})} x_e \geq 1$. Cette dernière inégalité est évidemment valide quand on a qu'une seule composante connexe (ça correspond à $|S| = 2$).

Quand $S = V$, les inégalités (2.7) peuvent être facilement retrouvées en utilisant le fait qu'on doit avoir $f_{ij} = 2$ pour quelques paires de sommets (on utilise les notations introduites dans la preuve précédente). Il n'est pas difficile de voir que $\sum_{i,j \in V, i < j} f_{ij}$ est minimum si le graphe contient exactement $\lfloor \frac{n}{p+1} \rfloor$ composantes connexes avec un cycle de taille $p+1$ dans chaque composante et tel que les tailles des composantes sont à peu près égales. En d'autres termes, si on considère deux composantes différentes, le nombre de sommets dans une composante ne doit pas dépasser 1 plus le nombre des sommets de l'autre composante (autrement, on décroît $\sum_{i,j \in V, i < j} f_{ij}$ en permutant un sommet d'une composante avec une autre). Après quelques calculs, on aura la valeur de $\sum_{i,j \in V, i < j} f_{ij}$ de ce graphe donnée par

$$\lfloor \frac{n}{p+1} \rfloor \frac{(p+1)p}{2} + \frac{1}{2} \lfloor \frac{n}{p+1} \rfloor \left(2n - \lfloor \frac{n}{p+1} \rfloor - \lfloor \frac{n}{p+1} \rfloor \lfloor \frac{n}{p+1} \rfloor \right)$$

Proposition 2.5: L'inégalité suivante est valide pour $P(G, p)$:

$$\sum_{i,j \in V, i < j} \sum_{e \in \delta(A_{ij})} x_e \geq \lfloor \frac{n}{p+1} \rfloor \frac{(p+1)p}{2} + \frac{1}{2} \lfloor \frac{n}{p+1} \rfloor \left(2n - \lfloor \frac{n}{p+1} \rfloor - \lfloor \frac{n}{p+1} \rfloor \lfloor \frac{n}{p+1} \rfloor \right). \quad (2.8)$$

\square

Étant donné un graphe de \mathcal{H}_p , il est clair que $\sum_{(i,j) \in \delta(i)} x_{ij} f_{ij}$ est égale à $2 + \sum_{(i,j) \in \delta(i)} x_{ij}$ si i appartient à un cycle et est égale à $\sum_{(i,j) \in \delta(i)} x_{ij}$ sinon. Comme il existe au moins $p+1$ sommets appartenant à un cycle, on peut alors directement en déduire que $\sum_{(i,j) \in E} x_{ij} f_{ij} \geq n + p + 1$.

Cette inégalité peut être améliorée si on dispose d'une bonne borne inférieure du nombre de composantes connexes. On rappelle que dans la preuve de la Proposition (2.4), on a observé que $\sum_{i,j \in S, i < j} f_{ij} \geq |S| - z$ et $\sum_{i,j \in S, i < j} f_{ij} \geq \frac{|S|}{2} (\frac{|S|}{z} - 1)$ où z est le nombre de composantes connexes. La seconde inégalité paraît difficile à exploiter alors que la première nous conduit à

$z \geq |S| - \sum_{i,j \in S, i < j} f_{ij}$. Ainsi, on obtient $\sum_{(i,j) \in E} x_{ij} f_{ij} \geq n + (p+1) \left(|S| - \sum_{i,j \in S, i < j} f_{ij} \right)$ pour tout sous-ensemble S . Notons que $\sum_{i,j \in S, i < j} f_{ij} \geq \frac{|S|}{2} \left(\frac{|S|}{z} - 1 \right)$ peut être introduit par la programmation semi définie positive. Cependant, nous allons dans cette thèse nous focaliser sur les méthodes linéaires.

Soit $\delta(A_{ij})$ une coupe séparant i et j . Alors, on a $\sum_{(i,j) \in E} x_{ij} f_{ij} \leq \sum_{(i,j) \in E} \sum_{e \in \delta(A_{ij})} x_{ij} x_e$. Pour toutes paires de sommets i et j de S , on considère toute coupe $\delta(A'_{ij})$ séparant i et j (on peut avoir $A_{ij} = A'_{ij}$). On a clairement $n + (p+1) \left(|S| - \sum_{i,j \in S, i < j} f_{ij} \right) \geq n + (p+1) \left(|S| - \sum_{i,j \in S, i < j} \sum_{e \in \delta(A'_{ij})} x_e \right)$. Par conséquent, l'inégalité suivante est valide

$$\sum_{(i,j) \in E} \sum_{e \in \delta(A_{ij})} x_{ij} x_e \geq n + (p+1) \left(|S| - \sum_{i,j \in S, i < j} \sum_{e \in \delta(A'_{ij})} x_e \right)$$

Le produit $x_{ij} x_e$ est bien sur inférieur à x_{ij} et à x_e . En remplaçant ce produit par soit x_{ij} ou x_e , on obtient une inégalité valide linéaire de $P(G, p)$. En effet, on verra plus loin dans la sous-section 2.5.5 que si on a un vecteur fractionnaire x , le produit est remplacé par le minimum de x_{ij} et x_e . L'inégalité valide est clairement établie ci-dessous.

Proposition 2.6: L'inégalité suivante est valide pour $P(G, p)$:

$$\sum_{(i,j) \in E} \sum_{e \in \delta(A_{ij})} x_{ij,e} \geq n + (p+1) \left(|S| - \sum_{i,j \in S, i < j} \sum_{e \in \delta(A'_{ij})} x_e \right) \quad (2.9)$$

où $x_{ij,e}$ note différemment x_{ij} ou x_e .

□

Il est important de bien comprendre que la notation $x_{ij,e}$ n'est pas liée à de nouvelles variables.

2.4 Sur la structure faciale de $P(G, p)$

On étudie la dimension de $P(G, p)$ et les inégalités valides définies dans la section précédente. Pour simplifier cette étude, on supposera dans cette section que $G = (V, E)$ est un graphe complet. On pose $m = \frac{n(n-1)}{2}$.

Proposition 2.7: La dimension de $P(G, p)$ est $m - 1$ si $2 \leq p \leq n - 2$, et $m - n$ si $p = n - 1$.

Preuve: : La dimension de $P(G, p)$ ne peut pas être supérieure à $m - 1$ puisque l'égalité $\sum_{e \in E} x_e = n$ est vérifiée. Si $p = n - 1$, alors $P(G, p)$ est l'enveloppe convexe des tournées Hamiltoniennes. Sa dimension est $m - n$ [95].

Supposons maintenant que $2 \leq p \leq n - 2$. On suppose aussi que $\alpha x = \beta$ pour tout vecteur x de $P(G, p)$. Soient u, v, w trois sommets. On considère aussi un graphe unicyclique basé sur un cycle de taille $n - 1$ plus une arête de plus. On suppose que : u est l'unique sommet pendant (de degré égal à 1), u est connecté à v , et w est l'un des deux voisins de v sur le cycle.

Ce graphe appartient à \mathcal{H}_p . Maintenant, en supprimant l'arête (w, v) et en ajoutant (w, u) , on obtient un autre graphe dans \mathcal{H}_p . Les deux graphes satisfont $\alpha x = \beta$. Par conséquent, on doit avoir $\alpha_{uw} = \alpha_{vw}$. Comme cela est valide pour trois sommets quelconques u, v, w , on en déduit alors que toutes les composantes de α sont égales. Cela implique que la contrainte $\alpha x = \beta$ est proportionnelle à $\sum_{e \in E} x_e = n$. Alors la dimension de $P(G, p)$ est $m - 1$. \square

Proposition 2.8: L'inégalité $x_e \geq 0$ définit une facette de $P(G, p)$ si et seulement si $n \geq 5$

Preuve: Si $p = n - 1$, le résultat est déjà connu [58, 95]. Dans ce qui suit, on s'intéressera au cas où $2 \leq p \leq n - 2$. En utilisant le fait que $x_e \geq 0$ est une facette du polytope de voyageur de commerce symétrique, on peut alors construire $m - n$ vecteurs z_1, z_2, \dots, z_{m-n} affinement indépendants qui représentent des tournées Hamiltoniennes qui ne traversent pas e . Tous ces vecteurs appartiennent à $P(G, p)$. Soient u et v les deux extrémités de e . On construit $n - 1$ autres vecteurs de $P(G, p)$ qui satisfont $x_e = 0$. On considère une tournée de taille $n - 1$ qui passe par tous les sommets sauf u . On obtient $n - 2$ nouveaux vecteurs $z_{m-n+1}, \dots, z_{m-2}$ de $P(G, p)$ en choisissant un sommet quelconque w de cette tournée (w différent de v) et en ajoutant une arête (u, w) . Pour tout sommet w , le graphe obtenu par cette méthode appartient à \mathcal{H}_p . Un dernier vecteur z_{m-1} peut être obtenu en considérant une tournée quelconque de longueur $n - 1$ qui contient u et non pas v avec une arête liant v avec un sommet $w \neq u$ dans le cycle.

D'abord, tous les vecteurs $(z_i)_{1 \leq i \leq m-1}$ satisfont $x_e = 0$. Supposons qu'il existe des nombres $(\alpha_i)_{1 \leq i \leq m-1}$ tels que $\sum_{i=1}^{i=m-1} \alpha_i z_i = 0$ et $\sum_{i=1}^{i=m-1} \alpha_i = 0$. Le degré de v est égal à 2 dans tous les graphes pris auparavant pour construire $(z_i)_{1 \leq i \leq m-2}$. Dans le dernier cas ($i = m - 1$), le degré de v est de 1. Ceci implique que $\alpha_{m-1} = 0$. Maintenant, on s'intéresse aux $m - 2$ vecteurs. Le degré de chaque sommet w est toujours égal à 2 sauf dans un cas (quand l'arête (w, u) est ajoutée). Par conséquent on a $\alpha_i = 0$ quand $m - n + 1 \leq i \leq m - 2$. En utilisant le fait que les $m - n$ premiers vecteurs sont affinement indépendants, on en déduit que $\alpha_i = 0$ quand $i \leq m - n$. Observons que la face induite par $x_e = 0$ ne contient pas tout le polytope. Il est aussi évident que $x_e \geq 0$ n'induit pas une facette quand $n \leq 4$. Ceci est alors suffisant pour finir la preuve. \square

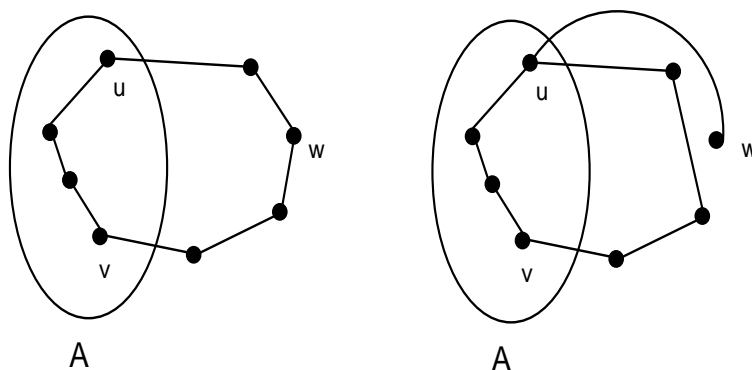


FIGURE 2.2 – Sur la preuve de la Proposition (2.9)

Proposition 2.9: L'inégalité (2.2) définit une facette de $P(G, p)$ si et seulement si $n \geq 4$ et $2 \leq |A| \leq \min(n - 2, p)$.

Preuve: Si $p = n - 1$, le résultat est une conséquence de la structure faciale du polytope de voyageur de commerce symétrique [59].

On considère le cas où $p \leq n - 2$. Soit A tout sous-ensemble de V tel que $2 \leq |A| \leq p \leq n - 2$. On peut construire $m - n$ vecteurs affinement indépendants z_1, z_2, \dots, z_{m-n} qui représentent les tournées Hamiltoniennes tel que $z_i(A) = |A| - 1$. Ceci est possible car on sait que $x(A) \leq |A| - 1$ induit une facette du polytope du voyageur de commerce symétrique. Considérons une de ces tournées. Elle contient un chemin de longueur $|A| - 1$ traversant les sommets de A . Soient u et v les deux extrémités de ce chemin. Pour tout sommet $w \in V \setminus \{u, v\}$, on construit un nouveau graphe dans \mathcal{H}_p en supprimant les deux arêtes incidentes qui joignent ses deux voisins et en ajoutant l'arête (u, w) (Figure (2.2)). Dans ce cas, on obtient $n - 2$ nouveaux vecteurs $z_{m-n+1}, \dots, z_{m-2}$ de $P(G, p)$ qui satisfont $x(A) = |A| - 1$. On construit un vecteur en plus en prenant $w = v$ et répéter les mêmes opérations.

Supposons qu'il existe des nombres $(\alpha_i)_{1 \leq i \leq m-1}$ tel que $\sum_{i=1}^{m-1} \alpha_i z_i = 0$ et $\sum_{i=1}^{m-1} \alpha_i = 0$. Il est clair que le degré de $w \in V \setminus \{u, v\}$ est toujours égal à 2 sauf dans un cas. Ceci implique que $\alpha_i = 0$ quand $m - n + 1 \leq i \leq m - 2$. Maintenant et en utilisant le fait que le degré de v est égal à 2 dans les premiers $m - n$ cas et égal à 1 dans le dernier cas $i = m - 1$, on en déduit que $\alpha_{m-1} = 0$. Comme les $m - n$ tournées Hamiltoniennes sont affinement indépendantes, ceci alors termine la preuve. \square

Notons que les inégalités $x_e \leq 1$ représentent un cas spécial des inégalités (2.2).

D'autres facettes de $P(G, p)$ sont induites par la formulation introduite dans la Section 3.2. Considérons l'inégalité

$$x(A) \leq |A|, A \subseteq V. \quad (2.10)$$

Il est clair que cette inégalité est valide pour $P(G, p)$ pour toute valeur de p . Comme $P(G, p = 2)$ est la projection de la formulation du couplage introduite auparavant, on n'a pas besoin de générer les inégalités (2.10). Et comme on est intéressé par la structure faciale de $P(G, p)$, on donne alors ci-dessous un résultat sur la face induite par l'inégalité (2.10).

Proposition 2.10: L'inégalité (2.10) induit une facette de $P(G, p)$ si et seulement si $p + 2 \leq |A| \leq n - 1$.

Preuve: Si $|A| \leq p$, alors l'inégalité (2.10) est dominée par l'inégalité (2.2). Il est aussi clair que le cas où $|A| = n$ n'induit pas une facette comme tous les vecteurs de $P(G, p)$ satisfont $x(V) = n$. En outre, si $|A| = p + 1$, alors tout vecteur x de la face induite par l'inégalité (2.10) satisfait $x(v, A) = 2$ où $v \in A$.

Concentrons nous maintenant sur le cas $p + 2 \leq |A| \leq n - 1$. Soit $\alpha x = \beta$ une égalité satisfaite par tous les vecteurs de $P(G, p)$ pour laquelle $x(A) = |A|$. Considérons un sous-graphe de G contenant un cycle de taille $|A| - 1$ passant par $|A| - 1$ sommets de A et supposons que tous les autres vecteurs de V sont liés à ce cycle par une arête. Le vecteur d'incidence de ce graphe appartient à la face induite par $x(A) = |A|$. Soit u l'unique sommet de A qui n'appartient pas au cycle et soit v son voisin sur le cycle. Soit w l'un des deux voisins de v sur le cycle. Il est clair qu'on peut supprimer l'arête (w, v) et ajouter l'arête (w, u) pour obtenir un nouveau vecteur d'incidence qui satisfait toutes les contraintes. Ceci implique alors que $\alpha_{wv} = \alpha_{wu}$. Comme u, v et w sont trois sommets arbitraires de A , on en déduit alors que α_{uv} est égal à une constante α_A pour tous deux sommets u et v de A . Considérons maintenant un sommet quelconque $b \in V \setminus A$. b peut être arbitrairement lié à un sommet quelconque de A . Ceci implique que α_{ba} est égal à une constante α_{bA} pour tout sommet $b \in V \setminus A$ et $a \in A$. Si $|A| = n - 1$, ceci est alors suffisant de déduire que l'égalité $\alpha x = \beta$ est une combinaison de $x(V) = n$ et $x(A) = |A|$. Quand $|A| \leq n - 2$, on peut considérer un autre sommet $c \in V \setminus A$. Supposons que c est lié à $a \in A$. Il est clair qu'on obtient un graphe qui satisfait toutes les contraintes si on supprime (c, a) et on ajoute (c, b) où

b est un quelconque autre sommet de $V \setminus A$. Par conséquent, $\alpha_{cb} = \alpha_{ca} = \alpha_{cA}$. Comme b et c sont arbitraires, on obtient que les deux termes α_{cb} et α_{ca} sont égaux à une constante α_{BA} . Autrement dit, $\alpha x = \beta$ est encore une combinaison linéaire de $x(V) = n$ et $x(A) = |A|$. \square

Les inégalités (2.10) avec les inégalités triviales $x_e \geq 0$ et $x_e \leq 1$ décrivent complètement $P(G, p = 2)$.

Proposition 2.11:

$$P(G, 2) = \{x \in \mathbb{R}^E; 0 \leq x_e \leq 1, \forall e \in E; x(A) \leq |A|, \forall A \subset V, 4 \leq |A| \leq n - 1; x(V) = n\}.$$

Preuve: On a déjà mentionné que les graphes de $G = (V, E)$ tels que chaque composante connexe contient au plus un cycle est un matroïde. Ceci implique que $P(G, 2)$ est défini par les contraintes de positivité, les contraintes $x(V) = n$, ainsi que les contraintes générales $x(B) \leq r(B)$ où $B \subseteq E$ et r est la fonction rang associée au matroïde. Soit $B' \subseteq B$ tel que $|B'| = r(B)$ et B' appartient au matroïde. Le graphe $G' = (V, B')$ induit par B' doit avoir quelques composantes connexes avec exactement un cycle, et d'autres sans cycle. Soit C l'union de toutes les composantes de B' qui contiennent un cycle. Par la maximisation de B' , toutes les arêtes de $B \setminus B'$ ont les deux extrémités dans C . Soit A l'ensemble des sommets de C . Il est clair que par l'ajout des contraintes $x(A) \leq |A|$ et les contraintes $x_e \leq 1, e \in B' \setminus C$ on obtient une contrainte qui est au moins meilleure que $x(B) \leq r(B)$. \square

Étudions maintenant les inégalités (2.4). Il est clair que si on pose $|A| = 1$, alors l'inégalité (2.4) est une combinaison des inégalités triviales $x_e \leq 1$. Le cas où $|B| = 0$ est aussi inclus dans les inégalités (2.2). Alors, on supposera que $|A| \geq 2$ et $|B| \geq 1$. Ceci implique directement que $p \geq 3$ (comme $|A| \leq p - 1$) et $n \geq 4$.

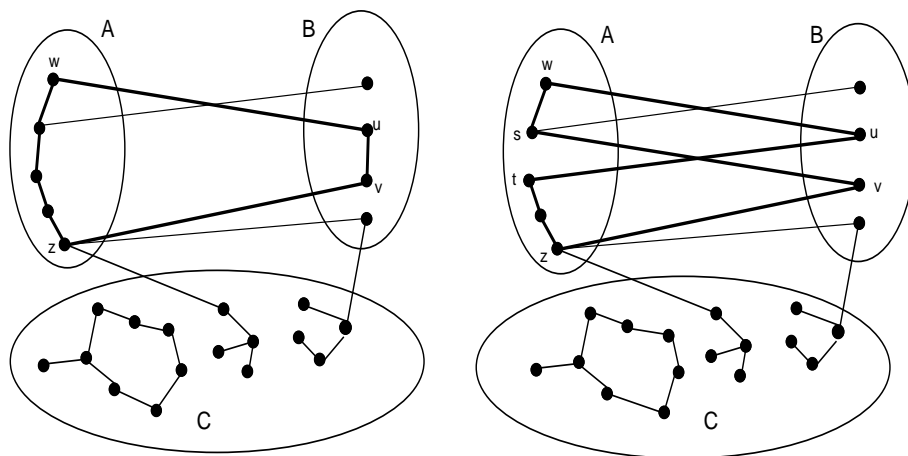


FIGURE 2.3 – Saturation de l'inégalité (2.4) : cas 1 et 2

Proposition 2.12: L'inégalité (2.4) définit une facette de $P(G, p)$ si et seulement si $|A| = p - 1, 3 \leq p \leq n - 2$ et $|B| \geq 3$.

Preuve: On utilise ici la notation introduite dans la preuve de la Proposition (2.1). On rappelle que l'inégalité (2.4) est équivalente à $x(A) \leq |A| - 2 + x(B) + x(C) + x(\delta(C)) - |C|$. Pour avoir l'égalité, Il y a seulement trois cas possibles :

- 1. $x(A) = |A| - 1, x(B) = 1$ et $x(C) + x(\delta(C)) - |C| = 0$
- 2. $x(A) = |A| - 2, x(B) = 0$ et $x(C) + x(\delta(C)) - |C| = 0$

$$- 3. x(A) = |A| - 1, x(B) = 0 \text{ et } x(C) + x(\delta(C)) - |C| = 1.$$

Dans les deux premiers cas, on a $x(C) + x(\delta(C)) - |C| = 0$. Ceci implique que chaque composante connexe de C qui ne contient pas de cycle est nécessairement connectée à $A \cup B$ par exactement une arête. Si la composante contenait déjà un cycle, alors il n'y a pas d'arêtes entre $A \cup B$ et cette composante connexe. Ceci implique que le graphe induit par $A \cup B$ contient au moins un cycle. En utilisant le fait que $|A| \leq p - 1$, il est alors facile de voir que dans les deux cas 1 et 2, la cardinalité de A doit être égale à $p - 1$. Dans le cas 1, A contient un chemin élémentaire de longueur $p - 2$. Un cycle de longueur $p + 1$ contient ce chemin avec deux arêtes liant A et B (voir Figure (2.3)).

Dans le cas 2, A contient deux chemins disjoints (au sens des sommets) et de longueur totale $p - 3$. Un cycle de longueur $p + 1$ contient ces deux chemins avec 4 arêtes liant A et B (voir Figure (2.3)).

Maintenant, on observe que si $|B| = 1$, alors les cas 1 et 2 ne peuvent pas se réaliser. Ceci veut dire qu'on a $x(A) = |A| - 1, x(B) = 0$ et $x(C) + x(\delta(C)) - |C| = 1$. Comme $x(A) \leq |A| - 1$ définit une facette (Proposition (2.9)) et qui est saturée ici, l'inégalité (2.4) ne définit pas une facette quand $|B| = 1$. En utilisant le même argument on déduit que l'inégalité (2.4) ne définit pas une facette si $|A| < p - 1$ (dans les cas 1 et 2, on doit avoir $|A| = p - 1$).

Supposons alors que $|B| \geq 2$ et $|A| = p - 1$. Si $p = n - 1$, alors on obtient $|A| = n - 2, |B| = 2$ et $|C| = 0$. L'inégalité (2.4) est alors dominée par les inégalités $x(\delta(v)) \leq 2$ pour tout $v \in A$ (valide pour $p = n - 1$). Autrement dit, quand $p = n - 1$ notre inégalité ne définit pas une facette. Le cas $p = 2$ (avec $|A| = p - 1$) peut aussi être éliminé car l'inégalité (2.4) est dominée par les inégalités triviales $x_e \leq 1$.

Concentrons nous alors sur le cas $|B| \geq 2, |A| = p - 1$ et $3 \leq p \leq n - 2$. On étudie d'abord le cas élémentaire où $|B| = 2$. On suppose que u et v sont les deux sommets de B . Il n'est pas difficile de voir que dans les trois cas énumérés dans le début de la preuve, on a $x(u, A) - x(v, A) = 0$. Ceci veut dire que tous les vecteurs appartenant à la face induite par l'inégalité (2.4) sont aussi contenus dans un autre hyperplan. Comme l'hyper-plan $x(u, A) - x(v, A) = 0$ est linéairement indépendant avec l'hyper-plan $x(E) = n$ et $2x(A) + x(A, B) = 2|A| + |B| - 2$, on en déduit alors que l'inégalité (2.4) ne définit pas une facette de $P(G, p)$ quand $|B| = 2$.

Dans ce qui va suivre, on montre qu'on obtient une facette quand $|B| \geq 3, |A| = p - 1$ et $3 \leq p \leq n - 2$. Soit $\alpha x = \beta$ une égalité satisfaite par tous les vecteurs de $P(G, p)$ pour lesquels $2x(A) + x(A, B) = 2|A| + |B| - 2$.

D'abord, on suppose que $|C| = 0$. Ceci implique qu'uniquement les cas 1 et 2 peuvent se réaliser. Considérons un graphe de \mathcal{H}_p qui correspond au cas 1. Soit (u, v) l'unique arête du graphe qui a les deux extrémités dans B et soient w et z les deux extrémités du chemin contenant A (Figure (2.3)). On peut supposer que tous les autres sommets de B (sauf u) sont connectés à z . Maintenant, si on supprime l'arête (u, v) et on la remplace par une autre arête (u, v') ($v' \in B$), on obtient un autre graphe qui satisfait toutes les conditions requises avant. Ceci implique que $\alpha_{uv} = \alpha_{uv'}$. Comme u, v et v' sont des sommets choisis arbitrairement, on en déduit alors que α_{uv} est égal à une constante α_B pour toute paire de sommets u et v de B .

Considérons un sommet quelconque $v' \in B \setminus \{u, v\}$. Il est clair que v' peut être connecté à un sommet quelconque y de A . Ceci veut dire que $\alpha_{v'y}$ dépend uniquement de v' . Comme u, v et v' sont choisis arbitrairement, on déduit alors que pour tout sommet $v' \in B$ et pour tout $y \in A$, $\alpha_{v'y}$ est égal à une constante $\alpha_{v'A}$.

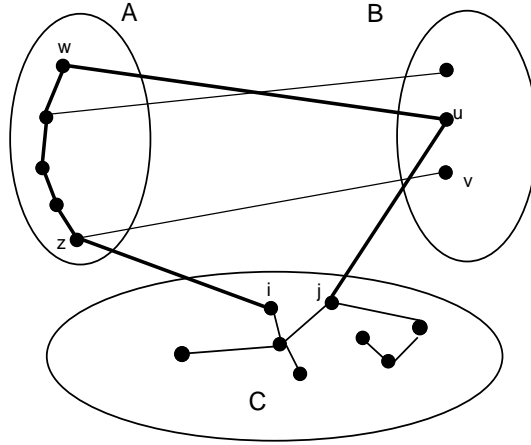


FIGURE 2.4 – Saturation de l'inégalité (2.4) : cas 3

Considérons un graphe de \mathcal{H}_p qui correspond au cas 2. On définit u, v, w, z, s et t comme le montre la Figure (2.3) (certains parmi eux peuvent être identiques). Il est clair que si on ajoute l'arête (s, t) , on supprime une quelconque autre arête de A et connecter u et v aux nouveaux sommets s et t , on obtient un autre graphe qui satisfait toutes les conditions requises auparavant. En utilisant le fait que α_{ut} et α_{vs} ne dépendent pas de s et t , on déduit que α_{st} est aussi une constante : $\alpha_{st} = \alpha_A$. Notons que la transformation de graphe décrite ici suppose implicitement que $|A| \geq 3$. Toutefois, si $|A| = 2$ on peut toujours dire que $\alpha_{st} = \alpha_A$ comme il y a qu'une seule arête induite par A .

En comparant les deux graphes introduits ci-dessus (cas 1 et cas 2), on déduit directement que $\alpha_A + \alpha_B = \alpha_{uA} + \alpha_{vA}$. Comme ceci se réalise pour tous u et v de B et $|B| \geq 3$, on obtient que α_{uA} est aussi égal à une constante α_{AB} qui ne dépend pas de u . On a aussi $\alpha_A + \alpha_B = 2\alpha_{AB}$. Par conséquent, l'égalité $\alpha x = \beta$ est équivalente à $\alpha_A x(A) + \alpha_B x(B) + \alpha_{AB} x(A, B) = \beta$. Elle peut s'écrire ainsi : $\alpha_B (x(B) + x(A) + x(A, B)) + (\alpha_{AB} - \alpha_B)(2x(A) + x(A, B)) = \beta$. Ceci met alors fin à la première partie de la preuve.

Étudions maintenant le cas $|C| \geq 1$. On a toujours $|B| \geq 3$, $|A| = p - 1$ et $3 \leq p \leq n - 2$. On montre exactement par la même façon que $\alpha_{uv} = \alpha_B$ ($u \in B$ et $v \in B$) et $\alpha_{st} = \alpha_A$ ($s \in A$ et $t \in A$). On obtient aussi facilement que α_{iy} est égal à une constante α_{AB} pour tous $u \in B$ et $y \in A$. L'égalité $\alpha_A + \alpha_B = 2\alpha_{AB}$ est réalisée.

Pour terminer la preuve, on doit introduire un troisième graphe. On considère un graphe qui correspond au cas 3. Il est défini ainsi : le graphe induit par C est un arbre, il existe une arête (z, i) qui relie A et C , une arête (j, u) liant B et C , le graphe induit par A est un chemin dont les extrémités sont w et z , u est connecté à w , chaque sommet de B est connecté à exactement un sommet de A (Figure (2.4)).

Comme i est un sommet arbitraire de C , α_{zi} ne dépend pas de i . En plus, z est aussi quelconque (on peut considérer un autre chemin de A avec d'autres extrémités). Par conséquent, α_{zi} est égal à une constante α_{AC} .

En utilisant le fait que l'arbre induit par C est quelconque, on peut facilement déduire que les coefficients α_{kl} sont égaux à une constante α_C pour tous sommets k et l de C .

Considérons maintenant un sommet $v \in B$ et supposons que v est connecté à z (Figure (2.4)).

On peut supprimer l'arête (j, u) et ajouter l'arête (u, v) pour obtenir un nouveau graphe (dont le type correspond au cas 1). En comparant ces deux graphes, on déduit que $\alpha_{ju} = \alpha_{uv} = \alpha_B$. Autrement dit, α_{ju} est égal à une constante $\alpha_{CB} = \alpha_B$ pour tous sommets $j \in C$ et $u \in B$. Au lieu d'éliminer l'arête (j, u) , on élimine (i, z) (on ajoute (u, v)). Ceci implique que $\alpha_{AC} = \alpha_B$. Observons que si $|C| \geq 2$, alors on peut supprimer une arête (séparant i et j) dont les deux extrémités sont dans C pour obtenir deux composantes connexes, et on ajoute l'arête (u, v) . Le nouveau graphe satisfait toujours toutes les conditions requises avant. Ceci implique clairement que $\alpha_{ij} = \alpha_C = \alpha_B$ pour tous sommets $i \in C$ et $j \in C$.

En combinant tous ces résultats, on trouve $\alpha_x = \beta$ est équivalent à $(\alpha_{AB} - \alpha_B)(2x(A) + x(A, B)) + \alpha_B(x(B) + x(C) + x(\delta(C) + x(A) + x(A, B))) = \beta$. L'inégalité (2.4) définit une facette de $P(G, p)$. \square

Proposition 2.13: Soient A et B deux sous-ensembles disjoints de V et soit $C = V \setminus \{A \cup B\}$. L'inégalité (2.5) définit une facette de $P(G, p)$ si et seulement si $3 \leq p \leq n - 2$ et un des cas suivants se réalise :

- $|B| = 0, 1 \leq |C| \leq p - 1$
- $1 \leq |B| \leq p - 4, |C| \geq 3$
- $|B| = p - 3, |C| \geq 4, p \geq 4$
- $|B| = p - 2, |A| \leq 1, |C| \geq 4$
- $|B| = p - 1, |A| = 0$.

Preuve: Il est clair que lorsque $p = n - 1$, l'inégalité (2.5) est dominée par les égalités triviales $x(\delta(v)) = 2$. On supposera alors que $p \leq n - 2$.

Soit G un graphe de \mathcal{H}_p pour lequel l'égalité suivante est vérifiée :

$$\sum_{v \in A} (2 - x(\delta(v))) + \sum_{v \in B} (x(\delta(v) \setminus \delta(A)) - 2) = n - (p + 1). \quad (2.11)$$

Si un tel graphe n'existe pas, alors l'inégalité (2.5) ne définit pas une facette de $P(G, p)$. Soit $A_1 \subseteq A$ l'ensemble des sommets v de A pour lesquels $x(\delta(v)) = 1$. Tous les sommets de $A_2 = A \setminus A_1$ ont un degré égal à 2 (sinon les membres gauches de l'équation (2.11) peuvent être augmentés en éliminant des sommets dont le degré est supérieur à 2). L'égalité (2.11) devient $\sum_{v \in B} (x(\delta(v) \setminus \delta(A)) - 2) = n - |A_1| - (p + 1)$. En outre, le graphe obtenu en éliminant tous les sommets de A_1 reste toujours dans \mathcal{H}_p . Alors écrire l'inégalité (2.5) pour ce nouveau sous-graphe nous donne $\sum_{v \in B} (x(\delta(v) \setminus \delta(A_1)) - 2) \leq n - |A_1| - (p + 1)$. Ceci implique directement que $x(\delta(v) \setminus \delta(A_1)) = x(\delta(v) \setminus \delta(A))$ pour tout $v \in B$. Ceci dit, on a $x(v, A_2) = 0$ pour tous les autres sommets de B . On peut aussi déduire que $x(\delta(v) \setminus \delta(A_1)) - 2 \leq 0$ pour tout $v \in C$ (sinon la somme $\sum_{v \in B} (x(\delta(v) \setminus \delta(A_1)) - 2)$ peut être augmentée en mettant v dans B). L'égalité précédente devient $\sum_{v \in B} (x(\delta(v) \setminus \delta(A_1)) - 2) = n - |A_1| - (p + 1)$. En utilisant le fait que $\sum_{v \in B} (x(\delta(v) \setminus \delta(A_1)) - 2) + \sum_{v \in A_2 \cup C} (x(\delta(v) \setminus \delta(A_1)) - 2) = 0$, on déduit que

$$\sum_{v \in A_2 \cup C} (2 - x(\delta(v) \setminus \delta(A_1))) = n - |A_1| - (p + 1).$$

En utilisant le fait que les sommets de A_2 qui ont le degré 2 et tous les sommets de C sont tels que $x(\delta(v) \setminus \delta(A_1)) - 2 \leq 0$, l'égalité précédente implique que $A_2 \cup C$ contient exactement $n - |A_1| - (p + 1)$ sommets de degré 1 alors que le reste des sommets ont un degré de 2 dans

$G \setminus A_1$. De plus, on sait que $G \setminus A_1$ contient au moins $p + 1$ sommets de degré au moins 2. Ceci implique que les sommets de B sont un unique cycle de longueur $p + 1$. Chaque sommet de $A_2 \cup C$ dont le degré est 1 dans $G \setminus A_1$ est nécessairement lié à un sommet de B . Par conséquent, tous les sommets de A_2 sont aussi sur le cycle (on rappelle que $x(v, A_2) = 0$ si $v \in B$). La Figure (2.5) synthétise la situation.

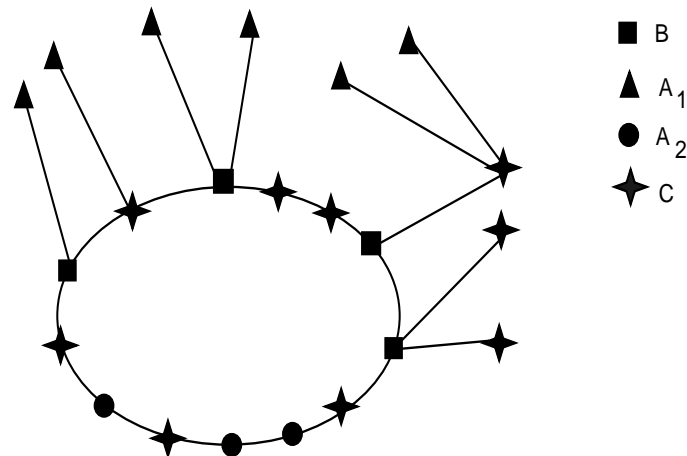


FIGURE 2.5 – Saturation de l'inégalité (2.5)

Il n'est pas difficile de voir que ce qui a été établi dans la Proposition (2.13) est équivalent à ce qui suit : l'inégalité (2.5) définit une facette de $P(G, p)$ si et seulement si : $3 \leq p \leq n - 2$ et une des conditions suivante soit vérifiée :

- $|B| = 0, 1 \leq |C| \leq p - 1$
- $|A| = 0, 1 \leq |B| \leq p - 1$
- $|A| = 1, 1 \leq |B| \leq p - 2, |C| \geq 4$
- $|A| \geq 2, 1 \leq |B| = p - 3, |C| \geq 4$
- $|A| \geq 2, 1 \leq |B| \leq p - 4, |C| \geq 3.$

Le reste de la preuve est relatif à ces cas.

Supposons que $|B| = 0$, alors tous les sommets de C sont sur le cycle et $|A_1| = n - (p + 1)$ (i.e, $|A_2| + |C| = p + 1$).

Si $|A_2| \leq 1$ (i.e, $n - (p + 1) \leq |A| \leq n - (p + 1) + 1$), alors l'égalité $x(A) = 0$ est toujours vérifiée. Cependant, $x(A) = 0$ n'est pas une combinaison linéaire de l'égalité $x(E) = n$ et de l'égalité (2.11). Par conséquent, l'inégalité (2.5) ne définit pas une facette dans ce cas ($|B| = 0, n - (p + 1) \leq |A| \leq n - (p + 1) + 1$).

Il est aussi clair que tout graphe qui sature l'égalité (2.11) ne peut pas avoir $|A_2| = p + 1$ car A_1 et lié seulement aux sommets de $B \cup C$. Donc on doit avoir $|A| \leq n - 1$.

Si on suppose que $|A_2| = 2$ et $|C| = 1$ (ce qui implique que $p = 2$), alors le nombre de graphes qui saturent l'égalité (2.11) est égal à $\frac{(n-1)(n-2)}{2}$ qui est inférieur à la dimension de $P(G, p)$. Ceci conduit à dire que l'inégalité (2.5) ne définit pas une facette dans ce cas.

Supposons maintenant que $|C| = 1$ et $|A_2| \geq 3$ (i.e, $|A| = n - 1$ et $p \geq 3$). Soit $\alpha x = \beta$ une égalité quelconque saturée par tous les points de la face induite par l'égalité (2.11). D'une façon similaire à la preuve de la précédente Proposition, on peut facilement montrer que α_{uv} est égal à une constante α_A pour toute paire de sommets u et v appartenant à A . On peut aussi

montrer que α_{uv} est une constante si v est l'unique sommet de C et u est quelconque de A . Autrement dit, $\alpha x = \beta$ est une combinaison de l'égalité $x(E) = n$ (qui se réduit dans ce cas à $x(A) + x(A, C) = n$) et de l'égalité (2.11) (qui se réduit ici à $2x(A) + x(A, C) = n + p - 1$). Par conséquent, l'inégalité (2.5) définit une facette dans ce cas.

On garde ci-dessous $|B| = 0$. Il existe un autre sous-cas à étudier : $2 \leq |A_2| \leq p - 1$ (i.e, $|C| \geq 2$ et $|A_2| \geq 2$). On considère aussi une égalité $\alpha x = \beta$ saturée par tous les graphes de cette face. Comme $|C| \geq 2$, on peut considérer un graphe de \mathcal{H}_p tel que tous les sommets de A_1 soient reliés à $u_1 \in C$ ou à $u_2 \in C$. On suppose aussi qu'il existe un sommet $v \in A_2$ relié à u_1 et à u_2 . Si on supprime l'arête (u_2, v) et on la remplace par une arête (u_2, u) où $u \in A_1$ est un sommet quelconque non relié à u_2 , on obtient un nouveau graphe qui sature l'inégalité (2.5). Ceci implique que $\alpha_{u_2v} = \alpha_{u_2u}$ pour tous sommets $u_2 \in C$, $u \in A$ et $v \in A$. De plus, un sommet de A_1 peut être relié à tout sommet quelconque de C qui implique que α_{u_2v} ne dépend pas de u_2 . Ceci dit, $\alpha_{uw} = \alpha_{AC}$ pour tous sommets $u \in A$ et $w \in C$. Il est aussi facile de voir que lorsque $|C| \geq 3$, alors $\alpha_{uw} = \alpha_C$ pour tous sommets $u \in C$ et $w \in C$ (on considère trois sommets u_1, u_2 et u_3 dans C où u_1 est relié à u_2 et u_2 est à son tour relié à u_3 et on change l'ordre des sommets pour obtenir un nouveau graphe).

Comme $|A_2| \geq 2$ et $|C| \geq 2$, on peut considérer un graphe de \mathcal{H}_p qui sature l'inégalité (2.5) avec deux sommets $v_1 \in A_2$ et $v_2 \in A_2$ tel que v_1 est relié à v_2 . On suppose aussi qu'il existe deux sommets $u_1 \in C$ et $u_2 \in C$ qui sont reliés. Si on change la position de v_2 sur le cycle en l'insérant entre u_1 et u_2 , on obtient un nouveau graphe qui satisfait toutes les conditions requises avant. Ceci implique que $\alpha_{uv} = \alpha_A$ pour tous sommets $u \in A$ et $v \in A$ et conclue aussi que $\alpha_A + \alpha_C = 2\alpha_{AC}$. Ceci est suffisant pour dire que $\alpha x = \beta$ est une combinaison de l'égalité $x(E) = n$ et de l'égalité (2.11).

En combinant les résultats ci-dessus, on obtient : si $|B| = 0$, alors l'inégalité (2.5) définit une facette de $P(G, p)$ si et seulement si $3 \leq p \leq n - 2$, et $n - (p - 1) \leq |A| \leq n - 1$.

En utilisant le fait que $\sum_{v \in B} (x(\delta(v)) - 2) = \sum_{v \in V \setminus B} (2 - x(\delta(v)))$, le résultat obtenu pour $|B| = 0$ nous conduit directement à une caractérisation similaire de facette que pour le cas $|A| = 0$. Si $|A| = 0$, alors l'inégalité (2.5) définit une facette de $P(G, p)$ si et seulement si $3 \leq p \leq n - 2$, et $1 \leq |B| \leq p - 1$.

On suppose maintenant que $|A| \geq 1$ et $|B| \geq 1$.

On rappelle que les sommets de B sont sur l'unique cycle de taille $p + 1$. Ceci implique que $|B| \leq p + 1$. Il est aussi clair que $|B| \geq p - 1$, alors en utilisant $x(B, A_2) = 0$ on obtient que tout graphe qui sature l'inégalité (2.5) doit avoir $|A_2| = 0$. Ce qui implique qu'on a $x(\delta(v)) = 1$ pour tout sommet v de A . Par conséquent, l'inégalité (2.5) ne définit pas une facette dans ce cas. On note que ceci implique aussi qu'on n'a pas de facette dans le cas où $p = 2$.

On se concentre alors sur le cas où $3 \leq p \leq n - 2$, $|A| \geq 1$ et $1 \leq |B| \leq p - 2$.

Comme $|B| \leq p - 2$, le cycle de taille $p + 1$ doit contenir au moins 3 sommets de C si $A_2 = \emptyset$, et au moins 2 sommets de C si $A_2 \neq \emptyset$ (parce que $x(B, A_2) = 0$). De plus, si $|C| = 2$, alors on a $x(C) = 0$ pour tout vecteur appartenant à la face induite par l'inégalité (2.5)). Comme cette égalité n'est pas une combinaison linéaire de l'égalité $x(E) = n$ et de l'égalité (2.11), on en déduit que la face n'est pas une facette. On supposera alors que $|C| \geq 3$.

Étudions le cas où $|A| = 1$ (avec $3 \leq p \leq n - 2$, $1 \leq |B| \leq p - 2$ et $|C| \geq 3$). On commencera avec le sous-cas $|C| \geq 4$. Considérons un graphe de \mathcal{H}_p qui sature l'inégalité (2.5). Soit a l'unique sommet de A . Comme les sommets de A peuvent être reliés à n'importe quel sommet

de $B \cup C$, on déduit que α_{av} est égal à une constante $\alpha_{AB} = \alpha_{AC}$ pour tout sommet v appartenant soit à B ou à C . Le même genre d'arguments implique que $\alpha_{uv} = \alpha_{uB}$ pour tout sommet $u \in C$ et $v \in B$.

Comme $p \leq n - 2$, on peut supposer que a appartient au cycle. Soit $u \in C$ et $v \in C$ son voisin et soit $w \in C$ un autre sommet qui n'est pas sur le cycle. En permutant a et w , on déduit que $\alpha_{AC} + \alpha_{wB} = \alpha_{wu} + \alpha_{wv}$. Maintenant, en utilisant le fait que $|C| \geq 4$ (v peut être remplacé par un autre sommet de C) on déduit que $\alpha_{vu} = \alpha_C$ pour tous sommets $u \in C$ et $v \in C$. Combiné avec l'égalité précédente, ceci nous conduit à $\alpha_{wB} = \alpha_{CB}$ pour tout $w \in C$. En d'autres mots, on a montré que $\alpha_{AC} + \alpha_{CB} = 2\alpha_C$. Si $|B| = 1$, alors les égalités $\alpha_{AB} = \alpha_{AC}$ et $\alpha_{AC} + \alpha_{CB} = 2\alpha_C$ sont suffisantes pour montrer que $\alpha x = \beta$ est une combinaison linéaire de l'égalité $x(E) = n$ ainsi que de l'égalité (2.11). Par conséquent, on obtient une facette si $|A| = 1$, $3 \leq p \leq n - 2$, $1 = |B| \leq p - 2$ et $|C| \geq 4$.

En outre, si $|B| \geq 2$, alors on considère un graphe de \mathcal{H}_p qui sature l'inégalité (2.5) où le cycle contient des sommets adjacents de C (u et v) et deux autres sommets adjacents de B (i et j) et on suppose que i et u sont aussi adjacents. En supprimant les arêtes (u, v) et (i, j) et en ajoutant les arêtes (u, j) et (i, v) on obtient un nouveau graphe qui satisfait toutes les conditions initiales. Ceci implique que $\alpha_C + \alpha_{ij} = 2\alpha_{CB}$. Comme i et j sont quelconques, on obtient α_{ij} qui est égal à une constante α_B pour tout sommet de B . On obtient aussi $\alpha_C + \alpha_B = 2\alpha_{CB}$. En combinant cette égalité avec les égalités $\alpha_{AB} = \alpha_{AC}$ et $\alpha_{AC} + \alpha_{CB} = 2\alpha_C$, on déduit que $\alpha x = \beta$ est une combinaison linéaire de l'égalité $x(E) = n$ et l'égalité (2.11). Par conséquent, on obtient une facette si $|A| = 1$, $3 \leq p \leq n - 2$, $1 \leq |B| \leq p - 2$ et $|C| \geq 4$.

Le second cas est donné par $|A| = 1$, $3 \leq p \leq n - 2$, $1 \leq |B| \leq p - 2$ et $|C| = 3$. On a $p = n - 2$, et $C = \{u, v, w\}$. En considérant tous les cas possibles, il est facile de vérifier qu'on a toujours $x(A, B) + x(A, C) + x(B) + x(u, w) + \frac{1}{2}x(u, v) + \frac{3}{2}x(w, v) + \frac{1}{2}x(u, B) + x(v, B) + \frac{3}{2}x(w, B) = |B| + 4$. Ceci veut dire que l'inégalité (2.5) ne déduit pas une facette de $P(G, p)$ quand $|A| = 1$, $3 \leq p \leq n - 2$, $2 \leq |B| \leq p - 2$ et $|C| = 3$. Le cas $|A| = 1$ est maintenant complètement étudié.

On s'intéresse maintenant au cas $|A| \geq 2$ (avec $3 \leq p \leq n - 2$, $1 \leq |B| \leq p - 2$ et $|C| \geq 3$). Si on considère le sous-cas $|B| = p - 2$, alors tout graphe de \mathcal{H}_p qui sature l'inégalité (2.5) satisfait $x(A) = 0$. Comme cette égalité n'est pas une combinaison linéaire de $x(E) = n$ et de l'égalité (2.11), on déduit alors qu'on n'a pas de facette dans ce sous-cas.

Le deuxième sous-cas est donné par $|A| \geq 2$, $3 \leq p \leq n - 2$, $1 \leq |B| \leq p - 3$ et $|C| \geq 3$. Comme tout sommet de A_1 peut être relié à tout sommet de $B \cup C$, on obtient que α_{ax} est égal à une constante α_{aB} pour tous les sommets $x \in B \cup C$ et $a \in A$. On obtient aussi par la même façon que $\alpha_{cx} = \alpha_{cB}$ pour tous les sommets $c \in C$ et $x \in B$.

Considérons un graphe de \mathcal{H}_p qui sature l'inégalité (2.5) où il existe deux sommets adjacents $u \in A$ et $v \in A$ appartenant au cycle et un sommet $j \in C$ adjacent à v et appartenant au cycle. On peut aussi supposer qu'il existe au moins un sommet $i \in C$ qui n'appartient pas au cycle. Par la suppression de l'arête (u, v) et l'arête qui relie i à B , et en ajoutant les arêtes (u, i) et (i, j) , on obtient un nouveau graphe qui satisfait toutes les conditions requises initialement. Ceci implique que $\alpha_{iB} + \alpha_{uv} = \alpha_{uC} + \alpha_{ij}$. Comme $|C| \geq 3$ et j est quelconque, on déduit de l'égalité précédente que α_{ij} est aussi égal à une constante α_C . On déduit aussi que $\alpha_{uv} = \alpha_A$, $\alpha_{iB} = \alpha_{CB}$ et $\alpha_{uC} = \alpha_{AC}$. Les égalités ci-dessus deviennent $\alpha_{BC} + \alpha_A = \alpha_{AC} + \alpha_C$ et $\alpha_{AB} = \alpha_{AC}$.

Considérons maintenant le sous-cas $|B| = p - 3$. Il est facile de vérifier que si $|C| = 3$ (et $|A| \geq 2$), alors on a $x(A) + x(A, B) + x(A, C) = |A| + 1$ impliquant que la face n'est pas une

facette.

En outre, si $|C| \geq 4$ (avec $1 \leq |B| = p - 3$ et $|A| \geq 2$), alors on peut considérer un graphe de \mathcal{H}_p qui sature l'inégalité (2.5) contenant 4 sommets de C sur le cycle $(u, v, w$ et $t)$. On suppose que u est adjacent à un sommet de B et à v . On suppose aussi que les arêtes (v, w) et (w, t) existent et il existe un sommet $a \in A$ relié à u . On modifie le graphe comme suit : on supprime les arêtes (v, u) et (v, w) , et on ajoute l'arête (a, w) et on connecte v à B . Ceci nous conduit à $\alpha_{BC} + \alpha_{AC} = 2\alpha_C$. Si $|B| = 1$, cette égalité combinée avec les égalités ci-dessus ($\alpha_{BC} + \alpha_A = \alpha_{AC} + \alpha_C$ et $\alpha_{AB} = \alpha_{AC}$) est suffisante pour déduire que $\alpha x = \beta$ est une combinaison linéaire de $x(E) = n$ et de l'égalité (2.11).

Il est aussi facile de voir que si $|B| \geq 2$ alors on considère un graphe de \mathcal{H}_p tel que le cycle contient dans cet ordre b_1, c_1, b_2 et c_2 où b_1 et b_2 appartiennent à B , c_1 et c_2 appartiennent à C et les arêtes suivantes existent : $(b_1, c_1), (c_1, b_2), (b_2, c_2)$. En permutant c_1 et b_2 on déduit que $\alpha_{b_1 b_2} + \alpha_C = 2\alpha_{BC}$. Comme b_1 et b_2 sont quelconques, on déduit que $\alpha_{b_1 b_2}$ est égal à une certaine constante α_B tel que $\alpha_B + \alpha_C = 2\alpha_{BC}$. En combinant cette égalité avec les inégalités précédentes, on trouve qu'on a une facette.

En d'autres termes, on a montré que si $1 \leq |B| = p - 3, |C| \geq 4$ et $|A| \geq 2$, l'inégalité (2.5) induit une facette de $P(G, p)$.

Il y a un dernier cas à étudier : $|A| \geq 2, 3 \leq p \leq n - 2, 1 \leq |B| \leq p - 4$ et $|C| \geq 3$. Le graphe des transformations défini auparavant pour montrer que $\alpha_{BC} + \alpha_A = \alpha_{AC} + \alpha_C$ et $\alpha_{AB} = \alpha_{AC}$ reste valide. Mais l'autre transformation pour montrer que $\alpha_B + \alpha_C = 2\alpha_{BC}$ n'est pas valide comme on peut avoir $|C| = 3$. Toutefois, comme $1 \leq |B| \leq p - 4$, on peut considérer un graphe de \mathcal{H}_p où le cycle contient dans cet ordre les sommets a_1, c_1, a_2 et c_2 ($a_1 \in A, a_2 \in A, c_1 \in C$ et $c_2 \in C$, et les arêtes $(a_1, c_1), (c_1, a_2), (a_2, c_2)$ existent). En permutant c_1 et a_2 , on obtient que $\alpha_A + \alpha_C = 2\alpha_{AC}$. En considérant les cas $|B| = 1$ et $|B| \geq 2$ comme on a déjà fait avant, on conclut que dans tous les cas on obtient une facette. En d'autres mots, quand $|A| \geq 2, 3 \leq p \leq n - 2, 1 \leq |B| \leq p - 4$ et $|C| \geq 3$, l'inégalité (2.5) induit une facette de $P(G, p)$. \square

Finalement, notons que les inégalités (2.6), (2.7), (2.8), et (2.9) ne définissent pas une facette en général. Une étude profonde des dimensions des facettes induites par ces inégalités semble être très compliquée.

2.5 Un Algorithme à Plans Coupants

Un algorithme à plans coupants est implémenté sur la base des inégalités valides de la section précédente. Nous présentons dans la suite les détails des algorithmes de séparation. Plus de détails et des résultats numériques suivront dans la section 2.6.

2.5.1 Séparation des inégalités (2.5)

On montre d'abord que les inégalités (2.5) peuvent être séparées en temps polynomial en minimisant une fonction sous-modulaire. Après, on donne une deuxième procédure de séparation pratique basée sur un algorithme de coupe minimum.

Étant donné un ensemble A , il est clair qu'un ensemble B qui maximise $\sum_{v \in A} (2 - x(\delta(v))) + \sum_{v \in B} (x(\delta(v) \setminus \delta(A)) - 2)$ contiendra que les sommets v tels que $x(\delta(v) \setminus \delta(A)) - 2 \geq 0$. En d'autres mots, maximiser $\sum_{v \in A} (2 - x(\delta(v))) + \sum_{v \in B} (x(\delta(v) \setminus \delta(A)) - 2)$ est équivalent à maximiser

ser $\sum_{v \in A} (x(\delta(v)) - 2) + \sum_{v \in V \setminus A} \min(2 - x(\delta(v) \setminus \delta(A)), 0)$. On définit une fonction $f : 2^V \rightarrow \mathbb{R}$ par

$$f(A) = \sum_{v \in A} (x(\delta(v)) - 2) + \sum_{v \in V \setminus A} \min(2 - x(\delta(v) \setminus \delta(A)), 0).$$

On montre que f est une fonction sous-modulaire. Ceci est clairement suffisant pour déduire que le problème de séparation peut être résolu en temps polynomial, en utilisant par exemple, les algorithmes de [75, 116].

Proposition 2.14: f est sous-modulaire.

Preuve: Soient A_1 et A_2 deux sous-ensembles de V . Il est clair qu'on a :

$$\sum_{v \in A_1 \cup A_2} (x(\delta(v)) - 2) + \sum_{v \in A_1 \cap A_2} (x(\delta(v)) - 2) = \sum_{v \in A_1} (x(\delta(v)) - 2) + \sum_{v \in A_2} (x(\delta(v)) - 2).$$

Alors, on a juste à montrer que :

$$\sum_{v \in V \setminus A_1 \cup A_2} \min(2 - x(\delta(v) \setminus \delta(A_1 \cup A_2)), 0) + \sum_{v \in V \setminus A_1 \cap A_2} \min(2 - x(\delta(v) \setminus \delta(A_1 \cap A_2)), 0) \leq \sum_{v \in V \setminus A_1} \min(2 - x(\delta(v) \setminus \delta(A_1)), 0) + \sum_{v \in V \setminus A_2} \min(2 - x(\delta(v) \setminus \delta(A_2)), 0).$$

On montrera que pour tout sommet v , le membre de gauche de l'inégalité ci-dessus est inférieur au membre de droite.

D'abord, on considère un sommet $v \in V \setminus A_1 \cup A_2$. Alors, v appartient à $V \setminus A_1 \cap A_2$, $V \setminus A_2$ et $V \setminus A_1$. On doit montrer alors que

$$\min(2 - x(\delta(v) \setminus \delta(A_1 \cup A_2)), 0) + \min(2 - x(\delta(v) \setminus \delta(A_1 \cap A_2)), 0) \leq \min(2 - x(\delta(v) \setminus \delta(A_1)), 0) + \min(2 - x(\delta(v) \setminus \delta(A_2)), 0).$$

Si $2 - x(\delta(v) \setminus \delta(A_1 \cup A_2)) \leq 0$, alors $2 - x(\delta(v) \setminus \delta(A_1 \cap A_2)) \leq 0$, $2 - x(\delta(v) \setminus \delta(A_1)) \leq 0$ et $2 - x(\delta(v) \setminus \delta(A_2)) \leq 0$. Le membre de gauche devient $2 - x(\delta(v) \setminus \delta(A_1 \cup A_2)) + 2 - x(\delta(v) \setminus \delta(A_1 \cap A_2))$ et le membre de droite est $2 - x(\delta(v) \setminus \delta(A_1)) + 2 - x(\delta(v) \setminus \delta(A_2))$. Ils sont égaux.

Maintenant, on suppose que $2 - x(\delta(v) \setminus \delta(A_1 \cup A_2)) > 0$. Si $2 - x(\delta(v) \setminus \delta(A_1 \cap A_2))$ est positif alors les autres termes sont aussi positifs : $2 - x(\delta(v) \setminus \delta(A_1)) > 0$ et $2 - x(\delta(v) \setminus \delta(A_2)) > 0$. Dans ce cas les deux membres de gauche et de droite sont égaux à 0.

Si $2 - x(\delta(v) \setminus \delta(A_1 \cup A_2)) > 0$, $2 - x(\delta(v) \setminus \delta(A_1 \cap A_2)) < 0$, $2 - x(\delta(v) \setminus \delta(A_1)) \geq 0$ et $2 - x(\delta(v) \setminus \delta(A_2)) \geq 0$, alors le membre de gauche est $2 - x(\delta(v) \setminus \delta(A_1 \cap A_2))$ alors que celui de droite est 0. L'inégalité est vérifiée.

Un autre sous-cas correspond à $2 - x(\delta(v) \setminus \delta(A_1 \cup A_2)) > 0$, $2 - x(\delta(v) \setminus \delta(A_1 \cap A_2)) < 0$, $2 - x(\delta(v) \setminus \delta(A_1)) < 0$ et $2 - x(\delta(v) \setminus \delta(A_2)) < 0$. Le membre de gauche est $2 - x(\delta(v) \setminus \delta(A_1 \cap A_2))$ et le membre de droite est $2 - x(\delta(v) \setminus \delta(A_1)) + 2 - x(\delta(v) \setminus \delta(A_2))$. La différence entre eux (le membre de droite moins le membre de gauche) est égale à $2 - x(\delta(v) \setminus \delta(A_1 \cup A_2))$ qui est encore positif.

Le dernier sous-cas est donné par $2 - x(\delta(v) \setminus \delta(A_1 \cup A_2)) > 0$, $2 - x(\delta(v) \setminus \delta(A_1 \cap A_2)) < 0$, $2 - x(\delta(v) \setminus \delta(A_1)) < 0$ et $2 - x(\delta(v) \setminus \delta(A_2)) > 0$. Alors, le membre de gauche est $2 - x(\delta(v) \setminus \delta(A_1 \cap A_2))$ et le membre de droite est $2 - x(\delta(v) \setminus \delta(A_1))$. Il est clair que le premier terme est inférieur au second.

On suppose maintenant que $v \in A_1 \setminus A_2$ (le cas $v \in A_1 \cap A_2$ n'est pas vérifié, et le cas $v \in A_2 \setminus A_1$ est symétrique au premier). Dans ce cas, on a juste à montrer que $\min(2 - x(\delta(v) \setminus \delta(A_1 \cap A_2)), 0) \leq \min(2 - x(\delta(v) \setminus \delta(A_2)), 0)$. Ceci est clairement valide.

La sous-modularité de f est donc montrée, ce qui termine la preuve. \square

Les inégalités (2.5) peuvent aussi être séparées par un algorithme de flot maximum. Étant donné un graphe $G = (V, E)$ et un vecteur fractionnaire x , on construit un réseau (un graphe orienté avec capacités sur les arcs) comme suit. Chaque sommet u_i de V est remplacé par deux sommets v_i et w_i . Chaque arête $e = (u_i, u_j) \in E$ est remplacée par deux arcs (w_i, v_j) et (w_j, v_i) . La capacité de chaque arc est x_e . On considère aussi deux nouveaux sommets s et t . Il existe des arcs de s à v_i et w_i avec une capacité de 2. Il y a aussi des arcs de v_i et w_i à t avec une capacité $x(\delta(u_i))$ (voir Figure (2.6)). En calculant un flot maximum de s à t , on obtient une coupe minimum séparant s et t . Les sommets u_i tels que leurs propres v_i et w_i sont du même côté que s vont constituer l'ensemble A . D'une façon similaire, B contient tous les sommets u_i tels que leurs propres v_i et w_i sont du même côté que t . Les autres sommets u_i (pour lesquels leurs w_i et v_i ne sont pas du même côté) vont constituer l'ensemble $C = V \setminus \{A \cup B\}$ (voir Figure (2.6)).

Dans un premier temps, il est clair que tous les sommets $u_i \in C$ sont tels que v_i est du même côté que s et w_i est de l'autre côté. Autrement dit, en permutant v_i et w_i , la capacité de la coupe ne peut pas diminuer. On peut aussi voir que si $u_i \in C$, alors la capacité des arêtes reliées soit à v_i ou à w_i et appartenant à la coupe est exactement $2 + x(\delta(u_i))$. En outre, si $u_i \in B$, alors la capacité des arêtes reliées soit à v_i ou à w_i et appartenant à la coupe est $x(u_i, A) + 4$.

La capacité de la coupe est alors donnée par $\sum_{u_i \in A} 2x(\delta(u_i)) + \sum_{u_i \in C} 2 + x(\delta(u_i)) + \sum_{u_i \in B} 4 + x(u_i, A)$. Comme la capacité de la coupe est minimale, si u_i appartient à C alors $2 + x(\delta(u_i)) \leq x(u_i, A) + 4$ (autrement la capacité peut diminuer en mettant u_i dans B). D'une façon similaire, si $u_i \in B$ alors $x(u_i, A) + 4 \leq 2 + x(\delta(u_i))$. Par conséquent, la capacité de la coupe est égale à $\sum_{u_i \in A} 2x(\delta(u_i)) + \sum_{u_i \in V \setminus A} \min(2 + x(\delta(u_i)), 4 + x(u_i, A))$. Après des calculs algébriques la capacité de la coupe devient $\sum_{u_i \in A} (x(\delta(u_i)) - 2) + 2|A| + \sum_{u_i \in A} x(\delta(u_i)) + \sum_{u_i \in V \setminus A} \min(0, 2 - x(\delta(u_i)) \setminus \delta(A)) + 2|V \setminus A| + \sum_{u_i \in V \setminus A} x(\delta(u_i))$. Donc, la capacité de la coupe est égale à $f(A)$ plus une constante ($4n$).

2.5.2 Séparation des inégalités (2.2)

Considérons le problème de décider s'il existe un ensemble A tel que $|A| \leq p$ et $x(A) - |A| \geq z$ où z et p sont deux nombres donnés. Dans un premier temps, on va prouver que ce problème est NP-Complet. On note que ce problème est plus général que le problème de séparation des inégalités (2.2).

Dans un deuxième temps, on donnera une heuristique de séparation basée sur un algorithme de plus court chemin et une relaxation Lagrangienne où le sous-problème Lagrangien est un problème de coupe minimum.

Considérons le problème de la clique de poids maximum avec contrainte de taille. Il se définit ainsi : étant donné un graphe complet avec des poids associés aux arêtes et deux nombres z et k , décider s'il existe une clique de taille inférieure ou égale à k tel que la somme des poids des arêtes de la clique est supérieure ou égale à z . Ce problème connu dans la littérature est NP-Complet. Il est aussi facile de voir que ce problème reste NP-Complet si on ajoute la contrainte suivante : la clique doit contenir une arête donnée (v_0, v_1) .

En plus, étant donnée une instance de ce problème (clique de poids maximum, de taille inférieure à p , contenant une arête donnée (v_0, v_1)) où $(y_e)_{e \in E}$ sont les poids des arêtes), on

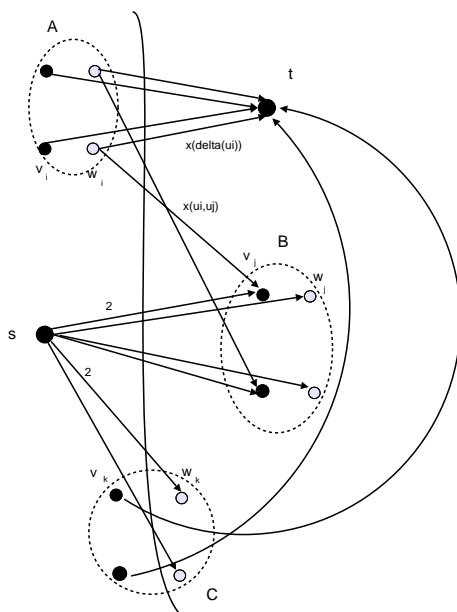


FIGURE 2.6 – Un algorithme de flot maximum pour séparer les inégalités (2.5)

peut construire une instance du problème de séparation comme suit : $x_e = y_e$ si e ne contient pas ni v_0 ni v_1 , $x_e = y_e + 1$ si e contient v_0 (et non v_1), et $x_e = y_e + M + 1$ si $e = (v_0, v_1)$ où M est un grand nombre. Ceci implique que $y(A) + M = x(A) - |A|$ pour tout ensemble A contenant (v_0, v_1) . Si M est suffisamment large, l'existence de A tel que $x(A) - |A| \geq M + z$ et $|A| \leq p$ est équivalente à l'existence d'une clique contenant (v_0, v_1) dont le poids est au moins z .

Toutefois, si p est inférieur à une constante, alors la séparation peut se faire en temps polynomial par simple énumération.

La séparation des inégalités (2.2) est liée au problème de séparation bien connu des inégalités $x(A) \leq |A| - 1$ (sans les contraintes de cardinalité) dans le contexte du polytope de l'arbre couvrant (voir [117]). Une technique simple est utilisée pour séparer ces contraintes en calculant une coupe minimum dans un graphe approprié. En effet, on peut facilement voir que $2|A| - 2x(A) = -\sum_{v \in V} x(\delta(v)) + 2|A| + x(A, V \setminus A) + \sum_{v \in V \setminus A} x(\delta(v))$. Considérons une version pondérée de $G = (V, E)$ où le poids de chaque arête e est x_e . On ajoute à G deux sommets s et t . On connecte s (resp. t) à chaque sommet de V par une arête de poids $x(\delta(v))$ (resp. 2). Alors, minimiser $|A| - x(A)$ est équivalent à calculer une coupe minimum séparant s et t (voir [117]).

Comme x appartient à $P(G, 2)$, on a $|V| - x(V) = 0$ et $|A| - x(A) \geq 0$ pour tout $A \subset V$. Autrement dit, si on minimise $|A| - x(A)$, on peut soit avoir $|A| = n$ ou $|A| = 0$. On rappelle qu'on est intéressé par les sous-ensembles A tels que $2 \leq |A| \leq p$ et $|A| - x(A)$ est inférieur à 1. On peut aussi inclure les ensembles A tels que $|A| = 1$ (dans ce cas $|A| - x(A) = 1$). Pour garantir que $|A| \geq 1$, on peut identifier le sommet s avec un sommet de V (ceci donne n problèmes différents).

Si $p \geq n - q$ où q est une constante, la séparation est alors facile. Ceci vient du fait que $|V \setminus A| \geq n - p$ qui nous conduit à un algorithme trivial d'énumération : pour tout sous-ensemble de sommets de V de cardinalité $n - p \leq q$, on identifie cet ensemble avec t et on calcule une

coupe minimum séparant s et t (rappelons que s est aussi identifié avec un sommet de V).

Retournons maintenant au dernier cas. Soit $e = (i, j)$ une arête quelconque. Soit P_{ij} un plus court chemin entre i et j au sens de $1 - x$ en utilisant au plus $p - 1$ arêtes non contenant e . P_{ij} peut être calculé en temps polynomial par une version modifiée de l'algorithme de Bellman-Ford appliqué sur un graphe pondéré où le poids de l'arête e est infini et le poids de toute autre arête e' est $1 - x_{e'}$. $P_{ij} \cup (i, j)$ est un cycle C avec moins de p arêtes contenant e tel que $x(E(C)) - |V(C)|$ est maximum ($V(C)$ est l'ensemble des sommets de C et $E(C)$ est l'ensemble d'arêtes). Alors, on prend $A = V(C)$ et on vérifie quand l'inégalité $x(A) \leq |A| - 1$ est violée. Cette procédure est répétée pour toutes les arêtes $e = (i, j)$.

Étant donné un vecteur fractionnaire x , si la procédure qu'on vient de décrire ne permet pas de détecter des contraintes violées, une relaxation Lagrangienne est alors proposée.

On relaxe alors la contrainte $|A| \leq p$. La fonction objectif devient alors $|A| - x(A) + u(|A| - p)$ où u est le multiplicateur Lagrangien correspondant. Il est alors facile de calculer

$$f(u) = \min_{A \neq \emptyset} |A| - x(A) + u(|A| - p). \quad (2.12)$$

Ceci peut être fait en changeant les poids des arêtes reliées à t : les poids deviennent $2 + u$ au lieu de 2. Alors pour tout sommet $v \in V$, on identifie s avec v et on calcule la nouvelle coupe minimum séparant s et t . Le minimum parmi toutes les identifications possibles nous donne la valeur de $f(u)$. La théorie classique de la relaxation Lagrangienne implique que f est concave. f est aussi linéaire par morceaux avec moins de n morceaux. Soit A_u l'ensemble qu'on obtient quand on minimise $|A| - x(A) + u(|A| - p)$. $|A_u| - p$ est la tangente de la courbe. La Figure (2.7) montre la situation. Notons que si $|A| \leq |A_u|$ pour une valeur donnée de u , alors on a nécessairement $|A| - x(A) \geq |A_u| - x(A_u)$.

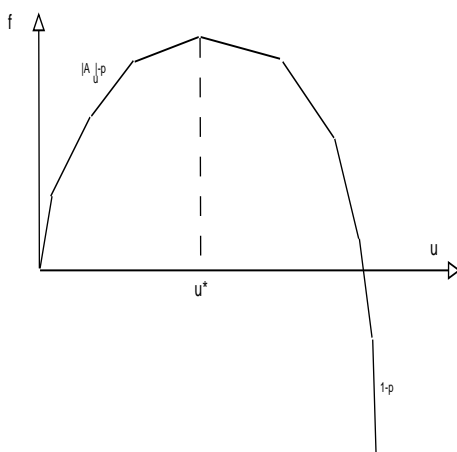


FIGURE 2.7 – Sur la fonction Lagrangienne

Si u est assez large (par exemple $u = n$), alors $|A_u| = 1$. On souhaite obtenir u^* tel que $|A_u| \geq p$ si $u < u^*$ et $|A_u| \leq p$ si $u > u^*$. Notons que u^* n'est pas nécessairement unique. L'heuristique qu'on utilise pour séparer les inégalités (2.2) est définie ainsi : on calcule approximativement u^* par une simple dichotomie dans l'intervalle $[0, n]$.

Si on obtient $f(u) \geq 1$ pour quelques valeurs de u , alors il n'y a aucune inégalité violée du type (2.2). En plus, si on obtient u , $|A_u| \leq p$ et $|A_u| - x(A_u) < 1$, alors on ajoute la contrainte $x(A_u) \leq |A_u| - 1$. Il existe une mauvaise situation dans laquelle on est incapable de décider si on a ou pas une violation de cette contrainte. Cette situation correspond au cas dans lequel la procédure de dichotomie a fini avec un certain u^* (avec une précision ε) tel que $|A_{u^*-\varepsilon}| > p$, $|A_{u^*+\varepsilon}| < p$, $f(u^* - \varepsilon) < 1$ et $|A_{u^*+\varepsilon}| - x(A_{u^*+\varepsilon}) \geq 1$. Cette situation apparaît de temps à autre.

2.5.3 Séparation des inégalités (2.4)

On montre que la séparation des inégalités (2.4) est NP-Difficile. On utilise une réduction à partir du problème de bi-clique maximum. Une bi-clique (S, T) est un graphe complet biparti où S et T sont deux ensembles disjoints de sommets qui définissent la partition. Étant donné un graphe biparti, on sait que calculer une bi-clique de taille maximum (i.e. $|S||T|$) est NP-Difficile [45, 113]. Il est aussi évident que le problème reste NP-Difficile si on ajoute une contrainte de cardinalité sur S ($|S| = k$ pour un nombre k donné).

Considérons une instance du problème de bi-clique maximum avec la contrainte de cardinalité ($|S| = p - 1$) définie par un graphe non orienté $G = (V, E)$. On définit un graphe pondéré $G' = (V', E', x)$ comme suit (x représente ici les poids) :

- Il existe deux sommets v_1 et v_2 qui correspondent à tout sommet $v \in V$;
- Il existe une arête reliant v_1 et v_2 pour tout $v \in V$; son poids est égal à 1 ;
- Si l'arête (v, w) n'existe pas dans G , alors les arêtes (v_1, w_2) et (v_2, w_1) existent dans G' et possèdent un poids égal à 1 ;
- Le graphe induit par $\{v_1, v \in V\}$ est complet et le poids de chaque arête du type (v_1, w_1) est un nombre large (par exemple $x_{v_1 w_1} = n^2$ où $n = |V|$) ;
- On ajoute un nouveau sommet z qui est connecté à tout sommet v_1 ; le poids de l'arête (v_1, z) est choisi tel que $x(\delta(v_1)) = n^2(n - 1) + n + 1$.

La Figure (2.8) nous montre bien cette situation.

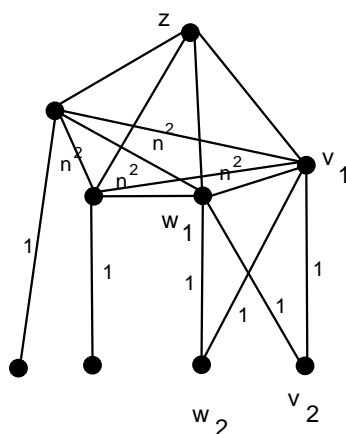


FIGURE 2.8 – Une réduction à partir du problème de bi-clique maximum

On montre que le calcul des ensembles A et B de V' maximisant $2x(A) + x(A, B) - 2|A| - |B|$ où $|A| \leq p - 1$ nous conduit à une solution du problème de bi-clique maximum avec la

contrainte $|S| = p - 1$.

Soit $C = V' \setminus \{A \cup B\}$. Alors $2x(A) + x(A, B) - 2|A| - |B|$ peut être réécrit ainsi : $2x(A) + x(A, B) - 2|A| - |B| = \sum_{v \in A} (x(\delta(v)) - 1) + |C| - |V'| - x(A, C)$. Autrement dit, maximiser $2x(A) + x(A, B) - 2|A| - |B|$ est équivalent à maximiser $\sum_{v \in A} (x(\delta(v)) - 1) + |C| - x(A, C)$. Comme les sommets du type v_1 possèdent un $x(\delta(v_1))$ large, A va contenir que les sommets du type v_1 et la cardinalité de A doit être exactement égale à $p - 1$. Comme on a $x(\delta(v_1)) = n^2(n - 1) + n + 1$ pour tout v_1 , la maximisation de la violation est en effet équivalente à trouver $p - 1$ sommets du type v_1 tel que $|C| - x(A, C)$ est maximum. En plus, étant donné un ensemble A , C contient les sommets v pour lesquels $x(A, v) = 0$ (tous les autres sommets peuvent être dans B). C'est à dire que C doit contenir que les sommets de type v_2 et il n'existe pas d'arêtes entre C et A . Il est aussi évident qu'on ne peut pas avoir $v_1 \in A$ et $v_2 \in C$. Maintenant en utilisant le fait qu'une arête (v_1, w_2) existe si et seulement si (v, w) n'existe pas dans G , on déduit alors qu'on obtient une bi-clique en mettant $S = \{v \in V, v_1 \in A\}$ et $T = \{v \in V, v_2 \in C\}$. Cette bi-clique a la taille maximum et satisfait la contrainte de cardinalité.

Comme la transformation précédente est polynomiale, la séparation des inégalités (2.4) est NP-Difficile. Observons ici que pour simplifier, on a considéré ci-dessus des problèmes d'optimisation au lieu des problèmes de décision.

Il est facile de voir que la séparation peut se faire en temps polynomial si $p \leq q$ où q est un nombre constant positif (par simple énumération).

Notons qu'on sait à partir de la Proposition (2.12) qu'on peut se focaliser sur $|A| = p - 1$. Dans ce cas, la séparation est aussi facile si $p \geq n - q$.

Proposition 2.15: Les inégalités (2.4) sont en générale NP-Difficile à séparer. Si $p \leq q$ où q est une constante positive quelconque, alors elles peuvent être séparées en temps polynomial. Les inégalités qui induisent des facettes (i.e, $|A| = p - 1$) peuvent aussi se séparer facilement quand $p \geq n - q$.

A partir du fait que $2x(A) + x(A, B) - 2|A| - |B| = \sum_{v \in A} (x(\delta(v) \setminus \delta(C)) - 1) + |C| - |V|$, il en découle que la séparation est facile quand $C = V \setminus \{A, B\}$ est fixé. Ceci va être utilisé pour séparer les inégalités (2.4) pour des valeurs intermédiaires de p .

La séparation se fait par une heuristique en utilisant la relaxation Lagrangienne de la section précédente. Si on obtient pour un certain u (le multiplicateur Lagrangien) $|A_u| > p - 1$, alors on prend $C = V \setminus A_u$ et on résout le problème de séparation d'une façon exacte.

Si $|A_u| \leq p - 1$ pour un certain u , on commence avec $A = A_u$ et on essaye d'augmenter la taille de A d'une manière gloutonne en ajoutant les sommets v pour lesquels $x(A, v) - 1$ est maximum. Si la taille de A atteint $p - 1$, on augmente alors la taille de B en utilisant encore une fois le même algorithme glouton.

2.5.4 Séparation des inégalités (2.6)

Étant donné un ensemble de nombres $(\alpha_e)_{e \in E}$ (sans restriction de signe) et un nombre $0 \leq i \leq \lfloor \frac{n}{p+1} \rfloor - 1$, γ_{n+i} peut être calculé par un simple algorithme glouton : calculer un arbre couvrant de poids minimum et ajouter $i + 1$ arêtes de poids minimum. β_i peut aussi être calculé de la même façon.

Rappelons que β_i est défini selon un arbre couvrant de coût minimum T_α . Mais il est facile de voir que l'inégalité (2.6) est aussi valide si on considère tout arbre couvrant T au lieu de T_α (la preuve de la Proposition (2.3) est basée uniquement sur la connectivité de T_α). Toutefois, β_i

est clairement minimum si on considère un arbre couvrant de coût minimum. En d'autres mots, on obtient toujours une inégalité valide meilleure si on considère un arbre couvrant minimum. De ce fait, pour simplifier la séparation, on considérera d'abord un arbre fixé T et on résout le programme linéaire suivant :

$$\left\{ \begin{array}{l} \min \sum_{e \in E} \alpha_e x_e - Z \\ Z \leq \gamma_{n+i} - i\beta; \forall 0 \leq i \leq \lfloor \frac{n}{p+1} \rfloor - 1 \\ \beta \geq \alpha_e; \forall e \in T \\ \gamma_{n+i} \leq \sum_{e \in H} \alpha_e; \forall 0 \leq i \leq \lfloor \frac{n}{p+1} \rfloor - 1, \forall H \text{ sousgraphe connexe avec } n+i \text{ arêtes} \\ -1 \leq \alpha_e \leq 1; \forall e \in E \\ \gamma_{n+i} \in \mathbb{R}; \forall 0 \leq i \leq \lfloor \frac{n}{p+1} \rfloor - 1 \\ Z \in \mathbb{R}, \beta \in \mathbb{R}. \end{array} \right. \quad (2.13)$$

Dans la formulation précédente, la variable β représente une borne supérieure de tout α_e . Au lieu d'utiliser β_i , on considère le produit $i\beta$ qui donne une borne supérieure du coût de tout ensemble de i arêtes de T . Une variable Z est aussi ajoutée pour noter $\min_{0 \leq i \leq \lfloor \frac{n}{p+1} \rfloor - 1} (\gamma_{n+i} - i\beta)$.

Le troisième ensemble de contraintes définit γ_{n+i} : pour tout graphe connexe H avec $n+i$ contraintes, on écrit le coût de H est plus grand que γ_{n+i} . Comme le nombre de graphes H peut être très large, on les ajoute d'une manière itérative par une génération de contraintes. Étant donné un vecteur courant α , on calcule par l'algorithme glouton un graphe connexe de coût minimum avec $n+i$ arêtes et on ajoute les contraintes correspondantes si le coût est inférieur à la valeur courante de γ_{n+i} .

Après la résolution du programme linéaire, on obtient un vecteur α qui maximise la violation. La contrainte peut être améliorée en remplaçant l'arbre T par T_α et on calcule les nombres β_i . Alors la contrainte ajoutée à notre relaxation linéaire doit être $\sum_{e \in E} \alpha_e x_e \geq \min_{0 \leq i \leq \lfloor \frac{n}{p+1} \rfloor - 1} (\gamma_{n+i} - \beta_i)$ s'il y a bien une violation.

Notons aussi que si x est le vecteur d'incidence d'un graphe avec plus de $\lfloor \frac{n}{p+1} \rfloor$ composantes connexes alors les inégalités données par le programme linéaire doivent être violées par x . En d'autres mots, ces inégalités constituent une formulation exacte du problème de design des réseaux avec une borne supérieure sur le nombre de composantes connexes.

2.5.5 Séparation des inégalités (2.7), (2.8) et (2.9)

Les inégalités (2.7) sont séparées avec une heuristique en utilisant un simple algorithme glouton. D'abord, on calcule le flot maximum f_{ij} entre chaque paire de sommets i et j (la capacité du lien e est x_e). Alors on construit un ensemble S par un glouton en ajoutant à chaque itération un sommet i qui minimise $\sum_{j \in S} f_{ij}$. Si on obtient une violation à une quelconque itération, on ajoute alors la contrainte.

Les inégalités (2.8) peuvent se séparer d'une manière exacte et en un temps polynomial. Après avoir calculé les flots maximums (et coupes minimums), on a juste à vérifier quand $\sum_{i,j \in V, i < j} f_{ij}$ serait inférieur au second membre de (2.8).

La séparation des inégalités (2.9) est aussi basée sur les techniques de flot. Pour toute arête $e = (i, j) \in E$, on calcule un flot maximum f'_{ij} entre i et j dans le graphe G où les capacités de chaque lien e' est le minimum de $x_{e'}$ et x_e . Ceci nous donne une coupe $\delta(A_{ij})$ tel que

$\sum_{e' \in \delta(A_{ij})} x_{ij,e'} = f'_{ij}$ où $x_{ij,e'} = x_{ij}$ si $x_e \leq x_{e'}$ et $x_{ij,e'} = x_{e'}$ sinon. En faisant cela pour toute arête $e = (i, j)$, on minimise le membre de gauche de l'inégalité (2.9). Maintenant, on peut d'une manière heuristique maximiser le membre de droite de l'inégalité (2.9) avec une méthode gloutonne. D'abord, on calcule deux sommets i et j qui minimisent f_{ij} . Si $f_{ij} \geq 1$, alors le meilleur membre de droite est simplement $n + p + 1$. Dans l'autre cas, on met i et j dans S et on essaye d'augmenter la taille de S en y ajoutant des sommets k pour lesquels $\sum_{i \in S} f_{ik} < 1$. On ajoute toujours à S le sommet $k \notin S$ qui minimise $\sum_{i \in S} f_{ik}$.

Ces inégalités ont été implémentées. Cependant, elles semblent être dans la plupart des cas dominées par les inégalités décrites auparavant dans ce chapitre. Ceci nous a amené à ne pas les inclure dans les expérimentations numériques présentées dans la section suivante.

2.6 Résultats Expérimentaux

Un algorithme à plans coupants est implémenté pour résoudre ce problème. On a utilisé la formulation (2.1) avec les inégalités (2.2), (2.4) et (2.5). Ces dernières sont considérées dans l'ordre suivant : (2.2), ensuite (2.4) et finalement (2.5). On rappelle que les inégalités (2.2) et (2.4) sont simultanément séparées. Les inégalités $x(\delta(v)) \leq n - p + 1$ sont directement incluses dans le programme linéaire.

Après l'algorithme à plans coupants, un algorithme de Branch&Cut est ensuite entamé en utilisant les fonctions de callback de [74]. Toutes ces expérimentations ont été réalisées sur un Pentium 4, de 3.25 Go de RAM, et de 2.39 GHz.

L'ensemble des 3 familles d'inégalités citées ci-dessus est généré dans la phase de l'algorithme des plans coupants (au niveau de la racine de l'arborescence), alors qu'on ne sépare que les inégalités (2.2) au niveau des autres noeuds de l'arbre du Branch&Cut. Ce choix a été pris pour deux raisons. La première est que la séparation des inégalités (2.5) est relativement lente. La seconde raison, est qu'on sait que toute solution entière satisfaisant les inégalités (2.2) est une solution réalisable, donc l'algorithme de Branch&Cut se termine nécessairement par une solution optimale (lorsque on ne limite pas le temps et l'espace mémoire). Notons que toute solution entière non réalisable peut être séparée en un temps polynomial grâce à l'algorithme (modifié) de Bellman-Ford (pour vérifier s'il existe des cycles de taille inférieure à p).

Une stratégie de recherche en profondeur d'abord est utilisée dans l'algorithme de Branch&Cut. Une borne supérieure est calculée par la recherche d'un cycle Hamiltonien de coût minimum.

Chaque fois qu'un calcul de flot maximum est nécessaire pour séparer des inégalités, nous utilisons l'algorithme de [50].

Nous avons considéré trois types d'instances générées aléatoirement et selon différents modes, et deux instances réelles fournies par Orange Labs contenant 32 et 23 sommets. Le premier ensemble relatif à la Table 2.1 est basé sur des graphes complets, aléatoires et Euclidiens. Les coordonnées de chaque sommet sont choisies d'une manière aléatoire et uniforme entre 0 et 1. Le poids d'une arête est proportionnel à la distance entre les extrémités de cette arête.

Le deuxième ensemble d'instances (Table 2.2) est relatif aux graphes complets aléatoires dont les poids des arêtes sont des nombres positifs choisis d'une manière aléatoire et uniforme

TABLE 2.1 – Graphes Complets aléatoires et Euclidiens

n	p	temps sep	temps relax	temps tot	gap(%)	gap2(%) (2.2)	(2.4)	(2.5)	Nb coupes	Nb noeuds
20	6	0.33	0	0.33	0	10.4	10	9.6	0	1
	10	1.3	0	1.3	0	11.7	15.6	12.6	0	1
	16	5.6	0.3	6	0	15.1	28.3	14.6	11.3	1
35	7	2.3	0.3	2.6	0	8.8	21.3	27.6	0	4.6
	16	12.3	0.3	15.3	0	10.1	37.3	20.3	0	36
	30	186.6	5	238.6	0	17.7	56.6	15	62.6	1
50	12	11	0	11.3	0	8.8	32.6	23.6	0	6.3
	25	36	0	104	0	9.9	47.3	16.3	0	330.3
	45	2540.3	272.7	6812.7	3.7	16.2	76.7	16.3	236.3	913.7
65	15	62	0.6	86.3	0	10.5	55.6	36.3	0	148
	30	233	2.3	394.6	0	10.1	79.3	27.6	0	340.3
	60	6213.3	303.3	7200	4.3	18.7	95.7	21	213.7	78
100	25	326.3	0.6	349	0	6.7	64.3	36.3	0	39.6
	50	1288	2.7	7200	6.8	14.2	81.3	22	0	998.7
	90	7160.3	39.7	7200	8.1	18.7	137	15.7	31	1

entre 0 et 1000. Le poids moyen est alors de 500, et l'écart type est de 288.7.

Le troisième ensemble d'instances (Table 2.3) est similaire au précédent. La seule différence, est qu'on a réduit l'écart type qui est à nouveau égal à 223.8. En d'autres mots, les poids de ce troisième ensemble d'instances sont assez proches de leur moyenne.

Différentes valeurs de n et de p ont été prises. Trois instances sont alors générées pour toute valeur de n et de p . Les résultats présentés dans des tables 2.1, 2.2 et 2.3 sont obtenus en calculant les moyennes des trois instances générées.

Dans ces tables, on note par "temps sep" le temps en secondes de séparation de toutes les coupes dans la phase de l'algorithme à plans coupants (avant le branch&cut), et par "temps relax", on note le temps mis pour la résolution de la relaxation (avant le branch&cut). "temps tot" note le temps total mis pour résoudre le problème, l'algorithme de Branch&Cut compris. Les nombres de contraintes générées par l'algorithme à plans coupants (avant le branch&cut) sont aussi donnés : ils sont notés par (2.2), (2.4) et (2.5). Le nombre de contraintes générées dans l'arbre du Branch&Cut est noté par "Nb coupes". On rappelle que dans le Branch&Cut, on a considéré que les inégalités (2.2). On donne finalement le nombre de noeuds de l'arbre construit, on le note par "Nb noeuds".

Le gap final, qui est la différence entre la borne supérieure et la meilleure solution trouvée, le tout divisé par la borne supérieure, est donné et noté par "gap". Si ce gap est égal à 0, alors le problème est résolu jusqu'à l'optimum. On donne aussi la valeur de "gap 2", qui représente la différence entre la meilleure borne supérieure et le coût obtenu avec $p = 2$, le tout étant divisé par la borne supérieure. Comme le coût obtenu lorsque $p = 2$ est une borne inférieure, on a

$\text{gap} \leq \text{gap } 2$.

Pendant les simulations, on a limité le temps total à 2 heures.

On remarque aussi que les contraintes (2.5) sont plus efficaces pour des valeurs de p assez larges. Seulement, leur séparation est relativement lente, ce qui fait que la phase de l'algorithme à plans coupants peut devenir très lente pour des valeurs de n et p assez grandes. Par exemple, lorsque $n = 100$ et $p = 90$, le temps limite est atteint avant d'entamer la phase de Branch&Cut, alors la borne supérieure est simplement donnée par le coût minimum d'un cycle Hamiltonien, et que la borne inférieure est donnée par l'optimum de la dernière relaxation linéaire.

On peut aussi remarquer que le problème devient plus facile à résoudre lorsque l'écart type des poids des arêtes est petite. "gap 2" a aussi tendance à diminuer avec la diminution de l'écart type.

TABLE 2.2 – Graphes complets aléatoire avec un écart type élevé

n	p	temps sep	temps relax	temps tot	gap(%)	gap2(%)	(2.2)	(2.4)	(2.5)	Nb coupes	Nb noeuds
20	6	0	0	0	0	2.3	1.7	2.7	0	0	1
	10	0	1.3	3	0	8.5	7.3	11.7	3.3	42.7	134
	16	9	0.7	10.7	0	26.2	21	24.7	24	1.3	1
35	7	0.7	0	1	0	0.6	1.3	1.3	0	0	1
	16	11.7	0	106.7	0	5.9	8.3	16	0	333.3	929.7
	30	399.7	62.7	2914.7	6.3	29.9	41.3	21	147.3	1854	24414.3
50	12	0	10.3	23.7	0	0.61	5.3	12.7	0	6.3	3.3
	25	145	2.3	7200	25.1	27.5	14.3	17	12.7	4934.7	20551
	45	3876.3	960.7	5399.3	12.1	38.4	57.3	33.3	373	127	1588.7
65	15	60.7	0.3	121.7	0	1.7	8.3	16.7	0	39.7	127.7
	30	529.3	6	7200	28.2	29.9	30	24	16.7	1819.3	6785.3
	60	6742	458	7200	12.3	36.2	87	37.7	221	0	1
100	25	479.3	2	5142	4.9	5.6	8.6	17.7	0	422.3	1123.7
	50	5628	38	7200	39.3	40.4	21.7	42.7	37	203.3	190
	90	7154.7	42.3	7200	22.5	41.3	77.3	48.3	20.7	0	1

Finalement, les figures (2.9) et (2.10) représentent le coût minimum (plus précisément "gap 2") en fonction de p . On présente alors ci-dessous deux instances réelles avec respectivement 32 et 23 sommets. Le problème est résolu jusqu'à l'optimum pour toute valeur de p entre 2 et 31 (respectivement 2 et 22). On observe alors sur ces figures un important saut du coût quand les valeurs de p sont proches de $|V| - 1$. Ceci est aussi observé pour d'autres instances.

2.7 Conclusion

Dans ce chapitre, on a présenté un algorithme à plans coupants suivi d'un Branch&Cut efficace pour résoudre le problème de synthèse d'un réseau avec composantes connexes unicycliques respectant une contrainte sur la taille des cycles. Plusieurs extensions de ces travaux

TABLE 2.3 – Graphes complets aléatoires avec faible écart type

n	p	temps sep	temps relax	temps tot	gap(%)	gap2(%) (2.2)	(2.4)	(2.5)	Nb coupes	Nb noeuds
20	6	0	0	0	0	0.3	2	3	0	1
	10	0.7	0	1.3	0	1.3	5.7	11	1	27.7
	16	16.3	0.7	74.3	0	3.9	20.3	22	43.3	411
35	7	1	0	1	0	0.1	2.7	0	0	1
	16	7	0.3	114	0	0.3	6.7	8.7	0	342.7
	30	226.3	9	2678	0	6.4	47.3	24	76	2756
50	12	7	0	28	0	0.1	5.3	11	0	1.7
	25	116.3	0.7	7200	5	5.5	14	14.3	8.7	4502.3
	45	2298.3	225.7	5536.7	2.3	7.7	55.3	30.7	218	19953.3
65	15	54.7	0	178	0	0.1	8	15.7	0	52.3
	30	450.3	2	7200	3.2	3.4	26.3	25.3	14	2083.7
	60	6883	317	7200	2.3	6.7	81.7	28.7	222	6330
100	25	501.3	1.7	6550	0	0.1	11.7	29.7	0	625
	50	3956	22.7	7200	4.7	4.8	25.7	36.3	40.3	594.7
	90	7123.3	76.7	7200	2.4	4.8	72.3	56	19	2164.3

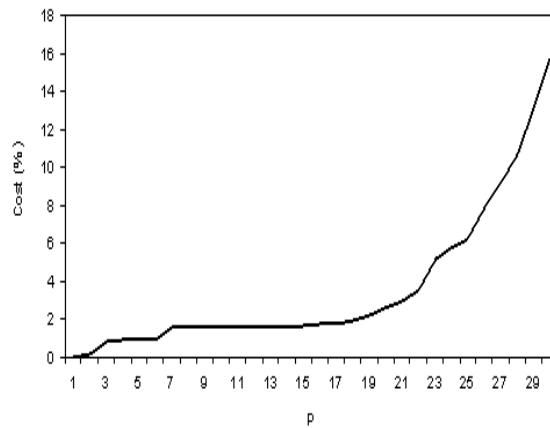


FIGURE 2.9 – Évolution du coût du réseau en fonction de p pour l'instance à 32 noeuds.

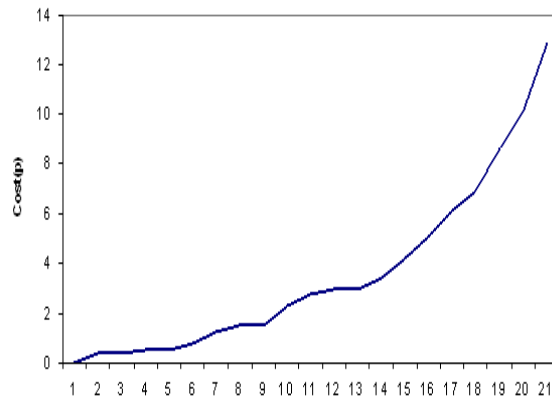


FIGURE 2.10 – Évolution du coût du réseau en fonction de p pour l’instance de 23 noeuds.

sont à envisager. À commencer par voir d’autres inégalités valides pour pouvoir résoudre le problème pour des instances plus larges. On a aussi mentionné dans ce chapitre que le problème traité est NP-Difficile lorsque p est une donnée. La complexité reste un sujet ouvert lorsque p est une constante.

Une autre perspective est liée à l’approximation du problème pour des classes particulières de graphes et aussi pour des petites valeurs de p . On peut également voir que la technique d’approximation primale-duale développée par [49] pour les problèmes de synthèse des forêts peut être facilement utilisée lorsque les poids des arêtes satisfont les inégalités triangulaires. La technique de [49] permet notamment de calculer en temps polynomial une forêt où le nombre de sommets par composante connexe est au moins $p + 1$ et dont le coût est inférieur au coût double d’une forêt optimale satisfaisant les mêmes contraintes.

En considérant cette forêt et en dédoublant les arêtes, on obtient un graphe eulérien par composante connexe. Ce graphe Eulérien peut être transformé en cycle de taille au moins $p + 1$ (à la manière de l’algorithme de Christofides). Le rapport d’approximation final est donc 4. Une question se pose : peut-t-on faire mieux ?

Finalement, il est probable que le travail présenté dans ce chapitre soit le premier où le matroïde bi-circulaire est utilisé dans le contexte de l’optimisation combinatoire. On pense que quelques problèmes d’optimisation combinatoire faisant intervenir des cycles peuvent être modélisés en utilisant la formulation compacte relative au matroïde bi-circulaire avec intégration d’autres contraintes.

Réseaux à composantes unicycliques avec contraintes de type Steiner

Dans ce chapitre, nous étudions une autre variante du problème de synthèse d'un réseau à composantes connexes unicycliques. Nous intégrons une contrainte de type Steiner : certains sommets doivent appartenir à un cycle. Ce problème est une généralisation du problème de calcul d'un 2-couplage de coût minimum. Nous montrons dans ce chapitre que le problème que nous venons d'introduire est également facile à résoudre. Une formulation linéaire exacte et étendue sera donnée. Nous présentons également une description partielle de l'enveloppe convexe des vecteurs d'incidence de réseaux vérifiant les contraintes.

3.1 Introduction

Étant donné un graphe pondéré et non orienté noté par $G = (V, E)$, et un sous-ensemble de sommets $S \subset V$, on veut calculer un sous-graphe $G' = (V, E')$ ($E' \subset E$) couvrant et de poids minimum, avec des composantes unicycliques, tel que chaque sommet de S appartient à un cycle. Un exemple de graphe de Steiner avec des composantes unicycliques est donné dans la figure 3.2 (la partie gauche), où $|V| = 150$ et S contient 70 sommets qui sont représentés par des carrés noirs.

Nous avons déjà montré l'intérêt des graphes à composantes connexes unicycliques pour construire des réseaux ayant un certain niveau de robustesse.

Dans ce chapitre, on s'intéresse aux contraintes de type Steiner. Les sommets de S sont supposés être "importants" et doivent être protégés contre les pannes des liens. C'est pour cette raison qu'ils doivent appartenir à des cycles.

Si ce type de contraintes de Steiner ne sont pas imposées ($S = \emptyset$), nous savons déjà que le problème est facile. Si $S = V$, on obtient le problème de 2-couplage parfait qui est un problème facile (voir, e.g., [117]).

Il existe aussi des liens avec le problème de l'étoile anneau. Dans ce cas, on cherche un seul cycle qui traverse certains sommets et que les autres seront connectés à ceux du cycle. Le problème de l'étoile anneau et ses applications sont étudiés dans [2, 82]. Quelques contraintes de type Steiner sont aussi considérées dans [123] où quelques sommets doivent appartenir au cycle, et s'il y a un sommet qui n'est pas sur le cycle, alors il existe une pénalité à payer. Le but est alors de trouver un cycle qui passe par un sous-ensemble de sommets incluant S et minimisant le coût du cycle et les pénalités des sommets non visités [123]. Ce problème est évidemment NP-Difficile.

Dans notre problème, le nombre de composantes connexes n'est pas fixé d'avance. Certains sommets dont ceux de S appartiennent aux cycles, alors que les autres doivent être connectés aux cycles par des arbres.

Ce chapitre sera organisé ainsi : on commence par donner une première formulation mathématique dans la section qui suit. Ensuite, on présente dans la section 3.3 un algorithme polynomial et une formulation linéaire étendue du problème étudié. Dans la Section 3.4, on présente une description partielle de l'enveloppe convexe des vecteurs d'incidence des graphes à composantes connexes unicycliques respectant une contrainte du type Steiner. Les algorithmes de séparation de quelques inégalités valides sont décrits dans la Section 3.5. Un de ces algorithmes de séparation est une modification non triviale de l'algorithme de Padberg&Rao [109] basé sur la solution du problème de la $s - t$ T -coupe de poids minimum calculée dans un graphe orienté. La Section 3.6 est dédiée à quelques extensions du problème de Steiner où la difficulté reste la même (problèmes polynomiaux). On termine ce chapitre par une conclusion dans la Section 3.8 avec quelques directions de recherche.

3.2 Une première formulation mathématique

Bien que les notations utilisées sont similaires à celles du chapitre précédent, nous les rappelons brièvement pour faciliter la lecture du chapitre. Nous considérons un graphe $G = (V(G), E(G))$ simple et non orienté. Lorsqu'il n'y a pas d'ambiguïté, on écrit V au lieu de $V(G)$, et E à la place de $E(G)$. Un coût w_e est associé à toute arête $e \in E$.

Une arête $e \in E$ est parfois notée par (i, j) où i et j sont les extrémités de e . Alors pour tout vecteur donné $(x_e)_{e \in E}$, x_e peut être aussi écrit comme x_{ij} ou $x_{(i,j)}$. Soient A et B deux sous-ensembles disjoints de sommets, et soit $E(A, B)$ l'ensemble des arêtes qui possèdent une extrémité dans A et l'autre extrémité dans B . Alors $x(A, B)$ est la somme des valeurs de x des arêtes qui ont une extrémité dans A et l'autre dans B : $x(A, B) = \sum_{e \in E(A, B)} x_e$.

On note par $\delta(A)$ l'ensemble d'arêtes qui ont exactement une extrémité dans A . Alors $\delta(A) = E(A, V \setminus A)$, et parfois on utilise \bar{A} pour exprimer $V \setminus A$. Si $A = \{v\}$ où $v \in V$, alors on écrit $\delta(v)$ au lieu d'écrire $\delta(\{v\})$. On utilise aussi $x(A)$ pour exprimer la somme des valeurs de x des arêtes qui ont les deux extrémités dans A : $x(A) = \sum_{(i,j) \in E, i \in A, j \in A} x_{ij}$.

Si C est un ensemble d'arêtes $C \subset E$, alors $x(C)$ représente la somme des valeurs de x des éléments de C : $x(C) = \sum_{e \in C} x_e$. On aura besoin aussi d'une notation relative aux graphes orientés.

On note par $[u, v]$ un arc orienté de u vers v (i.e. u est la queue et v la tête). $E^+(A, B)$ est l'ensemble d'arcs avec les extrémités initiales (queues) dans A et les extrémités terminales (têtes)

dans B . $E^+(A, \bar{A})$ peut être remplacé par $\delta^+(A)$.

Après cette introduction, on donne maintenant une formulation mathématique du problème posé ci-dessus.

Rappelons que nous avons montré au précédent chapitre que le problème de synthèse de réseaux à composantes unicycliques (sans les contraintes de Steiner) se formule comme suit :

$$\begin{aligned}
& \min \sum_{e \in E} w_e x_e \\
& x_e = y_{ie} + y_{je}, \forall e = (i, j) \in E \\
& \sum_{e=(i,j) \in E} y_{ie} = 1, \forall i \in V \\
& 0 \leq y_{ie}, \forall i \in V, e = (i, j) \in E \\
& x_e \leq 1, \forall e \in E.
\end{aligned} \tag{3.1}$$

Nous avons également déjà prouvé que les contraintes d'unicyclicité peuvent aussi être exprimées par le système suivant :

$$\begin{aligned}
& x(A) \leq |A|, \forall A \subset V \\
& x(V) = |V| \\
& 0 \leq x_e \leq 1, \forall e \in E.
\end{aligned} \tag{3.2}$$

Les contraintes (3.2) peuvent se séparer en temps polynomial (voir chapitre 2).

Les contraintes liées à S peuvent être exprimées par les inégalités suivantes :

$$x(A) + \frac{1}{2}x(A, S \setminus A) \leq |A|, \forall A \subset V. \tag{3.3}$$

Proposition 3.1: Les inégalités (3.3) sont valides.

Preuve: Les inégalités (3.3) sont équivalentes à $x(A, S \setminus A) \leq 2(|A| - x(A))$. Si x est le vecteur d'incidence d'un réseau de Steiner avec des composantes unicycliques, alors $x(A) - |A|$ représente le nombre de composantes connexes du graphe induit par A qui ne contient aucun cycle. Supposons qu'il existe une composante connexe de A et qui ne contient pas de cycle, et qui est connectée à 3 sommets de $S \setminus A$. Alors, un des 3 sommets n'est pas sur le cycle. Autrement, chaque composantes de A ne contenant pas de cycle est liée à au plus deux sommets de $S \setminus A$. De plus, chaque composante de A contenant un cycle n'est pas liée à $S \setminus A$, ceci implique alors que $x(A, S \setminus A) \leq 2(|A| - x(A))$. \square

On peut voir clairement que les inégalités (3.3) dominent les inégalités $x(A) \leq |A|$. Autrement dit, les inégalités (3.3) induisent directement l'unicyclicité (quand $x(V) = |V|$). En outre, il est facile de voir que si x est entier, alors les inégalités (3.3) sont suffisantes pour garantir que tous les sommets de S appartiennent aux cycles. Par conséquent, notre problème de synthèse de réseaux se formule comme suit :

$$\begin{aligned}
& \min \sum_{e \in E} w_e x_e \\
& x(A) + \frac{1}{2}x(A, S \setminus A) \leq |A|, \forall A \subset V \\
& x(V) = |V| \\
& x_e \in \{0, 1\}, \forall e \in E.
\end{aligned} \tag{3.4}$$

On verra un peu plus loin dans ce chapitre que les inégalités (3.3) peuvent se séparer en un temps polynomial.

3.3 Complexité et formulation étendue

Comme mentionné précédemment, lorsque $S = V$ notre problème d'optimisation devient le problème bien connu du 2-couplage parfait [37, 80, 117]. De même, si on prend $S = \emptyset$, on obtient le problème de synthèse de réseaux à composantes unicycliques qui est également facile à résoudre.

Dans ce qui suit, on va montrer que notre problème reste facile à résoudre pour tout ensemble S .

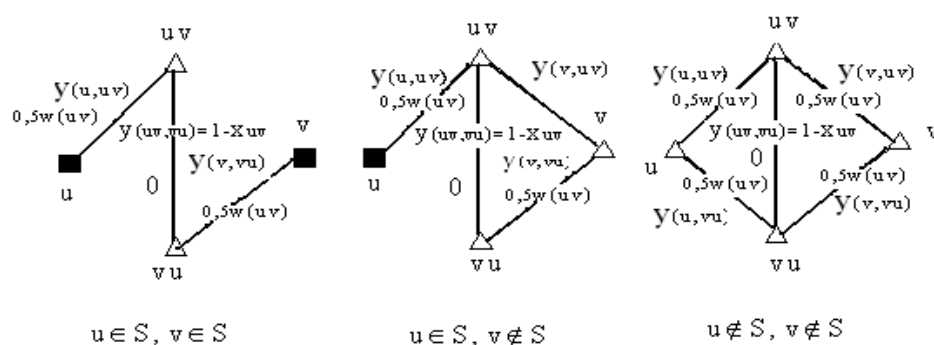


FIGURE 3.1 – Construction du nouveau graphe étendu

Proposition 3.2: Le problème de synthèse de réseaux à composantes unicycliques avec contrainte de type Steiner peut être résolu en temps polynomial.

Preuve: On va montrer que le problème d'optimisation considéré dans ce chapitre se réduit à un problème de b -couplage dans un graphe plus large et approprié qu'on construit comme suit.

Notons ce nouveau graphe H . Pour tout sommet $u \in V(G)$, il existe un autre sommet $u \in V(H)$. Pour toute arête $(u, v) \in E(G)$, on considère deux sommets uv et vu comme le montre la Figure (3.1). Une arête (uv, vu) est incluse dans $E(H)$. Si $u \notin S$ et $v \notin S$, 4 autres arêtes seront alors considérées : (u, uv) , (u, vu) , (v, uv) et (v, vu) . Si $u \in S$ et $v \notin S$, alors (u, vu) sera éliminée. Finalement, si $u \in S$ et $v \in S$, $E(H)$ contiendra (u, uv) et (v, vu) . Le poids de chaque arête du type (u, uv) et (u, vu) lorsqu'elle existe est de $\frac{w_{uv}}{2}$, alors que le poids de (uv, vu) est de 0.

On considère maintenant un b -couplage parfait de poids minimum où chaque sommet $u \in V(H)$ qui correspond à un sommet $u \in V(G)$ a un degré de 2, alors que chaque sommet de type uv possède un degré de 1. Il est clair que l'arête (uv, vu) appartient au b -couplage seulement si les 4 autres arêtes (u, uv) , (u, vu) , (v, uv) et (v, vu) ne font pas partie de la solution. De plus, il y a toujours un nombre pair (soit 0 ou 2) d'arêtes parmi (u, uv) , (u, vu) , (v, uv) et (v, vu) qui appartiennent à un b -couplage quelconque. Si (u, uv) est dans le b -couplage, alors (u, vu) est aussi dans ce b -couplage, et que ni (v, uv) ni (v, vu) ne sont dans le b -couplage, ou alors (v, vu) est dans le b -couplage et que ni (v, uv) ni (u, vu) ne sont dans le b -couplage. Notons que si

(uv, vu) n'est pas dans le b-couplage, alors il y a exactement 2 arêtes parmi (u, uv) , (u, vu) , (v, uv) et (v, vu) qui sont dans le b-couplage.

On construit une solution du problème de Steiner en prenant une arête (u, v) si et seulement si l'arête (uv, vu) n'appartient pas au b-couplage. On montre alors qu'on obtient vraiment un réseau de Steiner avec des composantes unicycliques de coût total minimum.

Pour cela, on pose P le nombre d'arêtes du type (uv, vu) qui sont dans le b-couplage et Q le nombre de celles qui ne le sont pas. Alors on a $P + Q = |E(G)|$ et $P + 2Q = |E(G)| + |V(G)|$, où la seconde égalité est impliquée par les conditions des degrés. Par conséquent, $Q = |V(G)|$. Sauf que Q est aussi le nombre d'arêtes dans la solution construite à partir du b-couplage, alors la solution qu'on obtient de cette façon est un graphe simple dans lequel le nombre d'arêtes est égal au nombre de sommets. Soit $S' \subset V(G) \subset V(H)$ le sous-ensemble de sommets u tels qu'il existe $v \in V(G)$ pour lequel l'une des deux arêtes (u, uv) et (u, vu) appartient au b-couplage. Notons que comme le degré de u est de 2 dans le b-couplage, alors l'existence d'un tel sommet v implique l'existence d'un autre sommet w qui satisfait la même condition (il y a exactement une arête parmi (u, uw) et (u, wu) qui appartient au b-couplage). Comme l'arête (u, vu) n'existe pas lorsque $u \in S$, alors il est clair que S' contient S . De plus, si $u \in S'$ et v est un sommet tel que exactement une des deux arêtes (u, uv) et (u, vu) appartient au b-couplage, alors v est aussi dans S' . La solution construite à partir du b-couplage contient l'arête (u, v) (car (uv, vu) n'est pas dans le b-couplage). En d'autres mots, la solution contient un b-couplage de S' : chaque sommet de S' a un degré 2. Alors on a montré que les sommets de S appartiennent aux cycles.

Maintenant, on s'intéresse aux sommets de $V(G) \setminus S'$. Pour tout sommet $u \in V(G) \setminus S'$, il y a exactement un sommet tel que à la fois (u, uv) et (u, vu) appartiennent au b-couplage. Ceci implique que (u, v) fait partie de la solution construite à partir du b-couplage. Si v est dans S' , alors on déduit que u appartient à une composante connexe qui contient au moins un cycle. Si v n'est pas dans S' , alors il y a un sommet $w \neq u$ tel qu'on a à la fois (v, vw) et (v, wv) dans le b-couplage. En continuant ce processus, il est alors clair qu'on va finir avec soit un sommet dans S' , ou avec un sommet qu'on a toujours considéré. Dans le premier cas, on sait toujours que ceci implique que u appartient à une composante qui contient au moins un cycle. Dans le second cas, en revenant sur un sommet qu'on a toujours pris en considération, alors on créera un cycle. Ceci prouve encore que u appartient à une composante qui contient au moins un cycle.

Pour résumer, on a montré que la solution construite à partir du b-couplage est telle que chaque sommet $u \in V(G) \subset V(H)$ est dans une composantes connexe qui contient au moins un cycle. De plus, on a déjà remarqué que le nombre d'arêtes est égal au nombre de sommets. Ceci implique que notre solution est un graphe dans lequel chaque composante connexe contient exactement un cycle. On a aussi noté que tous les sommets de S sont sur les cycles. En d'autres termes, la solution obtenue est un réseau à composantes connexes unicycliques satisfaisant la contrainte de type Steiner. Les coûts des arêtes dans le nouveau graphe construit ont été choisis de telle manière que le coût du réseau soit exactement égal au coût du b-couplage.

Finalement, un b-couplage de poids minimum peut être calculé en un temps polynomial soit par la génération de contraintes [109, 38] à travers l'algorithme des ellipsoïdes ou par des algorithmes combinatoires [93]. \square

Cette preuve implique une formulation linéaire étendue de notre problème de synthèse de réseaux. Soit $(y_e)_{e \in E(H)}$ un vecteur en 0-1 des arêtes du graphe H introduit dans la dernière preuve (voir aussi la Figure (3.1)). On rappelle que l'arête (uv, vu) appartient au b-couplage si et seulement si (u, v) fait partie de la solution du problème de Steiner construit à partir du b-couplage. Ainsi, on peut écrire que $y_{(uv, vu)} = 1 - x_{uv}$.

Une formulation en nombre entiers est alors donnée par :

$$\begin{aligned}
\min \quad & \sum_{e \in E(G)} w_e x_e \\
& y_{(u,uv)} + y_{(v,uv)} = x_{uv}, \forall (u,v) \in E(G), u \in V(G), v \in V(G) \setminus S \\
& y_{(u,uv)} = x_{uv}, \forall (u,v) \in E(G), u \in V(G), v \in S \\
\sum_{(u,v) \in E(G)} y_{(u,uv)} + y_{(u,vu)} &= 2, \forall u \in V(G) \setminus S \\
\sum_{(u,v) \in E(G)} y_{(u,uv)} &= 2, \forall u \in S \\
x_e &\leq 1, \forall e \in E(G) \\
y_e &\geq 0, \forall e \in E(H) \\
y_e &\in \{0, 1\}, \forall e \in E(H)
\end{aligned} \tag{3.5}$$

Les deux premiers ensembles d'égalités correspondent aux contraintes de degrés des sommets du type uv . Les troisième et quatrième imposent que les sommets de $V(G)$ soient de degré de 2.

Depuis les travaux de [38], on sait que les contraintes d'intégrité peuvent être remplacées par les inégalités dites de blossom. On considère un sous-ensemble $A = B \cup C \subset V(H)$ où $B \subset V(G)$ et $C \subset V(H) \setminus V(G)$ est un sous-ensemble de sommets de type uv . Soit F un sous-ensemble d'arêtes $F \subset E(A, V(H) \setminus A)$ tel que $|C| + |F|$ est impaire. Alors les inégalités blossom seront données par : $y(A) + y(F) \leq |B| + \lfloor \frac{|C| + |F|}{2} \rfloor$.

Il n'est pas difficile de vérifier que ces inégalités de blossom où F n'est pas vide sont dominées ou équivalentes aux inégalités où F est vide. Par exemple, si $uv \in A$ et il existe au moins une arête e incidente à uv qui est dans F , alors en supprimant uv de A et e de F , le membre droit $|B| + \lfloor \frac{|C| + |F|}{2} \rfloor$ ne croit pas et que le membre gauche $y(A) + y(F)$ ne décroît pas.

Finalement, pour obtenir une formulation linéaire étendue, on a juste à relaxer les contraintes d'intégrité et ajouter des contraintes du type

$$y(B \cup C) \leq |B| + \lfloor \frac{|C|}{2} \rfloor, B \subset V(G), C \subset V(H) \setminus V(G). \tag{3.6}$$

Les inégalités (3.6) peuvent se séparer par l'algorithme de [109]. Un autre algorithme [87], plus efficace peut aussi être utilisé. Un exemple est donné dans la figure (3.2) où une instance Euclidienne avec 150 noeuds est prise. Dans le premier cas (la partie gauche de la figure (3.2)), S contient 70 noeuds représentés par des carrés noirs. Le problème est alors résolu en utilisant la formulation étendue. On obtient une solution où le coût est approximativement égal à 852. Lorsque on relaxe les contraintes de Steiner, on obtient une solution tracée dans la partie droite de la figure (3.2), et le coût est de 759 approximativement.

3.4 Les inégalités valides

Le fait que le problème d'optimisation soit facile à résoudre laisse espérer l'existence d'une bonne description de l'enveloppe convexe des vecteurs d'incidence de ces réseaux. Par équivalence entre séparation et optimisation [60], on sait qu'il est possible de séparer en temps polynomial tout point rationnel et non réalisable de cette enveloppe convexe.

Une façon d'obtenir cette caractérisation consiste à projeter la formulation linéaire éten-

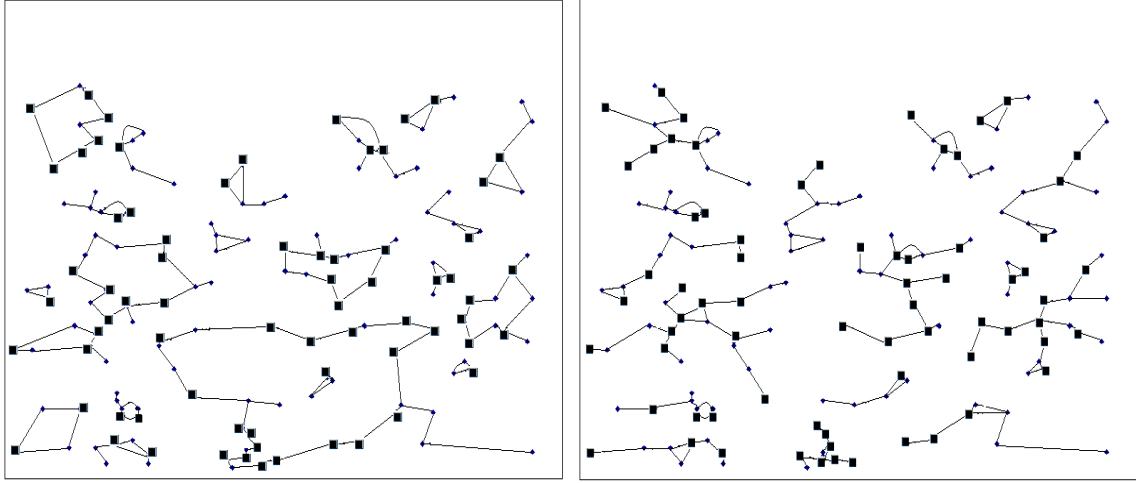


FIGURE 3.2 – Instance Euclidienne : $|V| = 150$: $|S| = 70$ de coût 852 dans la partie gauche, et $S = \emptyset$ de coût 759 dans la partie droite.

due de la Section 3.3 dans l'espace des variables $(x_e)_{e \in E(G)}$ en utilisant la dualité. Cependant, ceci reste une caractérisation compliquée. Alors, on essaiera de trouver directement quelques classes d'inégalités valides. Leur séparation sera étudiée dans la Section 3.5.

En plus des inégalités (3.3), on propose dans cette section deux autres familles d'inégalités valides.

$$x(A) + x(F) \leq |A| + \lfloor \frac{|F|}{2} \rfloor, \forall A \subset V, F \subset E(A, S \setminus A). \quad (3.7)$$

Proposition 3.3: Les inégalités (3.7) sont valides.

Preuve:

En ajoutant les inégalités $x(A) + \frac{1}{2}x(A, S \setminus A) \leq |A|$ et $\frac{1}{2}x(F) \leq \frac{1}{2}|F|$ et en utilisant le fait que $F \subset E(A, S \setminus A)$, on obtient $x(A) + x(F) \leq |A| + \frac{1}{2}|F|$. En prenant la partie entière inférieure, on obtient le résultat voulu. \square

Lorsque les inégalités (3.7) sont considérées, on doit supposer que $|F|$ est impair (sinon elles sont dominées par les inégalités (3.3) et les inégalités triviales).

On note que lorsque $S = V$, les inégalités (3.7) et (3.3) en plus des inégalités triviales sont suffisantes pour décrire l'enveloppe convexe. En effet, elles sont équivalentes à la formulation connue du problème de 2-couplage parfait [38, 80, 117].

Quand $S \neq V$, ces inégalités sont en général non suffisantes. On introduit ci-dessous une nouvelle famille d'inégalités valides.

Soient A et B deux ensembles disjoints de sommets. Leur union est notée par C : $C = A \cup B$. Soit F un ensemble d'arêtes tel que $F \subset E(B, S \setminus C) \cup E(A, S \cap B)$. Soit alors l'inégalité suivante :

$$2x(A) + x(A, S \setminus C) + x(B) + x(A, B) + x(F) \leq 2|A| + |B| + \lfloor \frac{|F|}{2} \rfloor. \quad (3.8)$$

Proposition 3.4: Les inégalités (3.8) sont valides.

Preuve: En écrivant l'inégalité (3.3) pour C , on obtient

$$x(A) + x(B) + x(A, B) + \frac{1}{2}x(A, S \setminus C) + \frac{1}{2}x(B, S \setminus C) \leq |A| + |B|.$$

en faisant idem pour l'ensemble A , on obtient

$$x(A) + \frac{1}{2}x(A, S \setminus A) \leq |A|.$$

En additionnant ces deux inégalités et en utilisant le fait que $S \setminus A = (S \setminus C) \cup (S \cap B)$ induit à

$$2x(A) + x(A, S \setminus C) + x(B) + x(A, B) + \frac{1}{2}x(B, S \setminus C) + \frac{1}{2}x(A, S \cap B) \leq 2|A| + |B|.$$

En prenant un sous-ensemble d'arêtes F dans $E(B, S \setminus C) \cup E(A, S \cap B)$, et en ajoutant les inégalités triviales $\frac{1}{2}x(F) \leq \frac{1}{2}|F|$, on obtient par simple arrondi les inégalités (3.8). \square

Si A est vide, les inégalités (3.8) deviennent les inégalités (3.7). De plus, si B est vide, alors l'inégalité (3.8) est la contrainte basique (3.3). Si $|F|$ est pair, alors l'inégalité (3.8) est dominée par les inégalités (3.3) et les inégalités triviales $x_e \leq 1$. En d'autres termes, lorsque nous considérons les inégalités (3.8), on peut supposer que $|A| > 0$ et $|F|$ est impaire (Ceci implique que $|B| > 0$).

Malheureusement, il est facile de vérifier que les inégalités (3.3), (3.7) et (3.8) ne sont pas suffisantes pour décrire l'enveloppe convexe. Le point fractionnaire de la figure (3.3) où S contient les sommets a et b et les valeurs de x sont semi-entières, est un point extrême du polytope défini par les inégalités (3.3), (3.7) et (3.8) en plus des inégalités triviales.

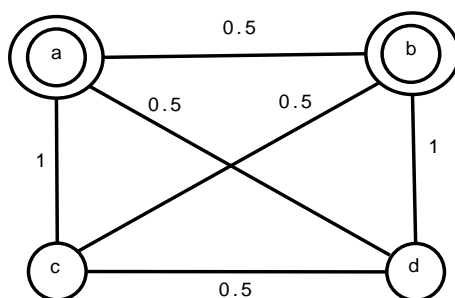


FIGURE 3.3 – Un point extremum fractionnaire qui satisfait les inégalités (3.3), (3.7) et (3.8)

Trouver une description polyédrale complète de l'enveloppe convexe des vecteurs d'incidence de ce type de réseaux reste donc un problème ouvert.

3.5 Les algorithmes de séparation

3.5.1 Séparation des contraintes (3.3)

On montre dans cette section que les contraintes (3.3) peuvent se séparer en un temps polynomial.

Proposition 3.5: Les inégalités (3.3) peuvent se séparer en un temps polynomial.

Preuve: Nous cherchons un sous-ensemble A qui maximise : $2(x(A) - |A|) + x(A, S \setminus A)$. Ceci est équivalent à minimiser la fonction $f(A) = \sum_{v \in A} (2 - x(\delta(v))) + x(A, S \setminus A)$. f peut se

réécrire comme suit :

$$\begin{aligned}
f(A) &= \sum_{v \in A} (2 - x(\delta(v))) + x(A, \bar{A} \setminus S) \\
&= \sum_{v \in A; 2-x(\delta(v)) \geq 0} (2 - x(\delta(v))) + \sum_{v \in A; 2-x(\delta(v)) < 0} (2 - x(\delta(v))) \\
&\quad - \sum_{v \in V; 2-x(\delta(v)) < 0} (2 - x(\delta(v))) + \sum_{v \in V; 2-x(\delta(v)) < 0} (2 - x(\delta(v))) + x(A, \bar{A} \setminus S) \\
&= x(A, \bar{A} \setminus S) + \sum_{v \in A; 2-x(\delta(v)) \geq 0} (2 - x(\delta(v))) + \sum_{v \in \bar{A}; x(\delta(v)) - 2 > 0} (x(\delta(v)) - 2) \\
&\quad + \sum_{v \in V; 2-x(\delta(v)) < 0} (2 - x(\delta(v)))
\end{aligned}$$

On note que le dernier terme $\sum_{v \in V; 2-x(\delta(v)) < 0} (2 - x(\delta(v)))$ ne dépend pas de A . Un sous-ensemble A minimisant f peut être calculé en résolvant un problème de coupe minimum dans un graphe orienté approprié. On construit un graphe orienté et pondéré qui a le même ensemble de sommets V en plus de 2 nouveaux sommets s et t . Si $2 - x(\delta(v)) \geq 0$, alors on ajoute un arc de v à t dont le poids est égal à $2 - x(\delta(v))$. Sinon, on ajoute un arc de s à v avec un poids égal à $x(\delta(v)) - 2$. Pour toute arête $(u, v) \in E$, on crée deux nouveaux arcs à sens opposés : un de u vers v et un autre de v vers u . L'arc de u vers v aura un poids égal à x_{uv} si $v \notin S$, et 0 sinon. \square

3.5.2 Séparation des contraintes (3.7)

Si $S = V$, les contraintes (3.7) sont les contraintes classiques de blossom introduites dans le contexte du 2-couplage (voir, e.g., [117, 80]). Padberg et Rao [109] ont montré que les inégalités dites de blossom peuvent se séparer en temps polynomial en résolvant un problème de T -coupe de poids minimum dans un graphe non orienté.

On va montrer que les contraintes (3.7) peuvent se séparer facilement si $|S| < |V|$. Le problème de séparation se réduit à un problème de $s - t$ T -coupe dans un graphe orienté. La preuve qu'on donne ici est une adaptation de la preuve donnée dans [109]. On essayera de suivre la présentation donnée dans [80].

Proposition 3.6: Les inégalités (3.7) peuvent se séparer en un temps polynomial.

Preuve: Dans cette preuve, on cherche un ensemble de sommets A et un ensemble impair d'arêtes $F \subset E(A, S \setminus A)$ minimisant $\sum_{v \in A} 2 - x(\delta(v)) + x(\delta(A) \setminus F) + (1 - x)(F)$. Comme les contraintes (3.3) sont faciles à séparer, on peut supposer qu'elles sont toujours satisfaites par le vecteur courant x . Toutes les inégalités triviales sont aussi satisfaites.

En utilisant les mêmes manipulations algébriques lors de la section précédente, la fonction objectif devient $\sum_{v \in A; 2-x(\delta(v)) \geq 0} (2 - x(\delta(v))) + \sum_{v \in \bar{A}; x(\delta(v)) - 2 > 0} (x(\delta(v)) - 2) + x(\delta(A) \setminus F) + (1 -$

$$\begin{aligned}
&x)(F) + \\
&\quad \sum_{v \in V; 2-x(\delta(v)) < 0} (2 - x(\delta(v))).
\end{aligned}$$

On note encore que $\sum_{v \in V; 2-x(\delta(v)) < 0} (2 - x(\delta(v)))$ est une constante. On utilise K pour noter

$$\sum_{v \in V; 2-x(\delta(v)) < 0} (x(\delta(v)) - 2).$$

Les inégalités (3.7) sont violées si et seulement si $|F|$ est impaire et

$$\sum_{v \in A; 2-x(\delta(v)) \geq 0} (2-x(\delta(v))) + \sum_{v \in \bar{A}; x(\delta(v))-2 > 0} (x(\delta(v))-2) + x(\delta(A) \setminus F) + (1-x)(F) < 1+K. \quad (3.9)$$

Notons que comme les inégalités (3.3) sont satisfaites, on a toujours pour tout A et F (sans aucune restriction sur la parité de F) :

$$\sum_{v \in A; 2-x(\delta(v)) \geq 0} (2-x(\delta(v))) + \sum_{v \in \bar{A}; x(\delta(v))-2 > 0} (x(\delta(v))-2) + x(\delta(A) \setminus F) + (1-x)(F) \geq K.$$

On construit un graphe orienté et pondéré $H = (V(H), E(H))$. Le poids d'un arc e est noté par y_e . $V(H) = V(G) \cup E(G) \cup \{s, t\}$ où s et t sont deux nouveaux sommets. Pour tout sommet $v \in V(G)$ pour lequel $x(\delta(v)) - 2 > 0$, $E(H)$ contient un arc $[s, v]$ de poids égal à $y_{[s,v]} = x(\delta(v)) - 2$. D'une façon similaire, si $2 - x(\delta(w)) \geq 0$ pour un sommet $w \in V(G)$, alors on considère un arc $[w, t]$ dont le poids est égal à $2 - x(\delta(w))$.

Les arêtes de $E(G)$ doivent avoir une orientation. Si $e = (u, v) \in E(G)$ a une extrémité $u \in S$ et une extrémité $v \in V \setminus S$, alors e est orientée de u à v . Pour les autres arêtes, l'orientation est arbitraire.

Les arêtes de $E(G)$ incluses dans $V(H)$ sont considérées comme des arcs (comme elles sont orientées). Soient u et v deux sommets adjacents de $V(G)$ (donc ils sont aussi sommets de H), et supposons que (u, v) est orienté de u à v ($[u, v] \in V(H)$). On considère trois cas possibles :

- si $u \notin S$ et $v \notin S$, alors $E(H)$ contient un arc $[u, [u, v]]$ de poids égal à x_{uv} , un arc $[[u, v], u]$ de poids égal à x_{uv} , un arc $[v, [u, v]]$ de poids égal à $+\infty$ et un arc $[[u, v], v]$ de poids égal à $+\infty$;
- si $u \in S$ et $v \notin S$, alors $E(H)$ contient un arc $[u, [u, v]]$ dont le poids est de x_{uv} , un arc $[[u, v], u]$ de poids égal à x_{uv} , un arc $[v, [u, v]]$ de poids égal à $1 - x_{uv}$ et un arc $[[u, v], v]$ de poids égal à $+\infty$;
- si $u \in S$ et $v \in S$, alors $E(H)$ contient un arc $[u, [u, v]]$ de poids égal à x_{uv} , un arc $[[u, v], u]$ de poids égal à x_{uv} , un arc $[v, [u, v]]$ de poids égal à $1 - x_{uv}$ et un arc $[[u, v], v]$ de poids égal à $1 - x_{uv}$.

Il n'est pas possible d'avoir $u \notin S$ et $v \in S$ comme (u, v) est orienté de u à v .

Soit T un sous-ensemble de $V(H)$ défini par :

$$T = \{[u, v] \in V(H), \{u, v\} \cap S \neq \emptyset\} \cup \{u \in V \setminus S, |E^+(S, u)| \text{ est impair}\} \cup \{u \in S, |E^+(S \setminus u, u)| \text{ est impair}\}.$$

Si on prend $E^+(S \setminus u, u) = E^+(S, u)$ lorsque $u \in S$, T devient $T = \{[u, v] \in V(H), \{u, v\} \cap S \neq \emptyset\} \cup \{u \in V, |E^+(S, u)| \text{ est impair}\}$. En d'autres mots, T contient tous les arcs qui ont une extrémité dans S , et aussi tous les sommets u tels que le nombre d'arcs de S à u est impair. Il est clair que T a une cardinalité paire.

On montrera que le graphe H contient une $s-t$ T -coupe de poids inférieur à $1+K$ si et seulement si il existe une contrainte violée de type (3.7). Soient $A \subset V$ et $F \subset E(A, S \setminus A)$. On a besoin de la notation suivante :

$$E_1 = \{[u, v] \in V(H) \cap F, u \in A, v \in V \setminus A\}; E_2 = \{[u, v] \in V(H) \cap F, u \in V \setminus A, v \in A\}$$

$$E_3 = \{[u, v] \in V(H) \setminus F, u \in A, v \in V \setminus A\}; E_4 = \{[u, v] \in V(H) \setminus F, u \in V \setminus A, v \in A\}.$$

Notons que $F = E_1 \cup E_2$, $E^+(A, V \setminus A) = E_1 \cup E_3$ et $E^+(V \setminus A, A) = E_2 \cup E_4$. Soit $W \subset V(H)$

défini par $W = A \cup E(A) \cup E_1 \cup E_4 \cup \{s\}$. On veut montrer que $|W \cap T|$ est impaire si et seulement si $|F|$ est impaire. On utilisera la notation \equiv pour dire que les deux nombres ont la même parité.

On a $|W \cap \{u \in V \setminus S, |E^+(S, u)| \text{ est impaire}\}| \equiv |E^+(S, A \setminus S)|$. On peut aussi écrire $|W \cap \{u \in S, |E^+(S \setminus u, u)| \text{ est impaire}\}| \equiv |E(A \cap S)| + |E^+(S \setminus A, S \cap A)|$. De plus, $E_1 \subset F$ et $F \subset T$. En prenant les arcs de $E(A)$ appartenant à T , on déduit que $|E(A) \cap T| \equiv |E^+(A \cap S, A \setminus S)| + |E(A \cap S)|$. D'une façon similaire, on a $|E_4 \cap T| \equiv |E_4 \cap E^+(S \setminus A, A)|$ (on rappelle que tous les arcs sont orientés de S à $V \setminus S$). En combinant tous ces résultats de parité, on obtient la conséquence suivante : $|W \cap T| \equiv |E^+(S \setminus A, A)| + |E_1| + |E_4 \cap E^+(S \setminus A, A)|$. De plus, on a $E_4 \cup E_2 = E^+(S \setminus A, A) \cup E^+(\bar{A} \setminus S, A \setminus S)$. Alors, $|E_4 \cup E_2| \equiv |E^+(S \setminus A, A)| + |E^+(\bar{A} \setminus S, A \setminus S)|$. En réalisant une intersection entre les deux termes avec E_4 , on déduit que $|E_4| \equiv |E^+(S \setminus A, A) \cap E_4| + |E^+(\bar{A} \setminus S, A \setminus S) \cap E_4|$. Observons que $E^+(\bar{A} \setminus S, A \setminus S)$ ne va pas faire intersection avec F comme $F \subset E(A, S \setminus A)$ alors que tous les arcs de $E^+(\bar{A} \setminus S, A \setminus S)$ n'a pas d'extrémité dans S . Par conséquent, on a $|E_4| \equiv |E^+(S \setminus A, A) \cap E_4| + |E^+(\bar{A} \setminus S, A \setminus S)|$. En combinant cette égalité avec celle reliée à $|E_4 \cup E_2|$, on obtient $|E_2| \equiv |E^+(S \setminus A, A) \cap E_4| + |E^+(S \setminus A, A)|$. En revenant sur $|W \cap T|$, on déduit que

$$|W \cap T| \equiv |E_1| + |E_2| = |F|. \quad (3.10)$$

Maintenant, on montre que le poids de la coupe $\delta^+(W)$ est donné par : $y(\delta^+(W)) = \sum_{v \in A; 2-x(\delta(v)) \geq 0} (2-x(\delta(v))) + \sum_{v \in \bar{A}; x(\delta(v))-2 > 0} (x(\delta(v))-2) + x(\delta(A) \setminus F) + (1-x)(F)$.

$\delta^+(W)$ contient des arcs $[s, w]$ où $w \in V \setminus A$ et $x(\delta(v)) - 2 > 0$; et les arcs $[v, t]$ où $v \in A$ et $2 - x(\delta(v)) \geq 0$. Le poids total de ces arcs est exactement donné par $\sum_{v \in A; 2-x(\delta(v)) \geq 0} (2-x(\delta(v))) + \sum_{v \in \bar{A}; x(\delta(v))-2 > 0} (x(\delta(v))-2)$.

En utilisant le fait que $\delta(A) = E_1 \cup E_2 \cup E_3 \cup E_4$ et W contient E_1 et E_4 , on déduit que les autres arcs de $\delta^+(W)$ ont 4 formes possibles :

- $[u, [u, v]]$ où $u \in A, v \in V \setminus A$, et $[u, v] \notin W$ implique que $[u, v] \in E_3$ et $y_{[u, [u, v]]} = x_{uv}$
- $[[u, v], u]$ où $u \in V \setminus A, v \in A$ et $[u, v] \in W$ implique que $[u, v] \in E_4$ et $y_{[[u, v], u]} = x_{uv}$
- $[v, [u, v]]$ où $u \in V \setminus A, v \in A$ et $[u, v] \notin W$ implique que $[u, v] \in E_2$. Comme $E_2 \subset F$, on doit avoir $u \in S$ et $y_{[v, [u, v]]} = 1 - x_{uv}$.
- $[[u, v], v]$ où $u \in A, v \in V \setminus A$ et $[u, v] \in W$ implique que $[u, v] \in E_1$. Comme $E_1 \subset F$, on doit avoir $v \in S$ et $y_{[[u, v], v]} = 1 - x_{uv}$.

La combinaison de ces résultats nous donne :

$$y(\delta^+(W)) = \sum_{v \in A; 2-x(\delta(v)) \geq 0} (2-x(\delta(v))) + \sum_{v \in \bar{A}; x(\delta(v))-2 > 0} (x(\delta(v))-2) + x(\delta(A) \setminus F) + (1-x)(F). \quad (3.11)$$

La construction de W et les résultats (3.9), (3.10) et (3.11) impliquent clairement que tous les ensembles A et $F \subset E(A, S \setminus A)$ de cardinalité impaire et qui violent les contraintes (3.7) vont nous conduire à une $s-t$ T -coupe $\delta^+(W)$ dont le poids est strictement inférieur à $1 + K$.

On montre maintenant le sens opposé. On suppose qu'on a une $s-t$ T -coupe $\delta^+(W)$ telle que $y(\delta^+(W)) < 1 + K$, et construisons A et F qui violent les contraintes (3.7).

On prend $A = W \cap V$. Admettons que si $\delta^+(W)$ contient un des arcs $[u, [u, v]]$, $[v, [u, v]]$, $[[u, v], u]$, $[[u, v], v]$, alors $(u, v) \in \delta(A)$. Cette hypothèse sera montrée ci-dessous. On définit F ainsi :

$$F = \{(u, v), v \in A, u \in S \setminus A, [u, v] \in \bar{W}\} \cup \{(u, v), v \in S \setminus A, u \in S \cap A, [u, v] \in W\}.$$

A et F sont donnés, on peut définir E_1, E_2, E_3 et E_4 comme on a fait avant. On a à montrer que W est exactement donné par $A \cup E(A) \cup E_1 \cup E_4 \cup \{s\}$. Par l'hypothèse d'avant, W contient $E(A)$ et ne doit contenir aucune arête de $E(V \setminus A)$. Alors, on a juste à montrer que W contient $E_1 \cup E_4$, et ne contient pas $E_2 \cup E_3$.

- Si $[u, v] \in E_1$, alors $u \in A$ et $v \in V \setminus A$. En plus, $[u, v] \in F$ qui implique avec la définition de F que $[u, v] \in W$.
- Si $[u, v] \in E_4$, alors $v \in A$ et $u \in V \setminus A$. Si $u \in S$, alors $[u, v] \in W$ (sinon $[u, v]$ doit être dans F). Si $u \notin S$, alors $v \notin S$ (comme (u, v) est orienté de u à v) et le poids de $[v, [u, v]]$ est égal à l'infini. Ceci est bien sur impossible comme $y(\delta^+(W)) < 1 + K$.
- Si $[u, v] \in E_2$, alors $v \in A$ et $u \in V \setminus A$. Comme $E_2 \subset F$, par la définition de F on doit avoir $[u, v] \in \overline{W}$.
- Si $[u, v] \in E_3$, alors $u \in A$ et $v \in V \setminus A$. Il y a 2 sous cas possibles. Lorsque $u \in S \cap A$ et $v \in S \setminus A$, alors $[u, v] \in \overline{W}$ (car $[u, v] \notin F$). L'autre sous cas est donné par $u \in A$ et $v \in \overline{A} \setminus S$. Comme le poids de $[[u, v], v]$ est infini, $[u, v]$ doit être dans \overline{W} .

Ainsi on a montré que $W = A \cup E(A) \cup E_1 \cup E_4 \cup \{s\}$. Par les résultats (3.9), (3.10) et (3.11), et en utilisant le fait que W est une $s-t$ T -coupe de poids inférieur à $1 + K$, on déduit que $|F|$ est impaire et l'inégalité (3.7) correspondant aux sous-ensembles A et F est violée.

Pour terminer la preuve, on a à revenir à notre hypothèse prise avant : si $\delta^+(W)$ contient un des arcs $[u, [u, v]]$, $[v, [u, v]]$, $[[u, v], u]$, $[[u, v], v]$, alors $(u, v) \in \delta(A)$. Supposons que l'hypothèse est fautive, alors en déplaçant les arcs $[u, v]$ qui violent cette hypothèse, on obtient un ensemble W' qui satisfait l'hypothèse. De plus, on a $y(\delta^+(W')) \leq y(\delta^+(W)) - 1$. En utilisant W' , on peut définir A', F', E'_1, E'_2, E'_3 et E'_4 comme on l'a déjà fait (en effet $A' = A$). Alors on a toujours $W' = A' \cup E(A') \cup E'_1 \cup E'_4 \cup \{s\}$. On note aussi que dans la preuve précédente, on n'a pas utilisé le fait que W définit une T -coupe. C'est pour cela que si W' ne définit pas une $s-t$ T -coupe, on peut toujours écrire

$$y(\delta^+(W')) = \sum_{v \in A'; 2-x(\delta(v)) \geq 0} (2-x(\delta(v))) + \sum_{v \in \overline{A}'; x(\delta(v))-2 > 0} (x(\delta(v))-2) + x(\delta(A') \setminus F') + (1-x)(F').$$

Comme

$$y(\delta^+(W')) \leq y(\delta^+(W)) - 1$$

et

$$y(\delta^+(W)) < 1 + K$$

on a

$$\sum_{v \in A'; 2-x(\delta(v)) \geq 0} (2-x(\delta(v))) + \sum_{v \in \overline{A}'; x(\delta(v))-2 > 0} (x(\delta(v))-2) + x(\delta(A') \setminus F') + (1-x)(F') < K$$

Mais on a toujours mentionné que ceci ne peut se produire sauf si $|F'|$ est paire.

Pour synthétiser, on a montré qu'étant donnée toute contrainte violée du type (3.7), on peut construire une $s-t$ T -coupe dont le poids est strictement inférieur à $1 + K$. On a aussi montré qu'étant donnée une $s-t$ T -coupe dont le poids est inférieur à $1 + K$, on peut alors construire une contrainte violée du type (3.7).

Comme une $s-t$ T -coupe de poids minimum peut être calculée en temps polynomial [60, 49], les contraintes (3.7) peuvent se séparer en temps polynomial. \square

Pour trouver une $s-t$ T -coupe de poids minimum dans le graphe H , on note que les deux

algorithmes [60] et [49] sont basés sur la solution d'approximativement $|V(H)|^2$ problèmes de $s - t$ coupe de poids minimum.

Focalisons nous sur l'algorithme de [49]. Pour tous $a \in V(H)$ et $b \in V(H)$, on ajoute deux arcs $[s, a]$ et $[b, t]$ de poids infinis, et on résout une $s - t$ -coupe de poids minimum (la coupe doit aussi être minimale au sens de la cardinalité). Une des $|V(H)||V(H) - 1|$ coupes sera la $s - t$ T -coupe de poids minimum.

La séparation des contraintes (3.8) est toujours un problème ouvert.

3.6 Extensions et liens avec d'autres problèmes

L'approche utilisée dans la Section 3.3 pour montrer que le problème de ce chapitre se résout polynomialement peut être légèrement modifiée pour intégrer de nouvelles contraintes.

On considère une variante dans laquelle il existe un sous-ensemble de sommets $T \subset V(G)$ tel que chaque sommet $u \in T$ doit être soit une feuille (sommet de degré 1) ou sur un cycle et ne doit pas avoir de voisins (sauf les deux voisins sur le cycle). Cette contrainte peut facilement être intégrée en supprimant l'arête (v, uv) pour tout $v \in V(G)$ lorsque $u \in T$. On note que si $u \in S \cap T$, alors le degré de u doit être égal à 2 et u est sur un cycle.

Une autre variante possible consiste à imposer que quelques arêtes doivent appartenir aux cycles. On suppose que (u, v) doit être sur un cycle. Ceci peut être réalisé en ajoutant un nouveau sommet a adjacent à u et v , supprimer (u, v) et mettre a dans S .

Il est possible qu'un ensemble d'arêtes appartienne à la solution. Ceci est facile à imposer en supprimant les arêtes (uv, vu) si on veut que (u, v) soit dans la solution.

On peut aussi imposer la contrainte suivante : soit l'arête (u, v) appartient au cycle, ou elle ne figure pas dans la solution. En d'autres mots, (u, v) ne doit pas figurer dans les sous arbres attachés aux cycles. Pour mieux cerner cette contrainte, on peut penser à la situation où toutes les arêtes qui ont un poids plus grand qu'un certain seuil, seront dans la solution si et seulement si elles contribuent à la construction des cycles. Cette contrainte peut être aussi satisfaite en supprimant les arêtes (u, vu) et (v, uv) .

On peut également supposer que chaque arête (u, v) a deux poids w_{uv}^1 et $w_{uv}^2 \geq w_{uv}^1$. Si l'arête appartient à un cycle, alors on a à payer w_{uv}^1 , sinon on doit payer w_{uv}^2 chaque fois que l'arête fait partie de la solution. Ce problème peut être facilement résolu par une légère modification sur les coûts. Le poids de chacune des arêtes (u, uv) et (v, vu) devient $\frac{w_{uv}^1}{2}$ alors que le poids de chacune des arêtes (u, vu) et (v, uv) est $w_{uv}^2 - \frac{w_{uv}^1}{2}$. En prenant $w_{uv}^2 = \infty$, on peut aussi résoudre la variante précédente.

En utilisant les mêmes techniques, il est facile de montrer que le problème de 2-couplage avec contrainte de type Steiner est facile à résoudre. Dans cette variante, un sommet est soit isolé ou appartient à un cycle. La solution est alors un ensemble de cycles disjoints couvrant un certain ensemble de sommets. Cependant, la résolution polynomiale de cette variante est un résultat bien connu [21].

3.7 Résultats numériques

Nous présentons dans cette section quelques résultats numériques. D'une part, nous avons implémenté un branch&cut utilisant la formulation (3.1) avec les inégalités (3.3) et (3.7). D'autre part, nous avons implémenté le programme linéaire étendu.

Toutes les expérimentations ont été réalisées sur un Pentium 4, de 3.25Go de RAM, et de 2.39 GHz.

Dans la première phase de l'algorithme du branch&cut (à la racine), les deux familles d'inégalités citées ci-dessus ont été intégrées. Uniquement les inégalités de base (3.3) sont rajoutées au niveau des autres noeuds de l'arbre de recherche. En effet, la séparation des inégalités (3.3) est exacte et rapide. Signalons qu'on a choisit la stratégie de recherche en profondeur d'abord dans l'algorithme de Branch&Cut, et une borne supérieure est calculée par la recherche d'un 2-couplage dans le graphe de départ.

TABLE 3.1 – Comparaison sur des graphes complets aléatoires et Euclidiens

n	S	temps sep	temps relax	temps tot	gap(%)	gap2(%) (3.3)	(3.7)	Nb coupes	Nb noeuds	temps form.etendue
6	1.7	0.3	2.7	0	9.4	21.7	21.7	7.3	9	1
25	12	3	0.3	6	0	16.4	31	31	22	7.7
	24	2.7	0	3	0	19.5	28	28	7.3	4.7
40	10	19	0.7	20.7	0	5.19	31.7	34	0.3	1
	20	13	0.7	35	0	10.4	30.7	31.7	36.3	29.3
	39	10	1.3	22	0	15.5	38	38.7	23.7	22.7
65	16	182.7	3.3	790.7	0	5.7	53	55	136.7	86.3
	32	303.3	14.3	1019.3	0	11.8	77.7	77.3	172.3	286.7
	64	50.7	2.3	125.3	0	13.9	34.3	33.3	29.7	32

La Table (3.1) nous donne les résultats trouvés sur des graphes complets, aléatoires et Euclidiens. Les coordonnées de chaque sommet sont choisies d'une manière aléatoire et uniforme entre 0 et 1. Le poids d'une arête est proportionnel à la distance entre les extrémités de cette arête.

On a généré pour les différentes valeurs de n et de $|S|$ trois instances, et les résultats présentés sont obtenus en calculant la moyenne de chaque cas traité.

Dans la Table (3.1), on peut trouver le temps en secondes de séparation de toutes les coupes (temps sep) dans la phase de l'algorithme à plans coupants (avant le Branch&Cut). Par (temps relax), on note le temps mis pour la résolution de la relaxation (avant le Branch&Cut), et (temps tot) est le temps total mis pour résoudre le problème jusqu'à la fin. On donne aussi le nombre de contraintes générées dans la phase de l'algorithme à plans coupants parmi les deux familles citées ci-dessus. Par "Nb coupes" et "Nb noeuds" on donne le nombre de coupes du type (3.3) générées dans la phase du Branch&Cut, ainsi que le nombre de noeuds de l'arborescence du Branch&Cut.

Le gap final, qui est la différence entre la borne supérieure et la meilleure solution trouvée, le tout divisé par la borne supérieure, est donné et noté par "gap". Si ce gap est égal à 0, alors le problème est résolu jusqu'à l'optimum. On donne aussi la valeur de "gap 2", qui représente la différence entre la meilleure borne supérieure et le coût obtenu avec $|S| = 0$, le tout sera divisé

sur la borne supérieure.

La dernière colonne de cette table représente le temps total mis par l'algorithme basé sur la formulation étendue.

On remarque que les temps d'exécution de l'algorithme de Branch&cut diminuent lorsque la taille de S se rapproche de $|V|$. Ceci est en réalité la conséquence de la bonne qualité de la borne supérieure donnée par la solution du 2-couplage.

On remarque aussi que la formulation étendue est largement meilleure que le branch&cut. Ceci est évidemment prévisible vu l'exactitude de cette formulation.

3.8 Conclusion

Dans ce chapitre, nous avons présenté un algorithme polynomial et une formulation linéaire étendue du problème de synthèse de réseaux de coût minimum à composantes connexes unicycliques satisfaisant une contrainte de type Steiner. Une description partielle de l'enveloppe convexe des vecteurs d'incidence de ces réseaux a été donnée. Un problème ouvert intéressant consiste donc à compléter cette description partielle afin de caractériser ces vecteurs d'incidence dans leur espace naturel.

Une autre direction de recherche qu'on peut envisager pour le futur consiste à montrer qu'il y a bien d'autres problèmes de Steiner relatifs à d'autres matroïdes transversaux qu'on peut résoudre facilement.

Finalement, on a aussi donné une modification non triviale de l'algorithme de Padberg&Rao [109] pour séparer une classe d'inégalités valides basée sur la solution du problème de la $s - t$ T -coupe dans un graphe orienté. Il serait alors intéressant de généraliser notre algorithme à d'autres problèmes.

Réseaux à composantes unicycliques avec de nouvelles contraintes techniques

L'objet de ce chapitre est d'intégrer d'autres types de contraintes. Nous souhaitons toujours obtenir un réseau à composantes connexes unicycliques. Nous considérons dans un premier temps des contraintes de degré. Ensuite, on intègre une contrainte sur le nombre de composantes connexes. Enfin, nous décrivons comment on peut intégrer des contraintes d'appartenance de sommets à une même composante connexe ainsi que des contraintes de séparation de certains sommets. Quelques résultats expérimentaux seront également présentés. Signalons que l'étude faite dans ce chapitre est très préliminaire.

4.1 Introduction

Les travaux exposés dans les deux précédents chapitres ont permis d'intégrer des contraintes techniques importantes en plus de la contrainte d'unicyclicité : taille minimale des cycles, et contraintes de type Steiner. Nous continuons sur la même lancée en intégrant d'autres contraintes. Signalons cependant que l'étude faite dans ce chapitre reste superficielle puisque nous n'avons pas pu faire une étude polyédrale poussée. Nous espérons pouvoir approfondir ce travail ultérieurement.

Notre ingrédient de base est encore la formulation mathématique issue de la transversalité du matroïde bi-circulaire.

$$\begin{aligned}
 & \min \sum_{e \in E} w_e x_e \\
 & x_e = y_{ie} + y_{je}, \forall e = (i, j) \in E \\
 & \sum_{e=(i,j) \in E} y_{ie} = 1, \forall i \in V \\
 & 0 \leq y_{ie}, \forall i \in V, e = (i, j) \in E \\
 & x_e \leq 1, \forall e \in E.
 \end{aligned} \tag{4.1}$$

Nous allons dans la suite présenter quelques contraintes techniques et montrer comment elles sont intégrées. Quelques résultats numériques seront également exposés.

Notons qu'en ce qui concerne les résultats expérimentaux qui vont suivre pour chaque cas, nous avons implémenté un algorithme à plans coupants suivi d'un algorithme de branch&cut basé sur les fonctions callbacks de [74]. Toutes les expérimentations ont été réalisées sur un Pentium 4, de 3.25 Go de RAM, et de 2.39 GHz de fréquence. On note aussi qu'une stratégie de recherche en profondeur d'abord a été choisie dans l'algorithme de branch&cut. Dans certains cas, une borne supérieure est calculée lorsque c'est possible.

4.2 Contraintes portant sur les degrés des sommets

Les noeuds du réseau peuvent avoir des rôles différents. Certains peuvent être des noeuds d'interconnexion avec de multiples voisins alors que d'autres doivent avoir un nombre très limité de voisins. Ce type de contraintes est évidemment très simple à intégrer.

Pour borner alors le degré du sommet a , il suffit d'écrire que

$$\underline{d}_a \leq \sum_i (y_{ai} + y_{ia}) \leq \overline{d}_a \quad (4.2)$$

où \underline{d}_a et \overline{d}_a représentent les bornes inférieures et supérieures du degré du sommet a respectivement.

Ces contraintes peuvent être renforcées à l'aide des inégalités classiques de type Blossom (provenant du b -couplage).

Signalons que nous ne connaissons toujours pas la complexité du problème de synthèse d'un réseau à composantes connexes unicycliques avec contraintes de degré.

4.3 Contrainte portant sur le nombre de composantes unicycliques

Lorsque on partitionne un réseau (graphe) donné par $G = (V, E)$ en composantes connexes et unicycliques, on peut retrouver un nombre de composantes relativement élevé. Ceci pourrait être gênant dans un contexte pratique.

Dans le chapitre 2, on a vu une manière pour y remédier en imposant une certaine contrainte sur la taille des cycles ce qui implique immédiatement une borne inférieure du nombre de noeuds par composante connexe.

Dans ce paragraphe, nous allons directement limiter le nombre de composantes unicycliques.

Pour ce faire, nous utilisons une variante des contraintes valides (2.6) présentées au chapitre 2.

Supposons que z représente le nombre maximal de composantes unicycliques connexes à ne pas dépasser. Soit $(\alpha_e)_{e \in E}$ un ensemble de nombres réels. Pour tout nombre $0 \leq i \leq z - 1$, γ_{n+i} est le coût minimum (au sens des $(\alpha_e)_{e \in E}$) d'un graphe connexe avec $n + i$ arêtes. Soit T_α un arbre couvrant de poids minimum et β_i est le coût maximum des i arêtes de T_α . Autrement dit,

$$\beta_i = \max_{E' \subseteq T_\alpha, |E'|=i} \sum_{e \in E'} \alpha_e.$$

Notons aussi que γ_{n+i} ainsi que β_i peuvent être calculés d'une manière gloutonne comme on peut le voir dans la section (2.5.4), et cela pour tout nombre i (voir chapitre 2).

Proposition 4.1: L'inégalité suivante est alors valide pour le problème posé ci-dessus :

$$\sum_{e \in E} \alpha_e x_e \geq \min_{0 \leq i \leq z-1} (\gamma_{n+i} - \beta_i). \quad (4.3)$$

Plus de détails sur la séparation de cette famille d'inégalités ont été donnés au chapitre 2.

Remarque : Si z est fixé à l'avance à 1, on retrouve alors le cas de la recherche d'un graphe connexe et unicyclique.

Notons également que lorsque la solution est entière, les inégalités (4.3) sont suffisantes pour garantir que le nombre de composantes connexes est au plus z . Cependant, les inégalités sont insuffisantes pour décrire l'enveloppe convexe des réseaux respectant la contrainte.

Le Tableau (4.1) résume quelques résultats numériques obtenus. Le contenu du tableau est similaire à celui des chapitres précédents.

On a limité le temps total d'exécution à 7200 secondes. Le nombre maximum de composantes unicycliques à obtenir varie de 1 à 3 (sauf pour $n = 20$, cette instance donne un partitionnement initial limité à 2 composantes unicycliques au plus). Si ce nombre est de 1, alors on utilise la résolution de l'algorithme glouton donné dans le chapitre 1, ce qui explique les temps de calcul nuls dans ce cas. Les instances traitées relèvent de graphes aléatoires et Euclidiens générés 3 fois. Les résultats présentés sont des moyennes sur les 3 instances.

On voit que malgré la pauvreté de notre modèle polyédral, on arrive à traiter des instances de tailles moyennes. Pour les instances de tailles importantes, nous avons besoin d'améliorer la relaxation linéaire.

TABLE 4.1 – Graphes aléatoires complets Euclidiens - Nombre Max de Composantes

n	Nb Max	sep	relax	temps tot	gap(%)	gap 2(%)	(4.3)	Nb coupes	Nb noeuds
20	1	0	0	0	0	3.7	0	0	1
	2	4.7	0	5.7	0	1.8	14.7	0	1
35	1	0	0	0	0	7.9	0	0	1
	2	403.3	8.3	414	0	5.06	103	0	1
	3	383.7	4.3	389.3	0	2.6	73.7	0	1
45	1	0	0	0	0	7.8	0	0	1
	2	2782	37	2825	0.9	6.3	135.7	0	1
	3	1491.3	12.7	1507.3	0	3.7	94	0	1
60	1	0	0	0	0	6.1	0	0	1
	2	7141	59	7200	2.8	6.1	222	0	1
	3	4945.7	74.3	5029.3	3	5.2	75.7	0	1
100	1	0	0	0	0	5.3	0	0	1
	2	7187	13	7200	3.4	5.3	37.7	0	1
	3	7192.7	7.3	7200	3.8	5.3	32.7	0	1

4.4 Séparation de certains couples de sommets

On peut exiger que certains sommets soient dans des composantes connexes différentes.

Supposons par exemple qu'on cherche à séparer a et b . Le vecteur d'incidence doit donc vérifier l'inégalité suivante :

$$\sum_{e \in P_{ab}} (1 - x_e) \geq 1, \quad (4.4)$$

où P_{ab} est un chemin reliant a et b .

Ces inégalités se séparent simplement. Il suffit en effet de calculer un plus court chemin entre a et b au sens des poids $1 - x_e$ et de rajouter l'inégalité si le poids du chemin est strictement inférieur à 1.

Signalons que les inégalités (4.4) sont suffisantes pour garantir la séparation de a et b lorsque la solution est entière.

Pour tester l'efficacité de ces inégalités, on a réalisé quelques expérimentations numériques. Pour plusieurs valeurs de n (nombre de noeuds), on génère des instances Euclidiennes aléatoires où on cherche à séparer certains couples de sommets. Pour chaque valeur de n et chaque valeur du nombre de couples à séparer, nous générons 3 instances. Les résultats présentés sont donc les moyennes des résultats des 3 instances.

Il est important de mentionner que les couples de sommets choisis étaient bien sur la même composante unicyclique au départ (lorsqu'on résout le problème non contraint).

Les résultats présentés dans le tableau 4.2 semblent montrer que les inégalités (4.4) permettent de résoudre des instances de grandes tailles. Notons cependant que le nombre de noeuds dans l'arbre de recherche peut devenir assez élevé. La difficulté du problème semble augmenter avec le nombre de couples à séparer.

4.5 Joindre certains couples de sommets

Dans certains cas, on peut exiger que certains sommets soient sur une même composante connexe afin de faciliter la communication entre ces noeuds.

Supposons par exemple que a et b doivent être sur la même composante. Il est alors clair que toutes les coupes $\delta(S)$ qui séparent a et b ($a \in S$ et $b \notin S$) doivent vérifier :

$$x(\delta(S)) \geq 1 \quad (4.5)$$

La séparation de ces inégalités se fait évidemment par le calcul d'une coupe minimum séparant a et b . Pour ce faire, nous utilisons l'algorithme de Goldberg et Tarjan ([50]).

Notons que les inégalités (4.5) sont suffisantes pour garantir que a et b soient dans la même composante.

Il est facile de prouver que le problème de synthèse d'un réseau à composantes connexes unicycliques respectant des contraintes d'appartenance de certains sommets à une même composante connexe est un problème NP-dur. Une réduction à partir de l'arbre de Steiner est quasi-immédiate.

Quelques résultats numériques sont résumés dans le tableau 4.3. Les expériences ont été conduites d'une manière similaire à ce qui a été fait précédemment : instances Euclidiennes, plusieurs valeurs pour le nombre de couples à séparer, 3 instances pour chaque variante, un algorithme à plans coupants suivi d'un Branch&Cut.

TABLE 4.2 – Graphes aléatoires complets Euclidiens, Séparer a et b

n	Nb cpl	sep	relax	temps tot	gap(%)	gap 2(%)	(4.4)	Nb coupes	Nb noeuds
10	1	0	0	0	0	10.1	2.7	17.7	19.3
	2	0	0	0	0	22.8	9.3	27.7	19
35	1	0	0	0	0	4.4	4.3	2	1
	2	0	0	0.3	0	9.1	6.7	8.3	8.3
	3	0	0	0	0	11.7	10.3	8.3	8.7
	10	1	0	2.7	0	25.5	38	25.7	26
	33	2.7	0	228	0	32.3	60.7	284.3	2288.7
45	1	0	0	0.3	0	1.2	2.3	16.3	32
	2	0	0	0	0	5.3	5	1	1.3
	3	0.7	0	0.7	0	3	5.7	0.7	1
	10	1	0	31.7	0	16.5	27	72	510.7
	33	4.7	0.3	960	0	35.4	85.3	350	5745
50	1	0.3	0	0.3	0	1.6	3.7	1.7	2.7
	2	0	0	0	0	2.9	3.7	0	1
	3	0	0	0	0	4.4	5	0.7	1
	10	1	0	2.7	0	11.7	20.3	12.3	3.7
	33	6.3	0	230	0	30.9	90.7	167.7	908.3
65	1	0	0	0.3	0	0.96	2.3	3	3.3
	2	0	0	0.7	0	3.4	4.3	6	2.7
	3	1	0	3.3	0	3.8	8.7	18	22.3
	10	2.3	0	196	0	10.02	24.3	68	1493.7
	33	11.7	0	820.7	0	26.7	98.3	185.7	2024.3
80	1	0.3	0	2	0	1.1	3.3	13.3	17
	2	0.7	0	1.3	0	1.9	6.7	4.3	1
	3	0.6	1	8.3	0	4.7	12	23.3	80
	10	5.3	0	280.7	0	10.2	35.3	72.3	1637.7
	33	17	1	221.7	0	26.12	97.7	62.3	308.3
100	1	0.3	0	1	0	0.6	2	1	1
	2	1.3	0	7	0	2	7.3	16.7	22.3
	3	2	0	8	0	1.8	8	21	21.3
	10	5.3	0	43	0	4.6	20.3	29.3	81.7
	33	26.3	0.3	3131	0.9	23.5	112.7	219	3337.3

On constate que joindre des sommets semble être plus difficile que séparer des sommets. Les temps de calcul et le nombre de noeuds visités dans l'arbre de recherche sont loins d'être négligeables.

TABLE 4.3 – Graphes aléatoires complets Euclidiens-Joindre a et b

n	Nb cpl	sep	relax	temps	gap(%)	gap 2(%)	(4.5)	Nb coupes	Nb noeuds
20	1	0	0	0	0	2.3	2.7	0	1
	2	0	0	0	0	2.6	3.7	5	1.7
	3	0	0	0	0	3.7	4.3	6	7.3
	4	0	0	0	0	3.7	4.7	7.7	7.3
35	1	0	0	0.7	0	3.8	6.7	10	10
	2	0	0	2.3	0	4.6	7	39.7	97
	3	0.6	0	3	0	6.9	10.7	26.3	85.3
	4	0.3	0	7.7	0	7.7	8.3	51.3	242
45	1	0.3	0	2.3	0	2	8.3	14.3	19.3
	2	1	0	18.3	0	4.3	11	47.7	869.7
	3	1.3	0	40	0	5	14.3	142	746.7
	4	1.3	0	86.7	0	6	14.3	153	1778.3
50	1	0.7	0	10.3	0	2.5	7.7	78	181
	2	1.3	0	4.7	0	2.1	9.3	25.7	34
	3	1.7	0	247.3	0	5.1	9.3	307	5996.7
	4	1.3	0	81	0	4.6	11.3	164.3	1099.3
65	1	4	0	4	0	0.7	11.3	0	1
	2	4.7	0	72	0	2.9	15	136.3	559
	3	6	0	270.7	0	4.1	18.7	366.3	1910
	4	5	0	3406.7	1.2	5.1	19.3	594.3	43553.3
80	1	5.3	0	15.3	0	1	9.3	2.7	3.7
	2	15.3	0	330.3	0	3.1	23.7	160	1870.3
	3	13.3	0	689.3	0	2.8	19.3	238.3	4738.7
	4	11	0	2605	1.1	3.3	18.7	233.3	6025.3
100	1	13.3	0	18	0	0.7	11	3.7	2.3
	2	21.7	0	375.3	0	1.5	16.3	196	712
	3	56.7	0	1225.7	0	3.5	36.7	406	2035.7
	4	39	0	5516.3	1.28	4.2	31.7	611.3	14368

4.6 Conclusion et perspectives

Bien que les inégalités introduites dans ce chapitre permettent d'intégrer certaines contraintes techniques, il est nécessaire d'approfondir l'étude polyédrale.

On peut également essayer d'intégrer plusieurs combinaisons des contraintes décrites dans ce chapitre. Ceci est déjà possible dans le logiciel réalisé dans cette thèse. Cependant, plus de travail est nécessaire pour réduire les temps de calcul.

Rappelons enfin que la complexité du problème de synthèse d'un réseau à composantes connexes unicycliques avec contraintes de degré reste encore à étudier.

Spectre de quelques graphes unicycliques

Après avoir étudié plusieurs variantes du problème de synthèse de réseaux à composantes connexes unicycliques, nous allons considérer deux classes particulières de graphes unicycliques. Nous étudions le spectre des matrices d'adjacence et des Laplaciens associés à ces graphes. Outre l'intérêt théorique de ces résultats, la connaissance des valeurs propres extrémales pourrait renforcer les relaxations linéaires utilisées dans la synthèse de ce type de réseaux.

5.1 Introduction

L'étude du spectre des matrices associées à un graphe permet d'avoir des éclairages intéressants sur la structure du graphe. Elle peut même être à l'origine d'algorithmes combinatoires. Rappelons par exemple que la programmation semi-définie positive est équivalente à optimiser dans l'ensemble des matrices dont toutes les valeurs propres sont positives.

L'objectif dans ce chapitre est d'étudier les valeurs propres extrémales des matrices associées à certaines classes de graphes unicycliques. Rappelons que G est un graphe d'ordre n , et de taille m . On note par $A(G)$ ou simplement par A la matrice d'adjacence de G , et par L la matrice obtenue de la différence entre la matrice d'adjacence et la matrice diagonale des degrés D . L est connue sous le nom de la *matrice d'admittance*, ou en général la *matrice du Laplacien*. On note aussi par d_1, d_2, \dots, d_n les degrés des sommets de G , ordonnés dans un ordre décroissant $d_1 \geq d_2 \geq \dots \geq d_n$ en posant $d_1 = \Delta$ et $d_n = \delta$ qui sont les degrés maximum et minimum respectivement.

On note en général les valeurs propres associées à la matrice d'adjacence par $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$ qui sont alors ordonnées dans un ordre décroissant, et par $\mu_1 \geq \mu_2 \geq \dots \mu_{n-1} \geq \mu_n = 0$ les valeurs propres associées à la matrice L ordonnées dans le même ordre. Notez qu'on sait que G est connexe si et seulement si la multiplicité de μ_n vaut 1 (i.e., $\mu_{n-1} \neq \mu_n$).

Ce chapitre se découpe en trois sections principales. Nous rappellerons quelques résultats connus dans la section qui suit. Ensuite nous présenterons nos propres résultats.

5.2 Rappels sur les bornes des valeurs propres extrêmes des matrices A et L

5.2.1 Bornes de λ_1 et de λ_n

Rappelons tout d'abord que la fonction $f(x) = x^t Ax$, où A est une matrice symétrique réelle et x est un vecteur de norme 1, atteint son maximum (resp. minimum) au niveau du vecteur propre associé à la plus grande (petite) valeur propre.

Cette propriété est à l'origine de la plupart des résultats liés au spectre.

Proposition 5.1: [23] Soit G un graphe d'ordre n et de taille m . Notons $\bar{d} = \frac{\sum_{i=1}^n d_i}{n}$. Alors

$$\Delta \geq \lambda_1 \geq \bar{d} \quad (5.1)$$

$$\sum_{i=1}^n \lambda_i = 0 \quad (5.2)$$

$$\sum_{i=1}^n \lambda_i^2 = 2m \quad (5.3)$$

$$\lambda_1 \geq 2\cos\left(\frac{\pi}{n+1}\right) \quad (5.4)$$

En 2003, une nouvelle borne de λ_1 a été établie :

Proposition 5.2: [121] Soit λ_1 la plus grande valeur propre de la matrice d'adjacence du graphe G . Alors

$$\lambda_1 \geq \sqrt{\Delta} \quad (5.5)$$

Proposition 5.3: [73] G est un graphe unicyclique alors :

$$\lambda_1 \leq 2\sqrt{\Delta-1} \quad (5.6)$$

Proposition 5.4: [27] Soit G un graphe connexe d'ordre n et de taille m , la borne suivante est valide :

$$\lambda_1 \leq \sqrt{2m - (n-1)\delta + (\delta-1)m_{max}} \quad (5.7)$$

où $m_{max} = \max_{i=1,n} m_i$ avec $m_i = \frac{1}{d_i} \sum_{j \sim i} d_j$

Proposition 5.5: [23] Pour le spectre d'un graphe G , on constate ce qui suit :

1. $1 \leq \lambda_1 \leq n-1$

2. $-\lambda_1 \leq \lambda_n \leq -1$

Par la suite, plusieurs résultats ont été proposés afin d'améliorer ces bornes. Citons par exemple une borne dûe à [30].

Proposition 5.6: [30] Soit G un graphe simple et connexe, d'ordre n et de taille m . Alors

$$\lambda_n \geq -\sqrt{2m - (n-1)\delta + (\delta-1)\Delta}.$$

5.2.2 Bornes de μ_1 et de μ_{n-1}

Proposition 5.7: [40] Soit G un graphe connexe, d'ordre n et de taille m . On note par $e(G)$ le nombre d'arêtes qu'il faut supprimer pour déconnecter le graphe. Alors :

$$\mu_{n-1} \geq 2e(G)\left(1 - \cos\left(\frac{\pi}{n}\right)\right) \quad (5.8)$$

Un autre résultat donné par [40] donne une expression de la connectivité algébrique.

Proposition 5.8: [40] Si G est un graphe d'ordre n , alors :

$$\mu_{n-1} = 2n \min_{x \neq 0} \frac{\sum_{uv \in E} a_{uv}(x_u - x_v)^2}{\sum_{u \in V} \sum_{v \in V} (x_u - x_v)^2} \quad (5.9)$$

Puis, on trouve aussi ce résultat qui ordonne les deux valeurs propres extrêmes μ_1 et μ_{n-1} par rapport aux degrés maximum et minimum du graphe :

Proposition 5.9: [102]

$$\mu_{n-1} \leq \frac{n}{n-1} \delta \leq \frac{n}{n-1} \Delta \leq \mu_1 \leq 2\Delta.$$

Si G est un graphe unicyclique, on a alors

$$\Delta + 1 \leq \mu_1 \leq \Delta + 2\sqrt{\Delta - 1}$$

Proposition 5.10: [19] Soit G un graphe de diamètre $D \geq 4$.

$$\mu_{n-1} \leq 1 - \frac{2\sqrt{\Delta - 1}}{\Delta} \left(1 - \frac{2}{D}\right) + \frac{2}{D}$$

Plus de détails et de références pourront être trouvés dans [24], [25], [68], [69] et [102].

5.3 Spectre des graphes unicycliques $G(q, n_1, 0)$

Nous définissons la classe des graphes unicycliques $G(q, n_1, n_2)$ comme suit : ce sont des graphes connexes unicycliques où la taille du cycle est q ; chaque sommet du cycle est relié à n_1 sommets qui sont eux même reliés à n_2 sommets pendants (voir Figure (5.3)).

Dans cette section, nous allons considérer le cas $n_2 = 0$ (Figure 5.1).

Proposition 5.11: Soit G un graphe unicyclique du type $G(q, n_1 = \Delta - 2, 0)$, alors

$$\lambda_1 = 1 + \sqrt{\Delta - 1} \quad (5.10)$$

Preuve:

Considérons un vecteur propre x associé à λ_1 . Le vecteur x a une composante par sommet.

Les sommets du cycle sont notés c_1, \dots, c_q . Pour chaque sommet c_i du cycle, il y a $\Delta - 2$ feuilles reliées à c_i qu'on note f_i^j avec $1 \leq j \leq \Delta - 2$.

En écrivant que x est un vecteur propre, on obtient les deux types d'égalités qui suivent ($\forall j = 1, \dots, (\Delta - 2)$ et $i = 1, \dots, q$) :

$$\lambda_1 x_{f_i^j} = x_{c_i} \quad (5.11)$$

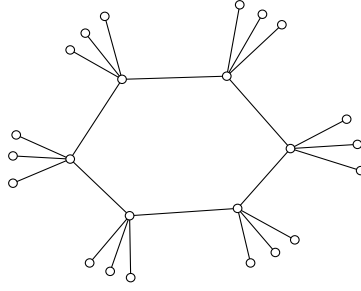


FIGURE 5.1 – Graphe unicyclique $G(6,3,0)$

$$\lambda_1 x_{c_i} = x_{c_{i+1}} + x_{c_{i-1}} + \sum_{j=1}^{\Delta-2} x_{f_i^j} \quad (5.12)$$

En combinant les deux types d'égalités, nous obtenons :

$$\left(\lambda_1 - \frac{\Delta-2}{\lambda_1}\right)x_{c_i} = x_{c_{i+1}} + x_{c_{i-1}} \quad (5.13)$$

En sommant les deux cotés de l'égalité précédente sur tous les sommets du cycle, on trouve

$$\left(\lambda_1 - \frac{\Delta-2}{\lambda_1} - 2\right) * \sum_i x_{c_i} = 0 \quad (5.14)$$

Par ailleurs, on sait que $\sum_i x_{c_i} \neq 0$ en vertu du théorème de Perron-Frobenius.

Théorème 5.1: [47] Soit A une matrice $n * n$ réelle et non négative d'un graphe G connexe. Alors sa plus grande valeur propre est simple. Le vecteur propre x associé à cette valeur propre a toutes ses composantes non nulles et de même signe.

De l'égalité (5.14) on déduit :

$$\lambda_1 - \frac{\Delta-2}{\lambda_1} = 2.$$

En multipliant par λ_1 , on obtient :

$$\lambda_1^2 - (\Delta - 2) = 2\lambda_1.$$

Le discriminant est donné par $Disc = 1 + \Delta - 2 = \Delta - 1 > 0$ ($\Delta \geq 3$)

d'où les deux racines suivantes :

$$\lambda_1^{(1)} = 1 - \sqrt{\Delta - 1}$$

$$\lambda_1^{(2)} = 1 + \sqrt{\Delta - 1}.$$

Sachant que le rayon spectral de la matrice d'adjacence A est tel que $\lambda_1 \geq 1$, et comme $\Delta \geq 3$, alors $\lambda_1^{(1)}$ sera rejetée, ce qui nous permet de conclure que

$$\lambda_1 = 1 + \sqrt{\Delta - 1}.$$

Proposition 5.12: Soit G un graphe unicyclique du type $G(q, n_1, 0)$. On a alors

$$\lambda_n = -1 - \sqrt{\Delta - 1}, \text{ si } q \text{ est pair.} \quad (5.15)$$

$$\lambda_n = -\cos\left(\frac{\pi}{q}\right) - \sqrt{\cos^2\left(\frac{\pi}{q}\right) + \Delta - 2}, \text{ si } q \text{ est impair.} \quad (5.16)$$

Preuve: De la même façon que dans la preuve de la proposition (5.11), on trouve ($\forall j = 1, \dots, (\Delta - 2)$ et $i = 1, \dots, q$) :

$$\lambda_n x_{f_i^j} = x_{c_i} \quad (5.17)$$

$$\lambda_n x_{c_i} = x_{c_{i+1}} + x_{c_{i-1}} + \sum_{j=1}^{\Delta-2} x_{f_i^j} \quad (5.18)$$

On obtient :

$$\left(\lambda_n - \frac{\Delta - 2}{\lambda_n}\right)x_{c_i} = x_{c_{i+1}} + x_{c_{i-1}}$$

Rappelons que $\lambda_n \leq -1$, donc $\lambda_n \neq 0$.

On obtient ainsi un système de q équations associé à une matrice circulante [54]

$$C = \begin{pmatrix} \gamma & -1 & -1 & 0 & \dots & 0 \\ -1 & \gamma & -1 & 0 & \dots & 0 \\ 0 & -1 & \gamma & -1 & \dots & 0 \\ \dots & 0 & -1 & \gamma & -1 & 0 \\ 0 & \dots & 0 & -1 & \gamma & -1 \\ -1 & 0 & \dots & 0 & -1 & \gamma \end{pmatrix}$$

avec $\gamma = \lambda_n - \frac{\Delta - 2}{\lambda_n}$, et qui admet $\gamma - 2\cos\left(\frac{2k\pi}{q}\right) \forall k = 1, \dots, q$ comme valeur propre. Il est clair que 0 doit être une valeur propre de C , qui équivaut à l'existence de k , tel que $\gamma - 2\cos\left(\frac{2k\pi}{q}\right) = 0$ (voir [23], [24], [54]).

Finalement, on résout l'équation

$$\lambda_n - \frac{\Delta - 2}{\lambda_n} = 2\cos\left(\frac{2k\pi}{q}\right)$$

qui est une équation de deuxième degré qui s'écrit sous la forme :

$$\lambda_n^2 - 2\cos\left(\frac{2k\pi}{q}\right)\lambda_n - (\Delta - 2) = 0$$

dont le discriminant est

$$Disc = \cos^2\left(\frac{2k\pi}{q}\right) + \Delta - 2 > 0, \quad \forall k = 0, \dots, q$$

la solution de cette équation est donnée par les deux racines suivantes, et cela $\forall k = 0, \dots, q-1$

$$\lambda_n^{(1)} = \cos\left(\frac{2k\pi}{q}\right) - \sqrt{\cos^2\left(\frac{2k\pi}{q}\right) + \Delta - 2}$$

$$\lambda_n^{(2)} = \cos\left(\frac{2k\pi}{q}\right) + \sqrt{\cos^2\left(\frac{2k\pi}{q}\right) + \Delta - 2}$$

Comme $\Delta \geq 3$ alors $\sqrt{\cos^2\left(\frac{2k\pi}{q}\right) + \Delta - 2} \geq 1$ et il est facile de déduire que

$$\sqrt{\cos^2\left(\frac{2k\pi}{q}\right) + \Delta - 2} + \cos\left(\frac{2k\pi}{q}\right) \geq 0$$

Ceci est en contradiction avec $\lambda_n < 0$ (voir [23]). Notre solution est donnée par :

$$\lambda_n^{(1)} = \cos\left(\frac{2k\pi}{q}\right) - \sqrt{\cos^2\left(\frac{2k\pi}{q}\right) + \Delta - 2}$$

Maintenant, on pose $f(k) = \cos\left(\frac{2k\pi}{q}\right) - \sqrt{\cos^2\left(\frac{2k\pi}{q}\right) + \Delta - 2}$. Le calcul de sa dérivée est donné par :

$$f'(k) = \sin\left(\frac{2k\pi}{q}\right) \left[\frac{\cos\left(\frac{2k\pi}{q}\right)}{\sqrt{\cos^2\left(\frac{2k\pi}{q}\right) + \Delta - 2}} - 1 \right]$$

le terme entre crochets est strictement négatif, car

$$\cos\left(\frac{2k\pi}{q}\right) < \sqrt{\cos^2\left(\frac{2k\pi}{q}\right) + \Delta - 2}$$

la dérivée est nulle pour les valeurs de $k = 0$, et $\frac{q}{2}$. D'après [23], on sait que λ_n est bornée. La fonction $f(k)$ est alors :

- décroissante : pour $k \in [0, \frac{q}{2}]$.
- croissante : pour $k \in [\frac{q}{2}, q]$

et atteint son minimum pour $k = \lfloor \frac{q}{2} \rfloor$. Ce qui nous donne :

- **q pair :**

$$\lambda_n^{(1)} = -1 - \sqrt{\Delta - 1}$$

- **q impair :**

$$\lambda_n^{(1)} = -\cos\left(\frac{\pi}{q}\right) - \sqrt{\cos^2\left(\frac{\pi}{q}\right) + \Delta - 2}$$

Étant données ces valeurs de λ_n , on sait que 0 est valeur propre de C . On dispose alors d'un vecteur x_C qui est le vecteur propre de C . Il est simple de reconstituer le reste du vecteur en utilisant les relations (5.17) et (5.18).

□.

Proposition 5.13: Soit G un graphe unicyclique du type $G(q, n_1, 0)$. La valeur exacte de la

connectivité algébrique est donnée par

$$\mu_{n-1} = \frac{1}{2} \left(\Delta - 1 + 4 \sin^2\left(\frac{\pi}{q}\right) - \sqrt{[\Delta - 1 + 4 \sin^2\left(\frac{\pi}{q}\right)]^2 - 16 \sin^2\left(\frac{\pi}{q}\right)} \right)$$

Preuve: À partir de la définition de la valeur propre du Laplacien, on a ($\forall j = 1, \dots, \Delta - 2$ et $\forall i = 1, \dots, q$) :

$$1. i \notin C_q : (1 - \mu_{n-1})x_{f_i^j} = x_{c_i}$$

$$2. i \in C_q : (\Delta - \mu_{n-1})x_{c_i} = x_{c_{i+1}} + x_{c_{i-1}} + \sum_{j=1}^{\Delta-2} x_{f_i^j}$$

en remplaçant la valeur de $x_{f_i^j}$ dans le deuxième cas, on obtient

$$\left(\Delta - \mu_{n-1} - \frac{\Delta - 2}{1 - \mu_{n-1}} \right) x_{c_i} = x_{c_{i+1}} + x_{c_{i-1}}$$

qui représente un système de q équations, et q inconnues. Pour le résoudre on pose $\gamma = \Delta - \mu_{n-1} - \frac{\Delta-2}{1-\mu_{n-1}}$, et on remarque que la matrice associée à ce système d'équations est aussi une matrice circulante :

$$C = \begin{pmatrix} \gamma & -1 & -1 & 0 & \dots & 0 \\ -1 & \gamma & -1 & 0 & \dots & 0 \\ 0 & -1 & \gamma & -1 & \dots & 0 \\ \dots & 0 & -1 & \gamma & -1 & 0 \\ 0 & \dots & 0 & -1 & \gamma & -1 \\ -1 & 0 & \dots & 0 & -1 & \gamma \end{pmatrix}$$

et qui admet $\gamma - 2 \cos\left(\frac{2k\pi}{q}\right) \forall k = 1, \dots, q$ comme valeur propre, tel que $\exists k, \gamma - 2 \cos\left(\frac{2k\pi}{q}\right) = 0$. D'après la Proposition 5.10, on constate que $\mu_{n-1} \neq 1$.

On résout alors

$$\Delta - \mu_{n-1} - \frac{\Delta - 2}{1 - \mu_{n-1}} = 2 \cos\left(\frac{2k\pi}{q}\right)$$

C'est une équation de deuxième degré, qu'on peut écrire sous la forme suivante

$$\Delta - \left(\frac{\mu_{n-1} - \mu_{n-1}^2 + \Delta - 2}{1 - \mu_{n-1}} \right) = 2 \cos\left(\frac{2k\pi}{q}\right)$$

$$\Rightarrow \Delta - \Delta \mu_{n-1} + \mu_{n-1}^2 - \mu_{n-1} - \Delta + 2 = 2 \cos\left(\frac{2k\pi}{q}\right) - 2 \cos\left(\frac{2k\pi}{q}\right) \mu_{n-1}$$

avec

$$\mu_{n-1} \neq 1$$

d'où

$$\mu_{n-1}^2 - (\Delta + 1 - 2 \cos\left(\frac{2k\pi}{q}\right)) \mu_{n-1} + 2(1 - \cos\left(\frac{2k\pi}{q}\right)) = 0$$

Le calcul du discriminant de l'équation trouvée, est donné par

$$Disc = (\Delta + 1 - 2 \cos\left(\frac{2k\pi}{q}\right))^2 - 8 + 8 \cos\left(\frac{2k\pi}{q}\right)$$

$$Disc = [\Delta + 1 - 2 \cos(\frac{2k\pi}{q})]^2 - 16 \sin^2(\frac{k\pi}{q})$$

on montre dans ce qui suit que le discriminant est positif :

$$Disc = [\Delta + 1 - 2 + 4 \sin^2(\frac{k\pi}{q})]^2 - 16 \sin^2(\frac{k\pi}{q})$$

$$Disc = (\Delta - 1)^2 + 16 \sin^4(\frac{k\pi}{q}) + 8(\Delta - 1) \sin^2(\frac{k\pi}{q}) - 16 \sin^2(\frac{k\pi}{q})$$

$$Disc = (\Delta - 1)^2 + 16 \sin^4(\frac{k\pi}{q}) + 8(\Delta - 3) \sin^2(\frac{k\pi}{q}) > 0$$

On obtient deux solutions :

$$\mu_{n-1}^{(1)} = \frac{1}{2} \left(\Delta + 1 - 2(1 - 2 \sin^2(\frac{k\pi}{q})) - \sqrt{[\Delta + 1 - 2(1 - 2 \sin^2(\frac{k\pi}{q}))]^2 - 16 \sin^2(\frac{k\pi}{q})} \right)$$

$$\mu_{n-1}^{(2)} = \frac{1}{2} \left(\Delta + 1 - 2(1 - 2 \sin^2(\frac{k\pi}{q})) + \sqrt{[\Delta + 1 - 2(1 - 2 \sin^2(\frac{k\pi}{q}))]^2 - 16 \sin^2(\frac{k\pi}{q})} \right)$$

après simplifications, on obtient :

$$\mu_{n-1}^{(1)} = \frac{1}{2} \left(\Delta - 1 + 4 \sin^2(\frac{k\pi}{q}) - \sqrt{[\Delta - 1 + 4 \sin^2(\frac{k\pi}{q})]^2 - 16 \sin^2(\frac{k\pi}{q})} \right)$$

$$\mu_{n-1}^{(2)} = \frac{1}{2} \left(\Delta - 1 + 4 \sin^2(\frac{k\pi}{q}) + \sqrt{[\Delta - 1 + 4 \sin^2(\frac{k\pi}{q})]^2 - 16 \sin^2(\frac{k\pi}{q})} \right)$$

Comme $\Delta \geq 3$, alors $\Delta - 1 + 4 \sin^2(\frac{k\pi}{q}) \geq 2$, ceci dit

$$\mu_{n-1}^{(2)} \geq 1 + \frac{\sqrt{[\Delta - 1 + 4 \sin^2(\frac{k\pi}{q})]^2 - 16 \sin^2(\frac{k\pi}{q})}}{2} > 1$$

sachant que $\mu_{n-1} < 1$, ceci nous aide à rejeter $\mu_{n-1}^{(2)}$ qui est supérieur à 1, et de ne prendre que

$$\mu_{n-1} = \mu_{n-1}^{(1)} = \frac{1}{2} \left(\Delta - 1 + 4 \sin^2(\frac{k\pi}{q}) - \sqrt{[\Delta - 1 + 4 \sin^2(\frac{k\pi}{q})]^2 - 16 \sin^2(\frac{k\pi}{q})} \right)$$

On fait l'étude de la fonction $\mu_{n-1}(k)$, avec $k = 0, \dots, q$, pour trouver la valeur de k qui minimise μ_{n-1} . On remarque alors que si $k = 0$, ou $k = q$, on trouve $\mu_{n-1} = 0$, qui est absurde. On s'intéresse qu'aux valeurs entières $k \in [1, q - 1]$.

Le calcul de la dérivée est donné par

$$\mu'_{n-1} = \frac{8\pi}{q} \sin(\frac{k\pi}{q}) \cos(\frac{k\pi}{q}) \left[1 - \frac{4 \sin^2(\frac{k\pi}{q}) + \Delta - 3}{\sqrt{(\Delta - 1)^2 + 16 \sin^4(\frac{k\pi}{q}) + 8(\Delta - 3) \sin^2(\frac{k\pi}{q})}} \right]$$

Le terme entre crochets est strictement positif, car

$$\sqrt{(\Delta-1)^2 + 16 \sin^4\left(\frac{k\pi}{q}\right) + 8(\Delta-3) \sin^2\left(\frac{k\pi}{q}\right)} > 4 \sin^2\left(\frac{k\pi}{q}\right) + \Delta - 3 \quad (\Delta \geq 3)$$

car en posant (avec $\Delta \geq 3$) :

$$\begin{aligned} & (\Delta-1)^2 > (\Delta-3)^2 \\ \Rightarrow & (\Delta-1)^2 + 16 \sin^4\left(\frac{k\pi}{q}\right) > (\Delta-3)^2 + 16 \sin^4\left(\frac{k\pi}{q}\right) \\ \Rightarrow & (\Delta-1)^2 + 16 \sin^4\left(\frac{k\pi}{q}\right) + 8(\Delta-3) \sin^2\left(\frac{k\pi}{q}\right) > (\Delta-3)^2 + 16 \sin^4\left(\frac{k\pi}{q}\right) + 8(\Delta-3) \sin^2\left(\frac{k\pi}{q}\right) \end{aligned}$$

on remarque alors que le passage aux racines carrées des deux cotés de l'inégalité nous donne

$$\sqrt{(\Delta-1)^2 + 16 \sin^4\left(\frac{k\pi}{q}\right) + 8(\Delta-3) \sin^2\left(\frac{k\pi}{q}\right)} > 4 \sin^2\left(\frac{k\pi}{q}\right) + \Delta - 3$$

La dérivée est nulle uniquement lorsque $\sin\left(\frac{k\pi}{q}\right) = 0$ ou $\cos\left(\frac{k\pi}{q}\right) = 0$, donc elle est nulle pour les valeurs suivantes $k = 0, q, \frac{q}{2}, \frac{3q}{2}$ et comme $k \in [1, q-1]$, elle est nulle uniquement pour $k = \frac{q}{2}$, avec $\mu_{n-1}(k = \frac{q}{2}) = \frac{\Delta+3-\sqrt{\Delta^2+6\Delta-7}}{2} < 1$.

La fonction μ , est bornée et se comporte ainsi :

- croissante : pour $k \in [1, \frac{q}{2}]$.
- décroissante : pour $k \in [\frac{q}{2}, q-1]$

On conclue alors que $\mu_{n-1}(k = q-1) = \mu_{n-1}(k = 1)$ et que la valeur à choisir est $k = 1$ (ou $k = q-1$), ce qui donne

$$\mu_{n-1} = \frac{1}{2} \left(\Delta - 1 + 4 \sin^2\left(\frac{\pi}{q}\right) - \sqrt{[\Delta - 1 + 4 \sin^2\left(\frac{\pi}{q}\right)]^2 - 16 \sin^2\left(\frac{\pi}{q}\right)} \right)$$

□

Proposition 5.14: Soit $G(q, n_1, 0)$ un graphe unicyclique. On a

1. Si q est pair : $\mu_1 = \frac{\Delta+3+\sqrt{(\Delta+3)^2-16}}{2}$
2. Si q est impair : $\mu_1 = \frac{(\Delta+1)+2\cos(\frac{\pi}{q})+\sqrt{[(\Delta+1)+2\cos(\frac{\pi}{q})]^2-8(1+\cos(\frac{\pi}{q}))}}{2}$

Preuve: En suivant le même raisonnement que celui de la preuve de la Proposition (5.13), et en utilisant la matrice circulante associée, on arrive à l'équation suivante :

$$\mu_1^2 - (\Delta + 1 - 2 \cos(\frac{2k\pi}{q}))\mu_1 + 2(1 - \cos(\frac{2k\pi}{q})) = 0$$

et on rappelle que pour un graphe unicyclique, on a bien

$$\Delta + 1 \leq \mu_1 \leq \Delta + 2\sqrt{\Delta - 1} \quad (5.19)$$

Le discriminant de cette équation est positif (voir preuve de la Proposition (5.13)), ce qui nous conduit à deux solutions :

$$1. \mu_1^{(1)} = \frac{\Delta+1-2\cos(\frac{2k\pi}{q})-\sqrt{[\Delta+1-2\cos(\frac{2k\pi}{q})]^2-8[1-\cos(\frac{2k\pi}{q})]}}{2}$$

$$2. \mu_1^{(2)} = \frac{\Delta+1-2\cos(\frac{2k\pi}{q})+\sqrt{[\Delta+1-2\cos(\frac{2k\pi}{q})]^2-8[1-\cos(\frac{2k\pi}{q})]}}{2}$$

Il est clair que $\mu_1^{(1)}$ ne vérifie pas la formule (5.19), car

$$\mu_1^{(1)} = \frac{\Delta+1}{2} - \left(\frac{2\cos(\frac{2k\pi}{q}) + \sqrt{[\Delta+1-2\cos(\frac{2k\pi}{q})]^2-8[1-\cos(\frac{2k\pi}{q})]}}{2} \right) < \Delta+1$$

ce qui nous permet de l'exclure et de ne garder que

$$\mu_1 = \frac{\Delta+1-2\cos(\frac{2k\pi}{q}) + \sqrt{[\Delta+1-2\cos(\frac{2k\pi}{q})]^2-8[1-\cos(\frac{2k\pi}{q})]}}{2}$$

μ_1 est une fonction en k , qui est symétrique autour de $\lfloor \frac{q}{2} \rfloor$, c'est à dire qu'on retrouve la même valeur de μ_1 pour des valeurs de $k = 0$ et $k = q$, ensuite $k = 1$ et $k = q - 1, \dots$

Avec ce raisonnement, on peut tracer l'évolution de la fonction μ_1 par rapport aux valeurs de $k = 0, \dots, q$ (voir Figure (5.2)). On rappelle aussi que $\cos(\frac{2(q-1)\pi}{2q}) = \cos(\pi - \frac{\pi}{q}) = -\cos(\frac{\pi}{q})$.

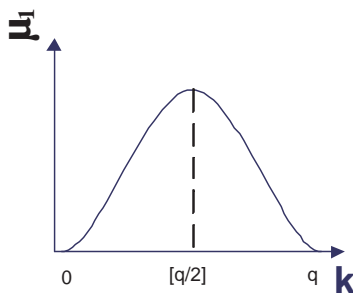


FIGURE 5.2 – La courbe de μ_1 en fonction de k

Le calcul de la dérivée de notre fonction μ_1 , nous donne

$$\mu_1' = \frac{1}{2} \left[\frac{4\pi}{q} \sin\left(\frac{2k\pi}{q}\right) + \frac{2(\Delta+1-2\cos(\frac{2k\pi}{q}))(\frac{4\pi}{q} \sin(\frac{2k\pi}{q})) - \frac{16\pi}{q} \sin(\frac{2k\pi}{q})}{2\sqrt{[\Delta+1-2\cos(\frac{2k\pi}{q})]^2-8[1-\cos(\frac{2k\pi}{q})]}} \right]$$

$$\mu_1' = \frac{2\pi}{q} \sin\left(\frac{2k\pi}{q}\right) \left[1 + \frac{(\Delta+1-2\cos(\frac{2k\pi}{q})) - 2}{\sqrt{[\Delta+1-2\cos(\frac{2k\pi}{q})]^2-8[1-\cos(\frac{2k\pi}{q})]}} \right]$$

et comme $\Delta \geq 3$, le terme entre crochets est strictement positif. La dérivée est égale à zéro pour les valeurs de $k = 0, \frac{q}{2}$, et comme l'étude se fait dans l'intervalle des entiers $k \in [1, q - 1]$, alors on garde l'unique valeur entière de $k = \lfloor \frac{q}{2} \rfloor$. Comme on l'a déjà vu précédemment, la fonction étudiée atteint son maximum lorsque $k = \lfloor \frac{q}{2} \rfloor$, et on distingue :

1. q pair : $k = \frac{q}{2}$, et

$$\mu_1 = \frac{\Delta + 3 + \sqrt{(\Delta + 3)^2 - 16}}{2}$$

2. q impair : $k = \frac{q-1}{2}$, et

$$\mu_1 = \frac{(\Delta + 1) + 2 \cos\left(\frac{\pi}{q}\right) + \sqrt{[(\Delta + 1) + 2 \cos\left(\frac{\pi}{q}\right)]^2 - 8(1 + \cos\left(\frac{\pi}{q}\right))}}{2}$$

□

5.4 Spectre des graphes unicycliques $G(q, n_1, n_2)$

Dans cette partie, on s'intéresse aux graphes unicycliques du type de la figure (5.3). On note par $G(q, n_1, n_2)$, le graphe unicyclique dont le cycle est C_q , puis sur chacun des sommets du cycle, on trouve n_1 sommets adjacents, et sur chacun de ces derniers, on trouve n_2 sommets pendants.

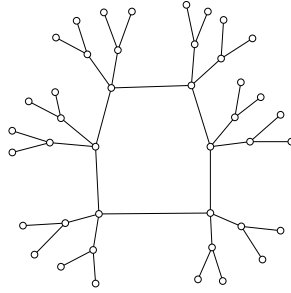


FIGURE 5.3 – Graphe unicyclique $G(6, 2, 2)$

On considère un vecteur propre x associé à λ_1 . Le vecteur x a une composante par sommet.

Les sommets du cycle sont notés c_1, \dots, c_q . Pour chaque sommet c_i du cycle, il y a n_1 sommets adjacents qu'on note $I_i^{j_1}$ avec $1 \leq j_1 \leq n_1$. Chaque sommet intermédiaire $I_i^{j_1}$ est adjacent à n_2 feuilles notées par $f_{i,j_1}^{j_2}$ avec $1 \leq j_2 \leq n_2$.

On a la proposition suivante :

Proposition 5.15: Soit G un graphe unicyclique du type $G(q, n_1, n_2)$. On a

$$\lambda_1(G) = \frac{2}{3} + 2\sqrt{\frac{-P}{3}} \cos\left(\frac{\arccos\left(\frac{-Q}{2}\sqrt{\frac{-27}{P^3}}\right)}{3}\right) \quad (5.20)$$

où

$$P = \frac{-4}{3} - (n_1 + n_2)$$

et

$$Q = \frac{-2}{27} (8 + 9(n_1 + n_2)) + 2n_2$$

Preuve: On note par N_1 l'ensemble des sommets adjacents aux sommets du cycle, et par N_2 l'ensemble des sommets pendants et adjacents aux sommets de N_1 .

Partant de la définition

$$\lambda_1 x_i = \sum_{j \sim i} x_j$$

avec la notation $j \sim i$ qui signifie que j est voisin de i . On distingue trois cas (avec $i = 1, \dots, q$, $j_1 = 1, \dots, n_1$ et $j_2 = 1, \dots, n_2$) :

$$1. i \in C_q : \lambda_1 x_{c_i} = x_{c_{i+1}} + x_{c_{i-1}} + \sum_{j_1=1}^{n_1} x_{I_i^{j_1}}$$

$$2. i \in N_1 : \lambda_1 x_{I_i^{j_1}} = x_{c_i} + \sum_{j_2=1}^{n_2} x_{f_{i,j_1}^{j_2}}$$

$$3. i \in N_2 : \lambda_1 x_{f_{i,j_1}^{j_2}} = x_{I_i^{j_1}}$$

Du troisième cas, on tire $x_{f_{i,j_1}^{j_2}} = \frac{x_{I_i^{j_1}}}{\lambda_1}$, et en le remplaçant dans le deuxième cas, on trouve

$$\begin{aligned} \lambda_1 x_{I_i^{j_1}} &= x_{c_i} + \frac{1}{\lambda_1} \sum_{j_2=1}^{n_2} x_{I_i^{j_1}} = x_{c_i} + \frac{n_2}{\lambda_1} x_{I_i^{j_1}} \\ \Rightarrow x_{I_i^{j_1}} &= \frac{\lambda_1}{\lambda_1^2 - n_2} x_{c_i} \end{aligned}$$

qui sera à son tour remplacé dans le premier cas. On obtient

$$\left(\lambda_1 - \frac{n_1 \lambda_1}{\lambda_1^2 - n_2}\right) x_{c_i} = x_{c_{i+1}} + x_{c_{i-1}} \quad (5.21)$$

D'après (5.2) ($\lambda_1 \geq \sqrt{\Delta}$), et du fait que $\Delta \geq n_2 + 1$, il est facile de voir que $\lambda_1^2 \neq n_2$.

À l'aide du théorème de Perron-Frobenius [47], l'équation (5.21), devient (du fait que $\sum_{c_i \in C_q} x_{c_i} \neq 0$) :

$$\lambda_1^3 - 2\lambda_1^2 - (n_1 + n_2)\lambda_1 + 2n_2 = 0 \quad (5.22)$$

Cette équation est du troisième degré, et le calcul du discriminant nous donne

$$disc = Q^2 + \frac{4}{27}P^3$$

où

$$Q = \frac{-2(8 + 9(n_1 + n_2))}{27} + 2n_2 = 2n_2 - \frac{16}{27} - \frac{2(n_1 + n_2)}{3}$$

et

$$P = -\frac{4}{3} - (n_1 + n_2)$$

alors

$$\begin{aligned} disc &= \left[\frac{4}{3}n_2 - \frac{2n_1}{3} - \frac{16}{27}\right]^2 - \frac{4}{27}\left[\frac{4}{3} + n_1 + n_2\right]^3 \\ \Rightarrow disc &= \frac{4}{9} \left(\left[2n_2 - n_1 - \frac{8}{9}\right]^2 - \frac{1}{3}\left[\frac{4}{3} + n_1 + n_2\right]^3 \right) \end{aligned}$$

Le signe du discriminant dépend uniquement du terme entre parenthèses. Si on pose $n_1 = kn_2$ avec $k \in \mathbb{R}^{*+}$, et après développement, on trouve que le signe de $disc$ dépendra du signe de

$$disc_0 = -n_2 \left((k+1)^3 n_2^2 + (k^2 - 4k - 8)n_2 + 16 \right)$$

On pose $f(n_2) = (k+1)^3 n_2^2 + (k^2 - 4k - 8)n_2 + 16$. On sait que $f(n_2) > n_2 \left((k+1)^3 n_2 + (k^2 - 4k - 8) \right)$. Il suffit donc de montrer que $(k+1)^3 n_2 + (k^2 - 4k - 8) \geq 0$. Sachant que $n_2 \geq 1$, et que $\forall k > 0$, $\frac{k^2 - 4k - 8}{(k+1)^3} < 1$ alors on déduit que $n_2 > \frac{k^2 - 4k - 8}{(k+1)^3}$. En d'autres termes, $(k+1)^3 n_2 + (k^2 - 4k - 8) \geq 0$ et on arrive au fait que $f(n_2) > 0$. On a alors pour tout $k \in \mathbb{R}^{*+}$: $disc_0 < 0$.

On en déduit que notre discriminant est négatif, et on a trois solutions réelles qui sont :

$$\begin{aligned} \lambda_1^{(1)} &= 2\sqrt{\frac{-P}{3}} \cos \left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right) + 2\pi}{3} \right) + \frac{2}{3} \\ \lambda_1^{(2)} &= 2\sqrt{\frac{-P}{3}} \cos \left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right) + 4\pi}{3} \right) + \frac{2}{3} \\ \lambda_1^{(3)} &= 2\sqrt{\frac{-P}{3}} \cos \left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3} \right) + \frac{2}{3} \end{aligned}$$

et qui deviennent après le développement du cosinus :

$$\begin{aligned} \lambda_1^{(1)} &= \frac{2}{3} - \sqrt{\frac{-P}{3}} \left[\cos \left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3} \right) + \sqrt{3} \sin \left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3} \right) \right] \\ \lambda_1^{(2)} &= \frac{2}{3} + \sqrt{\frac{-P}{3}} \left[-\cos \left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3} \right) + \sqrt{3} \sin \left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3} \right) \right] \\ \lambda_1^{(3)} &= \frac{2}{3} + 2\sqrt{\frac{-P}{3}} \cos \left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3} \right) \end{aligned}$$

On rappelle aussi que pour un angle β

$$\text{Arccos}(\beta) \in [0; \pi] \Rightarrow \frac{\text{Arccos}(\beta)}{3} \in [0; \frac{\pi}{3}]$$

Il est clair qu'en variant l'angle du cosinus entre 0 et $\frac{\pi}{3}$, et en tenant compte du résultat $\lambda_1 \geq \sqrt{\Delta}$ (voir [121]), on trouve que le terme entre crochets de $\lambda_1^{(1)}$ est positif, et dans ce cas $\lambda_1^{(1)} < \frac{2}{3}$, or elle devrait vérifier $\lambda_1^{(1)} \geq \sqrt{3}$, alors elle est rejetée.

D'après la Proposition 5.2, λ_1 doit toujours vérifier $\lambda_1 \geq \sqrt{\Delta}$.

Il est facile de voir que le terme entre crochets de $\lambda_1^{(2)}$ varie entre -1 et 1 . $\lambda_1^{(2)}$ atteint son maximum lorsque la valeur entre crochets est de 1 . Ainsi, $\lambda_1^{(2)} = \frac{2}{3} + \sqrt{\frac{4}{9} + \frac{n_1+n_2}{3}}$.

Il est facile de voir que $\Delta = n_2 + 1$ si $n_2 \geq n_1 + 2$, et $\Delta = n_1 + 2$ sinon.

1. $n_1 > n_2$, $\Delta = n_1 + 2$, et on pose $n_2 = k_1 n_1$ avec $k_1 \in \mathbb{R}^{*+}$ et $k_1 < 1$. On veut alors montrer que $\sqrt{\frac{4}{9} + \frac{k_1+1}{3}n_1} < \sqrt{n_1+2} - \frac{2}{3}$: après développement, on cherche finalement à montrer que $(n_1 + 2) - \frac{k_1+1}{3}n_1 - \frac{4}{3}\sqrt{n_1+2} \geq 0$. Le membre gauche de cette dernière inégalité peut s'écrire ainsi : $\left(\sqrt{2-k_1}\sqrt{n_1+2} - \frac{2}{\sqrt{2-k_1}}\right)^2 + 2(k_1+1) - \frac{4}{2-k_1}$. Comme on peut facilement vérifier que $2(k_1+1) - \frac{4}{2-k_1} > 0$, alors on constate que $\left(\sqrt{2-k_1}\sqrt{n_1+2} - \frac{2}{\sqrt{2-k_1}}\right)^2 + 2(k_1+1) - \frac{4}{2-k_1} \geq 0$.
2. $n_2 \geq n_1 + 2$, $\Delta = n_2 + 1$, et on pose $n_1 = k_1 n_2$ avec $k_1 \in \mathbb{R}^{*+}$ et $k_1 < 1$. On veut montrer que $\sqrt{\frac{4}{9} + \frac{k_1+1}{3}n_2} < \sqrt{n_2+1} - \frac{2}{3}$: après développement de cette inégalité, on cherche alors à montrer que $(2-k_1)(n_2+1) - 4\sqrt{n_2+1} + (k_1+1) \geq 0$. On fait un changement de variable sur le membre gauche de l'inégalité. On pose $z = \sqrt{n_2+1}$, avec $z \geq 2$. Ainsi, on trouve deux racines inférieures à 2 : $z_1 = \frac{2-\sqrt{k_1^2-k_1+2}}{2-k_1}$, et $z_2 = \frac{2+\sqrt{k_1^2-k_1+2}}{2-k_1}$. On conclue alors que $(2-k_1)(n_2+1) - 4\sqrt{n_2+1} + (k_1+1) \geq 0$

$\lambda_1^{(2)} < \sqrt{\Delta}$ pour les deux cas étudiés, donc elle est rejetée.

Finalement, notre solution est

$$\lambda_1^{(3)} = 2\sqrt{\frac{-P}{3}} \cos\left(\frac{\arccos\left(\frac{-Q}{2}\sqrt{\frac{-27}{P^3}}\right)}{3}\right) + \frac{2}{3}$$

□

Proposition 5.16: Soit G un graphe unicyclique du type $G(q, n_1, n_2)$. On a

1. **q pair :** $\lambda_n = -\frac{2}{3} - \sqrt{\frac{-P}{3}} \left[\cos\left(\frac{\arccos\left(\frac{-Q}{2}\sqrt{\frac{-27}{P^3}}\right)}{3}\right) + \sqrt{3} \sin\left(\frac{\arccos\left(\frac{-Q}{2}\sqrt{\frac{-27}{P^3}}\right)}{3}\right) \right]$
2. **q impair :** $\lambda_n = \frac{2}{3} \cos\left(\frac{2\pi}{q}\right) - \sqrt{\frac{-P}{3}} \left[\cos\left(\frac{\arccos\left(\frac{-Q}{2}\sqrt{\frac{-27}{P^3}}\right)}{3}\right) + \sqrt{3} \sin\left(\frac{\arccos\left(\frac{-Q}{2}\sqrt{\frac{-27}{P^3}}\right)}{3}\right) \right]$

où

$$P = \frac{-4\cos^2\left(\frac{2\pi k}{q}\right)}{3} - (n_1 + n_2)$$

et

$$Q = \frac{-2\cos\left(\frac{2\pi k}{q}\right)}{27} \left[8\cos^2\left(\frac{2\pi k}{q}\right) + 9(n_1 + n_2) \right] + 2n_2 \cos\left(\frac{2\pi k}{q}\right)$$

et que $k = \frac{q}{2}$, si q est pair, sinon $k = \frac{q-1}{2}$.

Preuve: Suivant le même raisonnement précédent, on obtient l'équation suivante

$$\left[\lambda_n - \frac{n_1 \lambda_n}{\lambda_n^2 - n_2} \right] x_{c_i} = x_{c_{i+1}} + x_{c_{i-1}}$$

qui est une équation parmi un système associé à une matrice circulante :

$$C = \begin{pmatrix} \gamma & -1 & -1 & 0 & \dots & 0 \\ -1 & \gamma & -1 & 0 & \dots & 0 \\ 0 & -1 & \gamma & -1 & \dots & 0 \\ \dots & 0 & -1 & \gamma & -1 & 0 \\ 0 & \dots & 0 & -1 & \gamma & -1 \\ -1 & 0 & \dots & 0 & -1 & \gamma \end{pmatrix}$$

et qui admet $\gamma - 2\cos(\frac{2k\pi}{q})$ comme valeur propre, tel que $\exists k, \gamma - 2\cos(\frac{2k\pi}{q}) = 0$.

D'après la Proposition (5.5), on a $\lambda_n \leq -1$, donc $\lambda_n^2 - n_2 > 1 - n_2 = \eta$ avec η qui prend ses valeurs dans \mathbb{Z}^- . On conclue que $\lambda_n^2 \neq n_2$.

La solution de ce système existe si et seulement si on a

$$\gamma = \lambda_n - \frac{n_1 \lambda_n}{\lambda_n^2 - n_2} = 2\cos(\frac{2k\pi}{q})$$

autrement dit

$$\lambda_n^3 - 2\cos(\frac{2k\pi}{q})\lambda_n^2 - (n_1 + n_2)\lambda_n + 2n_2\cos(\frac{2k\pi}{q}) = 0$$

C'est une équation du troisième degré qu'on peut résoudre avec la méthode de Cardan, en posant :

$$P = \frac{-4\cos^2(\frac{2\pi k}{q})}{3} - (n_1 + n_2)$$

et

$$Q = \frac{-2\cos(\frac{2\pi k}{q})}{27} \left[8\cos^2(\frac{2\pi k}{q}) + 9(n_1 + n_2) \right] + 2n_2\cos(\frac{2\pi k}{q})$$

Le calcul du discriminant est donné par

$$Disc = Q^2 + \frac{4}{27}P^3 = \cos^2(\frac{2\pi k}{q}) \left[2n_2 - \frac{2}{27}(8\cos^2(\frac{2\pi k}{q}) + 9n_1 + 9n_2) \right]^2 - \frac{4}{27} \left[\frac{4}{3}\cos^2(\frac{2\pi k}{q}) + n_1 + n_2 \right]^3$$

$$\Rightarrow Disc = \cos^2(\frac{2\pi k}{q}) \left[\frac{4}{3}n_2 - \frac{2}{3}n_1 - \frac{16}{27}\cos^2(\frac{2\pi k}{q}) \right]^2 - \frac{4}{27} \left[\frac{4}{3}\cos^2(\frac{2\pi k}{q}) + (n_1 + n_2) \right]^3$$

$$\Rightarrow Disc = \frac{4}{9}\cos^2(\frac{2\pi k}{q}) \left[2n_2 - n_1 - \frac{8}{9}\cos^2(\frac{2\pi k}{q}) \right]^2 - \frac{4}{27} \left[\frac{4}{3}\cos^2(\frac{2\pi k}{q}) + (n_1 + n_2) \right]^3$$

$$\Rightarrow Disc = \frac{4}{9}\cos^2(\frac{2\pi k}{q}) \left[\frac{8}{9}\cos^2(\frac{2\pi k}{q}) + (n_1 + n_2) - 3n_2 \right]^2 - \frac{4}{27} \left[\frac{4}{3}\cos^2(\frac{2\pi k}{q}) + (n_1 + n_2) \right]^3$$

On pose $n_1 = k_0 n_2$ avec $k_0 \in \mathbb{R}^{*+}$. $Disc$ aura le même signe que

$$Disc_0 = \frac{-n_2}{3} \left((k_0 + 1)^3 n_2^2 + \cos^2(\frac{2\pi k}{q})(k_0^2 - 4k_0 - 8)n_2 + 16\cos^4(\frac{2\pi k}{q}) \right)$$

Pour montrer que ce discriminant est négatif, on suivra les mêmes étapes de la preuve précédente.

On pose $H(n_2) = (k_0 + 1)^3 n_2^2 + \cos^2(\frac{2\pi k}{q})(k_0^2 - 4k_0 - 8)n_2 + 16 \cos^4(\frac{2\pi k}{q})$. On sait que $H(n_2) > n_2 \left((k_0 + 1)^3 n_2 + \cos^2(\frac{2\pi k}{q})(k_0^2 - 4k_0 - 8) \right)$. Il suffit donc de montrer que $(k_0 + 1)^3 n_2 + \cos^2(\frac{2\pi k}{q})(k_0^2 - 4k_0 - 8) \geq 0$. Sachant que $n_2 \geq 1$, et que $\forall k_0 > 0$, $\frac{k_0^2 - 4k_0 - 8}{(k_0 + 1)^3} \cos^2(\frac{2\pi k}{q}) < 1$ alors on déduit que $n_2 > \frac{k_0^2 - 4k_0 - 8}{(k_0 + 1)^3} \cos^2(\frac{2\pi k}{q})$. En d'autres termes, $(k_0 + 1)^3 n_2 + \cos^2(\frac{2\pi k}{q})(k_0^2 - 4k_0 - 8) \geq 0$ et on arrive au fait que $H(n_2) > 0$. On a alors pour tout $k_0 \in \mathbb{R}^{*+}$: $Disc_0 < 0$. Le discriminant $Disc$ est alors négatif, et on possède trois solutions réelles :

$$\begin{aligned}\lambda_n^{(1)} &= \frac{2}{3} \cos\left(\frac{2\pi k}{q}\right) + 2\sqrt{\frac{-P}{3}} \cos\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right) + 2\pi}{3}\right) \\ \lambda_n^{(2)} &= \frac{2}{3} \cos\left(\frac{2\pi k}{q}\right) + 2\sqrt{\frac{-P}{3}} \cos\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right) + 4\pi}{3}\right) \\ \lambda_n^{(3)} &= \frac{2}{3} \cos\left(\frac{2\pi k}{q}\right) + 2\sqrt{\frac{-P}{3}} \cos\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right)\end{aligned}$$

Une fois qu'on développe le cosinus dans les trois précédentes formules, on obtient de nouveau :

$$\begin{aligned}\lambda_n^{(1)} &= \frac{2}{3} \cos\left(\frac{2\pi k}{q}\right) - \sqrt{\frac{-P}{3}} \left(\cos\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) + \sqrt{3} \sin\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) \right) \\ \lambda_n^{(2)} &= \frac{2}{3} \cos\left(\frac{2\pi k}{q}\right) + \sqrt{\frac{-P}{3}} \left(-\cos\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) + \sqrt{3} \sin\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) \right) \\ \lambda_n^{(3)} &= \frac{2}{3} \cos\left(\frac{2\pi k}{q}\right) + 2\sqrt{\frac{-P}{3}} \left(\cos\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) \right)\end{aligned}$$

Sachant que l'Arccosinus d'un angle est toujours dans l'intervalle $[0, \pi]$, on constate alors que l'angle du cosinus (qui est le même avec celui du sinus) est dans $[0, \frac{\pi}{3}]$, ce qui rend le sinus et le cosinus toujours positifs ($\forall n_1, n_2, k$).

Dans ce cas, où la valeur propre qu'on cherche est liée à la matrice circulante C , il suffit de trouver la plus petite valeur parmi les trois. En effet :

1. $\lambda_n^{(1)} - \lambda_n^{(2)} = -\sqrt{\frac{-P}{3}} \left(2\sqrt{3} \sin\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) \right) < 0$ et donc : $\lambda_n^{(1)} < \lambda_n^{(2)}$
2. $\lambda_n^{(1)} - \lambda_n^{(3)} = -\sqrt{\frac{-P}{3}} \left(\cos\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) + \sqrt{3} \sin\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) \right) < 0$ et donc

$$\lambda_n^{(1)} < \lambda_n^{(3)}$$

On conclue que la valeur propre recherchée est

$$\lambda_n^{(1)} = \frac{2}{3} \cos\left(\frac{2\pi k}{q}\right) - \sqrt{\frac{-P}{3}} \left(\cos\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) + \sqrt{3} \sin\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) \right)$$

Le calcul de la dérivée de λ_n est donné par :

$$\lambda_n' = -\sqrt{\frac{4}{9} \cos^2\left(\frac{2\pi k}{q}\right) + n_2 + n_1} \left[\frac{\sqrt{3}(-B - \frac{48\pi(\frac{1}{27} \cos(\frac{2\pi k}{q})(8 \cos^2(\frac{2\pi k}{q}) + 9(n_2 + n_1)) - n_2 \cos(\frac{2\pi k}{q}))^2}{q(\frac{4}{3} \cos^2(\frac{2\pi k}{q}) + n_2 + n_1)^4} \cos(\frac{2\pi k}{q}) \sin(\frac{2\pi k}{q}))}{2\sqrt{A}} \right. \\ \left. + \frac{-\frac{2\pi}{27q}(8 \cos^2(\frac{2\pi k}{q}) + 9(n_2 + n_1)) \sin(\frac{2\pi k}{q}) - \frac{32\pi}{27q} \cos^2(\frac{2\pi k}{q}) \sin(\frac{2\pi k}{q}) + \frac{2\pi n_2}{q} \sin(\frac{2\pi k}{q})}{\left(\frac{4}{3} \cos^2(\frac{2\pi k}{q}) + n_2 + n_1\right)^{\frac{3}{2}}} \sqrt{3} + \right. \\ \left. \frac{8\pi(\frac{1}{27} \cos(\frac{2\pi k}{q})(8 \cos^2(\frac{2\pi k}{q}) + 9(n_2 + n_1)) - n_2 \cos(\frac{2\pi k}{q}))}{q\left(\frac{4}{3} \cos^2(\frac{2\pi k}{q}) + n_2 + n_1\right)^{\frac{5}{2}}} \sqrt{3} \cos(\frac{2\pi k}{q}) \sin(\frac{2\pi k}{q}) \right] + \\ \frac{8\pi \cos\left(\frac{2\pi k}{q}\right) \left(\sqrt{3} \sqrt{1 - \frac{3(\frac{1}{27} \cos(\frac{2\pi k}{q})(8 \cos^2(\frac{2\pi k}{q}) + 9(n_2 + n_1)) - n_2 \cos(\frac{2\pi k}{q}))^2}{\left(\frac{4}{3} \cos^2(\frac{2\pi k}{q}) + n_2 + n_1\right)^3}} + \frac{\frac{1}{27} \cos(\frac{2\pi k}{q})(8 \cos^2(\frac{2\pi k}{q}) + 9(n_2 + n_1)) - n_2 \cos(\frac{2\pi k}{q})}{\left(\frac{4}{3} \cos^2(\frac{2\pi k}{q}) + n_2 + n_1\right)^{\frac{3}{2}}} \sqrt{3}} \right) \sin\left(\frac{2\pi k}{q}\right)}{9q \sqrt{\frac{4}{9} \cos^2\left(\frac{2\pi k}{q}\right) + n_2 + n_1}}$$

$$\frac{4\pi}{3q} \sin\left(\frac{2\pi k}{q}\right)$$

$$\text{avec } A = 1 - \frac{3\left(\frac{1}{27} \cos\left(\frac{2\pi k}{q}\right)(8 \cos^2\left(\frac{2\pi k}{q}\right) + 9(n_2 + n_1)) - n_2 \cos\left(\frac{2\pi k}{q}\right)\right)^2}{\left(\frac{4}{3} \cos^2\left(\frac{2\pi k}{q}\right) + n_2 + n_1\right)^3}$$

$$B = \frac{6\left(\frac{1}{27} \cos\left(\frac{2\pi k}{q}\right)(8 \cos^2\left(\frac{2\pi k}{q}\right) + 9(n_2 + n_1)) - n_2 \cos\left(\frac{2\pi k}{q}\right)\right) B_1}{\left(\frac{4}{3} \cos^2\left(\frac{2\pi k}{q}\right) + n_2 + n_1\right)^3}$$

$$\text{avec } B_1 = -\frac{2\pi}{27q}(8 \cos^2\left(\frac{2\pi k}{q}\right) + 9(n_2 + n_1)) \sin\left(\frac{2\pi k}{q}\right) - \frac{32\pi}{27q} \cos^2\left(\frac{2\pi k}{q}\right) \sin\left(\frac{2\pi k}{q}\right) + \frac{2\pi n_2}{q} \sin\left(\frac{2\pi k}{q}\right)$$

λ_n atteint sa valeur minimum lorsque : $\frac{2\pi k}{q} \in]\frac{\pi}{2}, \frac{3\pi}{2}[\Rightarrow \frac{q}{4} < k < \frac{3q}{4}$ et $k = 1, \dots, q-1$ et la valeur entière comprise entre $\frac{q}{4}$ et $\frac{3q}{4}$ est bien $k = \lfloor \frac{q}{2} \rfloor$.

On en distingue alors deux cas :

1. **q pair** : $k = \frac{q}{2}$, et $\lambda_n = -\frac{2}{3} - \sqrt{\frac{-P}{3}} \left(\cos\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) + \sqrt{3} \sin\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) \right)$

2. **q impair** : $k = \frac{q-1}{2}$, et $\lambda_n = \frac{2}{3} \cos\left(\frac{2\pi}{q}\right) - \sqrt{\frac{-P}{3}} \left(\cos\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) + \sqrt{3} \sin\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) \right)$

□

Remarque : Ces calculs ont été réalisés avec [72].

Proposition 5.17: Soit G un graphe unicyclique du type $G(q, n_1, n_2)$. On a

1. **q pair** : $k = \frac{q}{2}$, et $\mu_1 = \frac{6+n_1+n_2}{3} + 2\sqrt{\frac{-P}{3}} \cos\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right)$

2. **q impair** : $k = \frac{q-1}{2}$, et $\mu_1 = \frac{4+n_1+n_2-2\cos\left(\frac{2\pi}{q}\right)}{3} + 2\sqrt{\frac{-P}{3}} \cos\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right)$

où

$$P = -\frac{\left(4 + n_1 + n_2 - 2\cos\left(\frac{2\pi k}{q}\right)\right)^2}{3} + n_1 + 2n_2 + n_1n_2 + 5 - 2(2 + n_2)\cos\left(\frac{2\pi k}{q}\right)$$

$$\text{et } Q = -\frac{4+n_1+n_2-2\cos\left(\frac{2\pi k}{q}\right)}{27} \left[2\left(4 + n_1 + n_2 - 2\cos\left(\frac{2\pi k}{q}\right)\right)^2 - 9(n_1 + 2n_2 + n_1n_2 + 5 - 2(2 + n_2)\cos\left(\frac{2\pi k}{q}\right)) \right] + 2\cos\left(\frac{2\pi k}{q}\right) - 2$$

Preuve: Partant de la définition du rayon spectral de la matrice du Laplacien de G , on a (avec $j_1 = 1 \dots, n_1$ et $i = 1, \dots, q$ et $j_2 = 1 \dots, n_2$) :

$$- i \in C_q : (n_1 + 2 - \mu_1)x_{c_i} = x_{c_{i+1}} + x_{c_{i-1}} + \sum_{j_1=1}^{n_1} x_{I_i^{j_1}}$$

$$- i \in N_1 : (n_2 + 1 - \mu_1)x_{I_i^{j_1}} = x_{c_i} + \sum_{j_2=1}^{n_2} x_{f_{i,j_1}^{j_2}}$$

$$- i \in N_2 : (1 - \mu_1)x_{f_{i,j_1}^{j_2}} = x_{I_i^{j_1}}$$

Ces équations forment un système associé à une matrice circulante :

$$C = \begin{pmatrix} \gamma & -1 & -1 & 0 & \dots & 0 \\ -1 & \gamma & -1 & 0 & \dots & 0 \\ 0 & -1 & \gamma & -1 & \dots & 0 \\ \dots & 0 & -1 & \gamma & -1 & 0 \\ 0 & \dots & 0 & -1 & \gamma & -1 \\ -1 & 0 & \dots & 0 & -1 & \gamma \end{pmatrix}$$

et qui admet $\gamma - 2\cos\left(\frac{2k\pi}{q}\right)$ comme valeur propre, tel que $\exists k, \gamma - 2\cos\left(\frac{2k\pi}{q}\right) = 0$.

La solution de ce système existe si et seulement si on a

$$\gamma = \left(n_1 + 2 - \mu_1 - \underbrace{\frac{n_1}{n_2 + 1 - \mu_1 - \frac{n_2}{1 - \mu_1}}}_{\text{}} \right) = 2\cos\left(\frac{2k\pi}{q}\right) \quad (\text{et } \mu_1 > 1)$$

Le terme indiqué sur l'accolade est différent de zéro, à cause du fait que $\mu_1 \geq \Delta + 1$. En effet, en calculant son discriminant on trouve :

$disc = n_2(n_2 - 4)$. On discute trois cas :

1. $n_2 < 4$: il n'y a pas de solution.
2. $n_2 = 4$: on trouve une solution double $\mu_1 = 1 - \frac{n_2}{2}$ qui est rejetée ($\mu_1 \geq \Delta + 1$).
3. $n_2 > 4$: on trouve deux solutions de μ_1 : $1 + \frac{n_2 + \sqrt{n_2(n_2 - 4)}}{2}$, et $1 + \frac{n_2 - \sqrt{n_2(n_2 - 4)}}{2}$ qui sont les deux rejetées, car $\mu_1 \geq \Delta + 1$.

On a alors une équation de troisième degré à résoudre

$$\mu_1^3 - \left(4 + n_1 + n_2 - 2\cos\left(\frac{2k\pi}{q}\right)\right)\mu_1^2 + \left(n_1 + 2n_2 + n_1n_2 + 5 - 2(2 + n_2)\cos\left(\frac{2k\pi}{q}\right)\right)\mu_1 + 2\cos\left(\frac{2k\pi}{q}\right) - 2 = 0.$$

Le calcul du discriminant de cette équation est :

$$disc = \left[2 \cos\left(\frac{2k\pi}{q}\right) - 2 - \frac{(4+n_1+n_2-2\cos(\frac{2k\pi}{q}))}{27} \left[2(4+n_1+n_2-2\cos(\frac{2k\pi}{q}))^2 - 9(n_1+2n_2+n_1n_2+5-2(2+n_2)\cos(\frac{2k\pi}{q})) \right] \right]^2 + \frac{4}{27} \left[-\frac{(4+n_1+n_2-2\cos(\frac{2k\pi}{q}))^2}{3} + (n_1+2n_2+n_1n_2+5-2(2+n_2)\cos(\frac{2k\pi}{q})) \right]^3$$

donc

$$disc = \left[2 \cos\left(\frac{2k\pi}{q}\right) - 2 - \frac{(4+n_1+n_2-2\cos(\frac{2k\pi}{q}))}{27} \left(2(4+n_1+n_2-2\cos(\frac{2k\pi}{q}))^2 - 9(n_1+2n_2+n_1n_2+5-2(2+n_2)\cos(\frac{2k\pi}{q})) \right) \right]^2 - \frac{4}{27} \left[\frac{(4+n_1+n_2-2\cos(\frac{2k\pi}{q}))^2}{3} - (n_1+2n_2+n_1n_2+5-2(2+n_2)\cos(\frac{2k\pi}{q})) \right]^3$$

On pose $n_1 = k_0 n_2$, avec $k_0 \in \mathbb{R}^{+*}$. Soient

$$\Gamma_1 = k_0 n_2^2 + (k_0 + 2)n_2 - 2(2 + n_2) \cos\left(\frac{2k\pi}{q}\right) + 5$$

et

$$\Gamma_2 = (k_0 + 1)n_2 + 4 - 2 \cos\left(\frac{2k\pi}{q}\right)$$

Le discriminant s'écrit de nouveau

$$disc = \left(2 \cos\left(\frac{2k\pi}{q}\right) - 2 - \frac{\Gamma_2}{27} (2\Gamma_2^2 - 9\Gamma_1) \right)^2 - 4 \left(\frac{\Gamma_2^2}{9} - \frac{\Gamma_1}{3} \right)^3$$

On pose $\Gamma_2 = k_1 \Gamma_1$, avec $k_1 \in \mathbb{R}^*$. On aura

$$disc = \left(2 \cos\left(\frac{2k\pi}{q}\right) - 2 - \frac{k_1 \Gamma_1}{27} (2k_1^2 \Gamma_1^2 - 9\Gamma_1) \right)^2 - 4 \left(\frac{k_1^2 \Gamma_1^2}{9} - \frac{3\Gamma_1}{3} \right)^3$$

$$disc = \left(2 \cos\left(\frac{2k\pi}{q}\right) - 2 + \frac{k_1 \Gamma_1^2}{3} - \frac{2k_1^3 \Gamma_1^3}{27} \right)^2 - 4 \left(\frac{k_1^2 \Gamma_1^2}{9} - \frac{3\Gamma_1}{3} \right)^3$$

Comme $\Gamma_1 \geq 0$, on peut alors écrire

$$disc \leq \left(\frac{k_1 \Gamma_1^2}{3} \right)^2 - 4 \left(\frac{k_1^2 \Gamma_1^2}{9} - \frac{3\Gamma_1}{3} \right)^3$$

Après développement du membre droit de cette inégalité, on trouve

$$disc \leq \left(\frac{k_1^2 \Gamma_1^2}{9} \right) \Gamma_1 \left(\Gamma_1 - \frac{4k_1}{3} \left(\frac{k_1 \Gamma_1}{3} - \frac{1}{k_1} \right)^3 \right)$$

Il est clair que le signe du discriminant dépend du signe du terme :

$$\Gamma_1 - \frac{4k_1}{3} \left(\frac{k_1 \Gamma_1}{3} - \frac{1}{k_1} \right)^3 = H(\Gamma_1)$$

On développe $H(\Gamma_1)$ et on obtient

$$H(\Gamma_1) = \frac{-4}{81k_1^2} \left(k_1^6 \Gamma_1^3 - 9k_1^4 \Gamma_1^2 - 27 \right) - \frac{\Gamma_1}{3}$$

On peut facilement montrer avec un changement de variable, que le terme au dessus de l'accolade est bien positif. Ceci signifie finalement, que le signe du discriminant de l'équation de troisième degré, est bien négatif.

On distingue trois solutions :

$$\begin{aligned} \mu_1^{(1)} &= \frac{4 + n_1 + n_2 - 2 \cos\left(\frac{2k\pi}{q}\right)}{3} + 2\sqrt{\frac{-P}{3}} \cos\left[\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right) + 2\pi}{3}\right] \\ \mu_1^{(2)} &= \frac{4 + n_1 + n_2 - 2 \cos\left(\frac{2k\pi}{q}\right)}{3} + 2\sqrt{\frac{-P}{3}} \cos\left[\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right) + 4\pi}{3}\right] \\ \mu_1^{(3)} &= \frac{4 + n_1 + n_2 - 2 \cos\left(\frac{2k\pi}{q}\right)}{3} + 2\sqrt{\frac{-P}{3}} \cos\left[\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right] \end{aligned}$$

et après développement des cosinus, on retrouve les trois formules suivantes :

$$\begin{aligned} \mu_1^{(1)} &= \frac{4 + n_1 + n_2 - 2 \cos\left(\frac{2k\pi}{q}\right)}{3} - \sqrt{\frac{-P}{3}} \left[\cos\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) - \sqrt{3} \sin\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) \right] \\ \mu_1^{(2)} &= \frac{4 + n_1 + n_2 - 2 \cos\left(\frac{2k\pi}{q}\right)}{3} - \sqrt{\frac{-P}{3}} \left[\cos\left[\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right] + \sqrt{3} \sin\left[\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right] \right] \\ \mu_1^{(3)} &= \frac{4 + n_1 + n_2 - 2 \cos\left(\frac{2k\pi}{q}\right)}{3} + 2\sqrt{\frac{-P}{3}} \cos\left(\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right) \end{aligned}$$

Pour trouver la valeur du rayon spectral parmi ces trois valeurs, on procède comme suit :

$$\mu_1^{(3)} - \mu_1^{(1)} = \sqrt{\frac{-P}{3}} \left[3 \cos\left[\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right] - \sqrt{3} \sin\left[\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right] \right] \geq 0$$

$$\mu_1^{(3)} - \mu_1^{(2)} = \sqrt{\frac{-P}{3}} \left[3 \cos\left[\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right] + \sqrt{3} \sin\left[\frac{\arccos\left(\frac{-Q}{2} \sqrt{\frac{-27}{P^3}}\right)}{3}\right] \right] \geq 0$$

Ceci nous aide à conclure que $\mu_1^{(3)}$ est la plus grande valeur propre associée au graphe $G(q, n_1, n_2)$.

Le calcul de la dérivée de $\mu_1^{(3)}$ est donné par

$$\begin{aligned} (\mu_1^{(3)})' &= \frac{2\sqrt{\frac{1}{9}\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)^2-\frac{1}{3}\left(-2(n_2+2)\cos\left(\frac{2\pi k}{q}\right)+n_1n_2+2n_2+n_1+5\right)A_1*A_2}}{3\sqrt{1-\frac{27\left(\frac{1}{54}\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)\left(2\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)^2-9\left(-2(n_2+2)\cos\left(\frac{2\pi k}{q}\right)+2n_2+n_1+5\right)\right)-\cos\left(\frac{2\pi k}{q}\right)+1\right)^2}{\left(2(n_2+2)\cos\left(\frac{2\pi k}{q}\right)+\frac{1}{9}\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)^2-n_1n_2-2n_2-n_1-5\right)^3}}}} + \\ & B_1 \cos\left(\frac{1}{3}\arccos\left(\frac{3\left(\frac{1}{54}\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)\left(2\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)^2-9\left(-2(n_2+2)\cos\left(\frac{2\pi k}{q}\right)+2n_2+n_1+5\right)\right)-\cos\left(\frac{2\pi k}{q}\right)+1\right)}{\left(2(n_2+2)\cos\left(\frac{2\pi k}{q}\right)+\frac{1}{9}\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)^2-n_1n_2-2n_2-n_1-5\right)^{\frac{3}{2}}}\sqrt{3}\right)\right) + \\ & \frac{\frac{4\pi}{3q}\sin\left(\frac{2\pi k}{q}\right)}{\sqrt{\frac{1}{9}\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)^2-\frac{1}{3}\left(-2(n_2+2)\cos\left(\frac{2\pi k}{q}\right)+n_1n_2+2n_2+n_1+5\right)}} \end{aligned}$$

avec :

$$\begin{aligned} A_1 &= \frac{3*A_1^{(1)}}{\left(2(n_2+2)\cos\left(\frac{2\pi k}{q}\right)+\frac{1}{9}\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)^2-n_1n_2-2n_2-n_1-5\right)^{\frac{3}{2}}\sqrt{3}-} \\ & \frac{9\left(\frac{1}{54}\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)\left(2\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)^2-9\left(-2(n_2+2)\cos\left(\frac{2\pi k}{q}\right)+2n_2+n_1+5\right)\right)-\cos\left(\frac{2\pi k}{q}\right)+1\right)A_1^{(2)}}{2\left(2(n_2+2)\cos\left(\frac{2\pi k}{q}\right)+\frac{1}{9}\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)^2-n_1n_2-2n_2-n_1-5\right)^{\frac{5}{2}}}\sqrt{3} \end{aligned}$$

$$\begin{aligned} A_1^{(1)} &= \frac{1}{54}\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right) + n_2 + n_1 + \\ & 4\left(\frac{16\pi}{q}\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)\sin\left(\frac{2\pi k}{q}\right)-\frac{36\pi}{q}(n_2+2)\sin\left(\frac{2\pi k}{q}\right)\right) + \\ & \frac{2\pi}{27q}\left(2\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)^2-9\left(-2(n_2+2)\cos\left(\frac{2\pi k}{q}\right)+2n_2+n_1+5\right)\right)\sin\left(\frac{2\pi k}{q}\right) + \\ & \frac{2\pi}{q}\sin\left(\frac{2\pi k}{q}\right) \end{aligned}$$

$$A_1^{(2)} = \left(\frac{8\pi}{9q}\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)\sin\left(\frac{2\pi k}{q}\right)-\frac{4\pi}{q}(n_2+2)\sin\left(\frac{2\pi k}{q}\right)\right)$$

$$A_2 = \sin\left(\frac{1}{3}\arccos\left(\frac{3*A_2^{(3)}}{\left(2(n_2+2)\cos\left(\frac{2\pi k}{q}\right)+\frac{1}{9}\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)^2-n_1n_2-2n_2-n_1-5\right)^{\frac{3}{2}}\sqrt{3}}\right)\right)$$

$$\begin{aligned} A_2^{(3)} &= \frac{1}{54}\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right) + n_2 + n_1 + \\ & 4\left(2\left(-2\cos\left(\frac{2\pi k}{q}\right)+n_2+n_1+4\right)^2-9\left(-2(n_2+2)\cos\left(\frac{2\pi k}{q}\right)+2n_2+n_1+5\right)\right) - \\ & \cos\left(\frac{2\pi k}{q}\right)+1 \end{aligned}$$

$$B_1 = \frac{8\pi}{9q} \left(-2 \cos\left(\frac{2\pi k}{q}\right) + n_2 + n_1 + 4 \right) \sin\left(\frac{2\pi k}{q}\right) - \frac{4\pi}{3q} (n_2 + 2) \sin\left(\frac{2\pi k}{q}\right)$$

$\mu_1^{(3)}$ atteint son maximum lorsque $k = \lfloor \frac{2q}{4} \rfloor = \lfloor \frac{q}{2} \rfloor$.

On distingue alors deux cas :

- **q pair** : $k = \frac{q}{2}$, et $\mu_1 = \frac{6+n_1+n_2}{3} + 2\sqrt{\frac{-P}{3}} \cos\left(\frac{\arccos(\frac{-Q}{2}\sqrt{\frac{-27}{P^3}})}{3}\right)$
- **q impair** : $k = \frac{q-1}{2}$, et $\mu_1 = \frac{4+n_1+n_2-2\cos(\frac{2\pi}{q})}{3} + 2\sqrt{\frac{-P}{3}} \cos\left(\frac{\arccos(\frac{-Q}{2}\sqrt{\frac{-27}{P^3}})}{3}\right)$

□

Pour chercher la valeur exacte de la connectivité algébrique des graphes $G(q, n_1, n_2)$, on peut suivre le même procédé qu'auparavant, tout en choisissant $k = q - 1$. Ceci nous donne la proposition suivante, dont la démonstration est similaire à celle de la proposition (5.17).

Proposition 5.18: Soit G un graphe unicyclique du type $G(q, n_1, n_2)$. La connectivité algébrique de G est donnée par

$$\mu_{n-1}(G) = \frac{4 + n_1 + n_2 - 2 \cos(\frac{2\pi}{q})}{3} + 2\sqrt{\frac{-P}{3}} \cos\left(\frac{\arccos(\frac{-Q}{2}\sqrt{\frac{-27}{P^3}}) + 2\pi}{3}\right)$$

où

$$P = \frac{\left(4 + n_1 + n_2 - 2 \cos(\frac{2\pi}{q})\right)^2}{3} + n_1 + 2n_2 + n_1n_2 + 5 - 2(2 + n_2) \cos\left(\frac{2\pi}{q}\right)$$

$$\text{et } Q = -\frac{4+n_1+n_2-2\cos(\frac{2\pi}{q})}{27} \left[2\left(4 + n_1 + n_2 - 2 \cos(\frac{2\pi}{q})\right)^2 - 9(n_1 + 2n_2 + n_1n_2 + 5 - 2(2 + n_2) \cos(\frac{2\pi}{q})) \right] + \left(2 \cos(\frac{2\pi}{q}) - 2\right)$$

La preuve est similaire à celle de la proposition précédente.

Proposition 5.19: Si on pose $n_2 = 0$, on retrouve le premier type de graphes unicycliques $G(q, n_1 = \Delta - 2, 0)$, et les résultats de cette section (avec $n_2 = 0$), coïncident avec ceux de la section 3.

5.5 Conclusion

Dans ce chapitre, nous avons réalisé une étude algébrique qui consiste à calculer les valeurs propres extrêmes et exactes des matrices d'adjacence et du Laplacien de deux classes particulières de graphes unicycliques qu'on a noté par $G(q, n_1, n_2)$.

Dans un proche temps, il sera intéressant de pouvoir généraliser ces calculs pour des graphes unicycliques quelconques, ou alors le cas échéant, de donner de bornes meilleures que celles existantes dans la littérature.

Comme indiqué dans le résumé de ce chapitre, la connaissance des valeurs propres extrêmes permet de renforcer la relaxation linéaire de la formulation entière du problème de synthèse d'un réseau appartenant à l'une de ces deux classes.

En effet, posons X la matrice d'adjacence d'un tel réseau où x_{uv} est une variable binaire indiquant l'existence de l'arête uv . X est supposé être variable. Si le réseau appartient à la classe $G(q, n_1, n_2)$, alors on peut par exemple écrire que $y^t X y \leq \lambda_1 \|y\|^2$ pour tout vecteur y . Ceci nous fournit un ensemble d'inégalités valides qui pourraient renforcer la relaxation courante. On peut faire de même en utilisant les valeurs propres $\lambda_n, \mu_1, \mu_{n-1}$. En effet, le Laplacien s'exprime linéairement par rapport à X puisque $L = X - D$, avec $D_{uv} = \sum_{v \sim u} x_{uv}$.

Signalons enfin que la séparation de ces inégalités valides, se fait par simple calcul des valeurs propres de la matrice X ou du Laplacien $X - D$.

Conclusion générale et perspectives

Nous avons pu étudier différentes variantes du problème de synthèse d'un réseau à composantes connexes unicycliques. Nous avons essentiellement utilisé l'approche polyédrale pour aborder le problème.

Les résultats les plus importants de la thèse sont probablement la polynomialité du problème où on considère une contrainte du type Steiner ainsi qu'une généralisation de l'algorithme de Padberg-Rao. Les résultats du deuxième chapitre sont également intéressants et non triviaux.

Nous essayons de rappeler quelques problèmes ouverts mentionnés dans la thèse.

- Lorsqu'on considère une contrainte sur la taille des cycles (au moins $p + 1$ sommets par cycle), le problème est généralement NP-Difficile. Cependant, on ne connaît toujours pas la complexité si p est inférieur à une constante.
- Nous avons proposé un algorithme d'approximation pour le problème ci-dessus lorsque les graphes sont complets et Euclidiens. Cet algorithme avait 4 comme rapport. Il serait utile d'affiner cette étude en améliorant ce rapport ou/et en donnant des indications sur l'inapproximabilité du problème.
- Nous avons prouvé que la variante dite de Steiner est facile à résoudre en donnant une formulation étendue. Nous avons également présenté une description partielle de l'enveloppe convexe. Compléter cette description est toujours un problème ouvert.
- Le résultat de polynomialité de cette variante est une conséquence des propriétés du matroïde bi-circulaire. On peut se demander si d'autres problèmes de type Steiner relatifs à d'autres matroïdes transversaux sont également faciles à résoudre.
- D'autres types de contraintes techniques ont été considérés. Les problèmes d'optimisation sous-jacents ont été étudiés superficiellement. Il convient d'approfondir l'étude polyédrale.
- Une variante particulièrement intéressante est celle où on considère des contraintes sur les degrés des sommets. La complexité du problème est toujours un problème ouvert.
- Nous avons calculé les valeurs propres extrêmes de la matrice d'adjacence et du Laplacien de certaines classes de graphes unicycliques. Il serait d'une part intéressant d'étudier le problème d'optimisation des réseaux appartenant à ces classes de graphes. D'autre part, l'étude spectrale pourrait être généralisée à d'autres classes de graphes unicycliques.

Bibliographie

Bibliographie

- [1] BAIYOU, M., MAHJOUB, A.R : Steiner 2-arête connected subgraph polytopes on series-parallel graphs. *SIAM J. Discrete Math.* **3**, 505-514 (1997)
- [2] BALDACCI, R., DELL'AMICO, M., SALAZAR GONZÁLEZ, J. : The capacitated m-ring-star problem. *Operations Research.* **55**, 147-1162 (2007)
- [3] BAPAT, R., KIRKLAND, S.J., NEUMANN, M. : On distance matrices and laplacians, *Linear algebra and its applications.* **401**, 193-209 (2005)
- [4] BELARDO F., MARZI, E.M.L., SIMIC, S. : Some results of the index of unicyclic graphs. *Linear algebra and its applications.* **416** 1048-1059 (2006)
- [5] BEN-AMEUR, W. : Constrained length connectivity and survivable networks. *Networks.* **36**, 17-33 (2000)
- [6] BEN-AMEUR, W., OUOROU, A. : Mathematical models of the delay constrained routing problem *Algorithmic Opérations Resaearch.* **1**, 94-103 (2006)
- [7] BEN-AMEUR, W., BELAIDOUNI, M. : A superadditive approach to solve the minimum cost single path routing problem : Prelimarny results INOC 2003. 67-71 (2003)
- [8] BEN-AMEUR, W., GOURDIN, E. : Internet Routing and Related Topology Issues. *SIAM J. Discrete Mathematics,* **17**, 18-49 (2003)
- [9] BEN-AMEUR, W., KERIVIN, H. : Routing of uncertain demands. *Optimization and Engineering,* **3**, 283-313 (2005)
- [10] BEN-AMEUR, PHAM., PHAM, L. : Design of survivable networks based on end to end rerouting. In proceeding of DRCN2001, Budapest (2001)
- [11] BEN-NAIM, E., KRAPIVSKY, P.L. : Unicyclic components in random graphs. *Journal of Physics A : Mathematical and General.* **37** (2004)
- [12] BERMAN, A., FORSTER, K.H. : Algebraic connectivity of trees with a pendant edge of infinite weight. *Electronic journal of linear algebra.* **13**, 175-186 (2005)
- [13] BOROVICANIN, B., GRUNEWALD, S., GUTMAN, I., PETROVIC, M. : Harmonic ghraps with small number of cycles. *Discrete Mathematics.* **265**, 31-44 (2003)
- [14] CHANG, A., TIAN, F. : On the spectral radius of unicyclic graphs with perfect mathings. *Linear algebra and its applications.* **370**, 237-250 (2003)
- [15] CHANG, G.J., JOU, M-J. : The number of maximal independent sets in connected tri-angle free graphs. *Discrete Mathematics.* **197/198** 169-178 (1999)

- [16] CHARDAIRE, P. : Hierarchical two level location problems. In B. Sanso and P. Soriano (Eds). Telecommunications network planning, Kluwer, Boston. 33-54 (1999)
- [17] CHARON, I., GERMA, A., HUDRY, O. : Méthodes d'Optimisation Combinatoire. Masson, Paris. (1996)
- [18] CHEN, W.Y.C., LI, X., WANG, C., ZHANG, X. : Linear time algorithms to the minimum all-ones problem for unicyclic and bicyclic graphs. Discrete mathematics. **17**, 93-98 (2004)
- [19] CHUNG, F.R.K. : Spectral graph theory. Regional conference series in mathematics. **92** (1994)
- [20] COOK, W.J., CUNNINGHAM, W.H., PULLEYBLANK, W.R., SCHRIJVER, A. : Combinatorial Optimization. Discrete Mathematics and Optimization. (1998)
- [21] COULARD, C.R., PULLEYBLANK, W.R. : On cycle cones and polyhedra, Linear Algebra Appl, **114/115**, 613-640, (1989)
- [22] CORNUÉJOLS, G., PULLEYBLANK, W. : A matching problem with side constraint. Discrete Mathematics. **29**, 135-159 (1980)
- [23] CVETKOVIC, D.M., DOOB, M. SACHS, H. : Spectra of graphs, theory and application . (1980)
- [24] CVETKOVIC, D.M., GUTMAN, I. : A new spectral method for determining the number of spanning trees. Publications de l'Institut Mathématique. **43**, 49-52 (1981)
- [25] CVETKOVIC, D.M., SIMIC, S. : On graphs with second largest eigenvalue does not exceed $(\sqrt{5} - 1)/2$. Discrete Mathematics. **138**, 213-227 (1995)
- [26] DAHL, G. : An introduction to convexity, polyhedral theory and combinatorial optimization. (1997)
- [27] DAS, CH. : Sharp upper bounds on the spectral radius of the laplacian matrix of graphs. Acta Math. Univ Comenianae. **2**, 185-198 (2005)
- [28] DAS, CH. : An improved upper bound for laplacian graph eigenvalues . Linear algebra and its applications. **368**, 269-278 (2003)
- [29] DAS, CH. : Characterization of extremal graphs from least non-zero laplacian eigenvalue. Combinatorial and computational mathematics center.cicicicic
- [30] DAS, CH. : Some new bounds on the spectral radius of graphs. Discrete Mathematics. **281**, 149-161 (2004)
- [31] DAS, CH. : The laplacian spectrum of a graph. Computers and mathematics with applications. **48**, 715-724 (2004)
- [32] DE ABREU, N.M.M. : Old and new results on algebraic connectivity of graphs. Linear algebra and its applications. **423**, 53-73 (2006)
- [33] DE CAEN, D. : An upper bound on the sum of squares of degrees in a graph. Discrete Mathematics. **185**, 245-248 (1998)
- [34] DIDI BIHA, M., MAHJOUR, A.R. : Steiner k-edge connected subgraph polyhedra. Journal of Combinatorial Optimization. **4**, 131-144 (2000)
- [35] DIDI BIHA, M., MAHJOUR, A.R. : Steiner k-edge connected polyhedra on series-parallel graphs. Oper. Res. Lett. **19**, 71-78 (1996)

- [36] DIESTEL, R. : Graph Theory. Springer Verlag .Deuxième édition. New York. (2000)
- [37] EDMONDS, J. : Maximum matching and a polyhedron with 0, 1 vertices, J. Research Nat. Bur. Standards. **69** (b), 125-130 (1965)
- [38] EDMONDS, J., JHONSON, E.L. : Matching : a well-solved class of integer linear programs, In : Combinatorial Structures and Their Applications ; Proceedings of the Galgary International Conference on Combinatorial Structures and Their Applications 1969 (R. Guy, H. Hanani, N. Sauer, J. Schonheim, eds.), Gordon and Breach, New York, 69-87, (1970)
- [39] EULER, R., MAHJOUB, A.R. : On a composition of independence systems by circuit-identification. Journal Of Combiatorial Theory Ser. B. **53**, 235-259 (1991)
- [40] FIEDLER, M. : Algebraic connectivity of graphs. Praha .(1972)
- [41] FORTZ, B. : Hand book of optimization in telecommunications. Springer, New York .(2006)
- [42] FUGENSCHUH, A., MARTIN, A. : Computational Integer Programming and Cutting Planes. Handbooks in OR & MS . Vol **12** (2005)
- [43] FORTZ, B., LABBÉ, M., MAFFIOLIF, F : Solving the two-connected network with bounded meshes problem. Oper. Res. **48**, 866-877 (2000)
- [44] FORTZ, B., MAHJOUB, A.R., MCCORMICK, S.T., PESNEAU, P. : The 2-edge connected subgraphs problem with bounded rings. Math. Prog. (2003)
- [45] GAREY, M., JHONSON, D.S. : Computers and Intractability. W.H. Freeman and Company. (1979)
- [46] GHOSH, A., BOYD, S. : Upper bound on algebraic connectivity via convex optimization . Linear algebra and its applications. **418**, 693-707 (2006)
- [47] GODSIL, C., ROYLE, G. : Algebraic graph theory . Graduate texts in Mathematics. (2000)
- [48] GOEMANS, M.X., RAMAKRISHNAN, V.S. : Minimizing submodular functions over families of sets. To appear in Combinatorica. (1993)
- [49] GOEMANS, M.X., WILLIAMSON, D.P. : A general approximation technique for constrained forest problems. SIAM Journal on Computing. **24**, 296-317 (1995)
- [50] GOLDBERG, A.V., TARJAN, R.E. : A new approach to the maximum flow problem. Journal of the ACM. **35**, 921-940 (1988)
- [51] GONDRON, M., MINOUX, M. : Graphes et algorithmes. Editions Eyrolles. (1990)
- [52] GOUVEIA, L., JANSSEN, E. : Designing reliable tree networks with two cable technologies. European Journal of Operations Research. **105**, 552-568 (1998)
- [53] GROSSMAN, J.P. : An eigenvalue bound for the laplacian of a graph. Discrete mathematics. **300**, 225-228 (2005)
- [54] GRAY, R.M. : Toeplitz and circulant matrices : A review. Stanford university.
- [55] GRÖTSCHEL, M., MONMA, C.L., STOER, M. : Design of survivable networks. Handbook in Operations research and Management science. Ball, Monma and Nemhauser (editors). North-Holland. **7**, 617-671 (1995)

- [56] GRÖTSCHEL, M., MONMA, C.L., STOER, M. : Polyderal approaches to networks survivability . Reliability of Computer and Communication Networks. DIMACS Ser. Discrete Math. Theoret. Comput. Sci. 5, F. Roberts, F. Hwang, and C. L. Monma, eds., AMS, Providence, RI.5, 121-141 (1991)
- [57] GRÖTSCHEL, M., MONMA, C.L. : Integer polyhedra arising from certain networks design problems with connectivity constraints. SIAM J. Discrete Maths. **3**, 502-523 (1990)
- [58] GRÖTSCHEL, M., PADBERG, M. : On the symmetric travelling salesman problem I : inequalities. Mathematical Programming. **16**, 265-280 (1979)
- [59] GRÖTSCHEL, M., PADBERG, M. : On the symmetric travelling salesman problem II : lifting theorems and facets. Mathematical Programming. **16**, 281-302 (1979)
- [60] GRÖTSCHEL, M., LOVÁSZ, L., SHRIJVER, A. : Geometric Algorithms and Combinatorial Optimization. Springer. (1985)
- [61] GRUNBAUM, B. : Graphs of polyhedra ; polyhedra as graphs. Discrete Mathematics. **307**, 445-463 (2007)
- [62] GUO, J-M. : On the second largest laplacian eigenvalue of trees . Linear algebra and its applications. **404**, 252-261 (2005)
- [63] GUO, J-M. : A new upper bound for the laplacian spectral eigenvalue of graphs. Linear algebra and its applications. **400**, 61-66 (2005)
- [64] GUO, J-M. : On the laplacian spectral radius of a tree. Linear algebra and its applications. **368**, 379-385 (2003)
- [65] GUO, S-G. : The spectral radius of unicyclic and bicyclic graphs with n vertices and k pendant vertices. Linear algebra and its applications. **408**, 78-85 (2005)
- [66] GUTMAN, I. Relation between the laplacian and the ordinary characteristic polynomial. Communications in mathematical and in computer chemistry. **47**, 133-140 (2002)
- [67] HE, L. LIU, X. STRANG, G. : Laplacian eigenvalues of growing trees. Department of mathematics. Cambridge.
- [68] HENROT, A. : Minimization problem for eigenvalues of the laplacian. Ecole des mines and institut Elie Cartan. Nancy.
- [69] HONG, Y., SHU, J-L : A sharp upper bound of the spectral radius of graphs. Journal of combinatorial theory. **81**, 177-183 (2001)
- [70] HOU, Y., GUTMAN, I., WOO, C-W. : Unicyclic graphs with maximal energy. Linear Algebra and its applications. **356**, 27-36 (2002)
- [71] HOU, Y., TIAN, F. : Unicyclic graphs with exactly two main eigenvalues. Applied mathematics letters. **19**, 1143-1147 (2005)
- [72] <http://wims.auto.u-psud.fr/wims/wims.cgi>
- [73] HU, S. : The largest eigenvalue of unicyclic graphs . Discrete mathematics. **307**, 280-284 (2006)
- [74] ILOG CPLEX. ILOG CPLEX 9.0 user's guide
- [75] IWATA, S., FLEISCHER, L., FUJISHIGE, S. : A combinatorial strongly polynomial time algorithm for minimizing submodular functions. Journal of the ACM. **48**, 761-777 (2001)

- [76] KELMANS, A., YONG, X. : On the distribution of eigenvalues of graphs. *Discrete mathematics*. **199**, 251-258 (1999)
- [77] KERIVIN, H., MAHJOUR, A.R. : Design of Survivable Networks : A survey. *Networks*. **46**, 1-21 (2005)
- [78] KERIVIN, H., MAHJOUR, A.R. : On Survivable Network Polyhedra. *Discrete Mathematics*. **290**, 183-210 (2005)
- [79] KERIVIN, H., MAHJOUR, A.R., NOCQ, C. : (1,2)-survivable networks : facets and branch-and-cut.. *The Sharpest Cut : The impact of Manfred Padberg and His Work, MPS/SIAM Ser. Optim., M. Grotchel, ed.,SIAM, Philadelphia.* 121-152 (2004)
- [80] KORTE, B., VYGEN, J. : *Combinatorial optimization : theory and algorithms*, Springer, (2005).
- [81] LABBÉ, M., GOURDIN, E., YAMAN, H. : A branch and cut algorithm for hub location problems with single assignment. *Mathematical programming*. **102(2)**, 371-405 (2005)
- [82] LABBÉ, M., LAPORTE, G., RODRIGUEZ MARTIN, I., SALAZAR GONZÁLES, J.J. : The ring star problem : polyhedral analysis and exact algorithm. *Networks*. **43**, 177-189 (2004)
- [83] LAPORTE, G., RODRIGUEZ MARTIN, I. : Locating cycles in transportation and telecommunications networks. *Networks*. **50**, 92-108 (2007)
- [84] LATIFI, S., BAGHERZADEH, N. : On embedding ring into a star-related network. *Informatics and computer science*. **99**, 21-35 (1997)
- [85] LAWLER, E.L. : *Combinatorial optimization : networks and matroids*. Dover Publications. (2001)
- [86] LAWLER, E.L., LENSTRA, J.K., RINNOOY KAN, A.H.G., SHMOYS, D.B. : *The Traveling Salesman Problem. Discrete Mathématique and Optimization.* (1990)
- [87] LETCHFORD, A., REINELT, G., THEIS, D.O. : Odd minimum cut-sets and b-matchings revisited, *SIAM J. Discr. Math.*, **22(4)**, 1480-1487 (2008)
- [88] LI, J-S, ZHANG, X-D. : On the laplacian eigenvalues of a graph. *Linear algebra and its applications*. (1998)
- [89] LU, M., LIU, H., TIAN, F. : Bounds of laplacian spectrum of graphs based on the domination number. *Linear algebra and its applications*. **402**, 390-396 (2005)
- [90] LU, M., ZHANG, L-Z., TIAN, F. : Lower bounds of the laplacian spectrum of graphs based on diameter. *Linear algebra and its applications*. **420**, 400-406 (2007)
- [91] LUTTON, J.L., NACE, D., CARLIER, J. : Assigning spare capacities in survivable mesh networks. *Telecommunications systems*. **13**, 441-452 (2000)
- [92] MAHJOUR, A.R. : Two-edge connected spanning subgraphs and polyhedra. *Math. Programming*. **2**, 199-208 (1994)
- [93] MARSH, A.B. : *Matching algorithms*, Ph.D. thesis, Johns Hopkins University, Baltimore, (1979)
- [94] MATTHEWS, L.R. : Bicircular matroids. *Quart. J. Math. Oxford*. **28**, 213-228 (1977)
- [95] MAURRAS, J.F. : Some results on the convex hull of hamiltonian cycles of symmetric complete graphs. *Combinatorial Programming : Methods and Applications*, B. Roy, ed, Reidel. 179-190 (1975)

- [96] MERRIS, R. : Laplacian matrices of graphs : A survey . Linear algebra and its applications. **197 & 198**, 143-176(1994)
- [97] MERRIS, R. : Laplacian graph eigenvectors. Linear algebra and its applications. **278**, 221-236 (1998)
- [98] MERRIS, R. : A note on laplacian graph eigenvalues. Linear algebra and its applications. **285**, 33-35 (1998)
- [99] MINOUX, M. : Network synthesis and optimum network design problems : Models, solution methods and applications. Networks. **19**, 313-360 (1989)
- [100] MOHAR, B. : The laplacian spectrum of graphs. Graph Theory, Combinatorics, and Applications. **2**, 871-898 (1991)
- [101] MOHAR, R. : Semidefinite programming and laplace eigenvalues of graphs . Department of Mathematics. University of Ljubljana. icicicic
- [102] MOHAR, B. : Some applications of laplace eigenvalues of graphs. Graph symmetry : Algebraic methods and applications. 225-275 (1997)
- [103] MOHAR, B. : Eigenvalues, diameter, and mean distance in graphs. Graphs and combinatorics. **7**, 53-64 (1991)
- [104] MONMA, C.L., SHALLCROSS, D.F. : Methods for designing communications networks with certain two-connected survivability constraints. Oper. Res. **37**, 531-541 (1989)
- [105] NACE, D., PIORI, M. : A tutorial on Max-Min fairness and its applications to routing and load balacing in telecommunication networks. IEEE communication surveys and tutorials. à paraitre.
- [106] NIKIFOROV, V. : Bounds on graph eigenvalues I. Linear algebra and its applications. **420**,667-671 (2006)
- [107] OUOROU, A. : Robust capacity assignment in telecommunications. Operation researchs, Submitted.
- [108] OXLEY, J. : Matroid Theory. Oxford University Press, New York. (1992)
- [109] M. W. PADBERG, M.W., RAO, M.R. : Odd minimum cut-sets and b -matchings, Math. Oper. Res. **7**, 67–80 (1982)
- [110] PAN, Y-L. : Sharp Upper bounds for the laplacian graph eigenvalues. Linear algebra and its applications. **355**, 287-295 (2002)
- [111] PATI, S. : The third smallest eigenvalue of the laplacian matrix . The electronic journal of linear algebra. **8**, 128-139 (2001)
- [112] PEDERSEN, A.S, VESTERGAARD, P.D. : The number of independent sets in unicyclic graphs. Discrete applied mathematics. **152**, 246-256 (2005)
- [113] PEETERS, R. : The maximum edge biclique problem is NP-complete. Discrete Applied Mathematics. **131**, 651-654 (2003)
- [114] PETROU, G, LE MARÉCHAL, C, OUOROU, A. : An approach to robust network design in telecommunications. RAIRO Operation research. **41(4)**, 411-426 (2007)
- [115] RIZZI, R. : A simple minimum T-cut algorithm. Discrete applied mathematics. **129**, 539-544 (2003)

- [116] SCHRIJVER, A. : A Combinatorial algorithm minimizing submodular functions in strongly polynomial time. *Journal of Combinatorial Theory B.* **80**, 346-355 (2000)
- [117] SCHRIJVER, A. : *Combinatorial optimization : polyhedra and efficiency.* Springer. (2003)
- [118] SCHRIJVER, A. : *On the History of Combinatorial Optimization.* Handbooks in OR & MS. Vol **12**, (2005)
- [119] SHU, J-L., HONG, Y., WEN-REN, K. : A sharp upper bound on the largest eigenvalue of the laplacian matrix of a graph. *Linear algebra and its applications.* **347**, 123-129 (2002)
- [120] SIMIC, S.K. : On the largest eigenvalue of unicyclic graphs, *Publication de l'institut mathématique.* **56**, 13-19 (1987)
- [121] STEVANOVIC, D. : Bounding the largest eigenvalue of trees in terms of the largest vertex degree . *Linear algebra and its applications.* **360**, 35-42 (2003)
- [122] STOER, M., WAGNER, F. : A simple min-cut algorithm. *Journal of the ACM.* **44**, 585-591 (1997)
- [123] SALAZAR-GONZÁLEZ, J. J. : The Steiner cycle polytope, *European J. Oper. Res.* **147**, 671-679 (2003)
- [124] WOLSEY, L.A. : *Integer Programming. Discrete Mathematics and Optimization.* (1998)
- [125] XU, G-H. : On unicyclic graphs whose second largest eigenvalue dose not exceed 1. *Discrete applied mathematics.* **136**, 117-124 (2004)
- [126] YAMAN, H. : A survey on location problems with applications in telecommunications. *Combinatorial optimization.* **16**, 9-23 (2005)
- [127] YAN, W., YEH, Y-N. : On the nullity and the matching number of unicyclic graphs, Supported by FMSTF (2004J024) and NSFF (E0540007).
- [128] YANNAKAKIS, M. : Edge-deletion problems. *SIAM Journal on Computing.* **10**, 297-309 (1981)
- [129] YU, A., TIAN, F. : On the spectral radius of unicyclic graphs. Institute of systems science. Academy of mathematics and system sciences, China. (2003)
- [130] ZELINKA, B. : The 3-Path-Step operator on trees and unicyclic graphs. *Mathematica Bohemica.* **127**, 33-40 (2002)

Table des figures

1.1	Exemple d'ensembles convexes et d'ensembles non convexes	5
1.2	Représentation d'un polyèdre	6
1.3	Polyèdre , face et facettes	6
1.4	Graphe biparti	20
2.1	Sur la validité de l'inégalité (2.4)	27
2.2	Sur la preuve de la Proposition (2.9)	31
2.3	Saturation de l'inégalité (2.4) : cas 1 et 2	33
2.4	Saturation de l'inégalité (2.4) : cas 3	35
2.5	Saturation de l'inégalité (2.5)	37
2.6	Un algorithme de flot maximum pour séparer les inégalités (2.5)	43
2.7	Sur la fonction Lagrangienne	44
2.8	Une réduction à partir du problème de bi-clique maximum	45
2.9	Évolution du coût du réseau en fonction de p pour l'instance à 32 noeuds.	51
2.10	Évolution du coût du réseau en fonction de p pour l'instance de 23 noeuds.	52
3.1	Construction du nouveau graphe étendu	56
3.2	Graphe Euclidien : $ V = 150$: $ S = 70$ à gauche, et $S = \emptyset$ à droite.	59
3.3	Un point extremum fractionnaire qui satisfait les inégalités (3.3), (3.7) et (3.8)	60
5.1	Graphe unicyclique $G(6, 3, 0)$	80
5.2	La courbe de μ_1 en fonction de k	86
5.3	Graphe unicyclique $G(6, 2, 2)$	87

Liste des tableaux

2.1	Graphes Complets aléatoires et Euclidiens	49
2.2	Graphes complets aléatoire avec un écart type élevé	50
2.3	Graphes complets aléatoires avec faible écart type	51
3.1	Comparaison sur des graphes complets aléatoires et Euclidiens	66
4.1	Graphes aléatoires complets Euclidiens - Nombre Max de Composantes	71
4.2	Graphes aléatoires complets Euclidiens, Séparer a et b	73
4.3	Graphes aléatoires complets Euclidiens-Joindre a et b	74

