



HAL
open science

Heuristiques et approche polyedrale du probleme de voyageur de commerce international

Djawad Bouali

► **To cite this version:**

Djawad Bouali. Heuristiques et approche polyedrale du probleme de voyageur de commerce international. Modélisation et simulation. Institut National Polytechnique de Grenoble - INPG, 1996. Français. NNT: . tel-00004978

HAL Id: tel-00004978

<https://theses.hal.science/tel-00004978v1>

Submitted on 23 Feb 2004

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
présentée par

Djawad BOUALI

Pour obtenir le titre de

Docteur de l'Institut National Polytechnique de Grenoble
(arrêtés ministériels du 5 juillet 1984 et du 30 mars 1992)

Spécialité : Informatique

Formation Doctorale : Recherche Opérationnelle

**Heuristiques et Approche Polyédrale
du Problème de Voyageur de Commerce
International**

soutenue le 12 juillet 1996 devant le jury suivant :

MM	Thomas Liebling	Président
	Matteo Fischetti	Rapporteur
	Jean Fonlupt	Rapporteur
	Gerd Finke	Examineur
	Denis Naddef	Directeur

Thèse préparée au sein du laboratoire LMC-IMAG

Remerciements

Je commencerai par remercier Monsieur le Professeur Thomas Liebling qui nous fait l'honneur de présider ce jury de thèse.

Je remercie également Messieurs les Professeurs Matteo Fischetti et Jean Fonlupt, rapporteurs, et Monsieur le Professeur Gerd Finke pour l'intérêt qu'ils ont porté à mon travail et les nombreuses remarques qu'ils m'ont proposées.

Ma reconnaissance va ensuite à mon directeur de thèse Monsieur le Professeur Denis Naddef. Je tiens à le remercier sincèrement pour ses conseils, sa disponibilité et surtout pour son amitié.

Je remercie également Mademoiselle Wafa Rezig et Messieurs Mohamed Bouzgarrou et Ismaïl Zighem pour leurs nombreuses lectures et commentaires.

Je remercie le laboratoire LMC-IMAG d'avoir mis à ma disposition tous les moyens nécessaires pour la rédaction de ce document.

Je voudrais aussi remercier tous les membres qui composaient le laboratoire ARTEMIS-IMAG pour leurs amitiés.

Enfin, mes remerciements vont à mes amis et ma famille pour le soutien qu'ils m'ont apporté tout au long de cette thèse.

Résumé

Le problème du voyageur de commerce, noté TSP , consiste à trouver un parcours de longueur minimum que doit emprunter un voyageur pour visiter une et une seule fois chaque ville s'il démarre de la ville de son domicile et y revient en fin de parcours. Dans cette thèse, nous étudions une généralisation de ce problème. Si on regroupe les villes par pays, on s'intéresse à un parcours de longueur minimum qui visite une et une seule ville de chaque pays. Cette généralisation est ainsi appelée problème du voyageur de commerce international, noté $ITSP$.

Le $ITSP$ est un problème NP -difficile. Dans une première partie, nous décrivons des heuristiques pour le $ITSP$ et nous évaluons numériquement une nouvelle heuristique pour le TSP basée sur des notions introduites par Glover. Nous décrivons une réduction polynomiale du $ITSP$ au TSP et nous donnons une nouvelle formulation du $ITSP$ en un programme linéaire en nombres entiers.

La deuxième partie est réservée à l'approche de résolution polyédrale. Nous définissons la relaxation graphique du $ITSP$ et nous donnons des résultats sur la dimension et la structure faciale du polyèdre associé. Nous étudions la relation polyédrale qui existe entre le $ITSP$ et sa relaxation graphique et nous introduisons plusieurs classes de facettes du polytope du $ITSP$.

L'étude polyédrale du $ITSP$ nous a permis de développer un algorithme de branchement et de coupe pour sa résolution. Nous présentons des heuristiques et des algorithmes exacts pour la séparation de certaines classes de facettes et nous donnons les principales modifications qu'il faut apporter à un algorithme de branchement et de coupe pour le TSP afin d'obtenir un tel algorithme pour le $ITSP$. Nous finissons cette étude en donnant quelques résultats numériques et une analyse de l'implantation que nous avons réalisée.

Mots clefs : problème de voyageur de commerce international, branchement et coupe, approche polyédrale, facette.

Abstract

This thesis deals with a generalization of the traveling salesman problem in which the nodes are partitioned into clusters which can be thought as countries and the salesman has to visit exactly one node in each cluster.

The problem is to find such a solution of minimum length. We call it the international traveling salesman problem and we denote it by *ITSP*.

First, we give some heuristic for the *ITSP* and a new heuristic for the *TSP* based on notions introduced by Glover. Then, we describe a polynomial reduction of the *ITSP* to the *TSP* and we give a new formulation of the *ITSP* as an integer linear program.

Then, we focus on the polyhedral approach of the *ITSP*. We define the graphical relaxation of the *ITSP* and we give results on the dimension and the facial structure of the corresponding polyhedron. We study the polyhedral relationship between the *ITSP* and its graphical relaxation. We give several classes of facet-inducing inequalities of the *ITSP* polytope and we study some properties of its facets.

This polyhedral study allows us to design a branch and cut algorithm to solve the *ITSP* starting with such an algorithm for the *TSP*. We present some exact algorithms and heuristics for the separation problem of the main classes of facet-inducing inequalities. Computational results are finally reported.

Key words : international traveling salesman problem, branch and cut, polyhedral approach, facet.

Table des matières

Introduction	1
1 État de l'art et Applications	5
1.1 Présentation du problème	6
1.2 État de l'art	8
1.3 Applications	9
1.3.1 Collecte du courrier	9
1.3.2 Distribution à deux étapes	10
1.4 Extensions	11
1.5 Conclusion	12
2 Heuristiques et Algorithmes Exacts	13
2.1 Heuristiques	14
2.1.1 Algorithmes directs	15
2.1.1.1 Heuristiques constructives	15
2.1.1.2 Heuristiques amélioratrices	18
2.1.2 Méthode des chaînes alternées pour le TSP	23
2.1.2.1 Structures	24
2.1.2.2 Algorithme	28
2.1.2.3 Résultats numériques	36
2.1.3 Algorithme de décomposition	38
2.1.3.1 Complexité du <i>ITSP</i>	38
2.1.3.2 Algorithme	39
2.2 Algorithmes exacts	41
2.2.1 Quelques notations	42
2.2.2 Approche par réduction au TSP	43
2.2.2.1 Description de la réduction	43

2.2.2.2	Approche polyédrale via la réduction	46
2.2.3	Méthode de branchement et d'évaluation	48
2.2.4	Approche polyédrale	53
2.2.4.1	Notions polyédrales de base	53
2.2.4.2	Algorithme de coupe	56
2.2.4.3	Une nouvelle formulation du ITSP	59
3	Étude Polyédrale	61
3.1	Définitions et notations	62
3.2	La relaxation graphique	65
3.2.1	Étude de la dimension	67
3.2.2	Structure faciale de $P_0(G)$	71
3.2.2.1	Facettes triviales	72
3.2.2.2	Inéquations de coupe	74
3.2.2.3	Forme triangulaire serrée	75
3.3	Le polytope ITSP(n, p)	77
3.3.1	Ensemble (W, u) -maximal	78
3.3.2	Dimension du polytope ITSP(n, p)	80
3.3.3	Facettes triviales	82
3.3.4	Relation avec le polyèdre graphique	84
3.3.4.1	Forme triangulaire serrée	85
3.3.4.2	Relation avec $P_0(n, p)$	86
3.3.5	Facettes régulières	89
3.3.5.1	Les inéquations d'élimination de sous-tours	92
3.3.5.2	Les inéquations de peignes	93
3.3.5.3	Les inéquations d'arbres de cliques	93
3.3.6	Facettes non régulières	94
3.3.6.1	Les sous-tours non réguliers	94
3.3.6.2	Les arbres de cliques non réguliers	101
3.3.7	Extensions	102
3.4	Conclusion	106
4	Algorithme de branchement et de coupe	107
4.1	Principales phases de l'algorithme	108
4.1.1	Séparation des contraintes	110
4.1.2	Gestion des variables	111
4.1.3	Stratégies de branchement	113

4.2	Adaptation d'un algorithme de TSP	114
4.3	Procédures de séparation	115
4.3.1	Contraintes de sous-tours	116
4.3.1.1	Ensemble minimal	117
4.3.1.2	Ensemble régulier	119
4.3.1.3	Ensemble non régulier	120
4.3.2	Contraintes de peigne	127
4.3.2.1	Peigne régulier	128
4.3.2.2	Contraintes de 2-couplage non réguliers . . .	129
4.3.2.3	Peigne non régulier	131
4.4	Instances et résultats numériques	134
4.4.1	Génération d'instances	134
4.4.2	Résultats numériques	134

Bibliographie**140**

Introduction

Le problème du voyageur de commerce, noté *TSP*, consiste à trouver un cycle hamiltonien de longueur minimum dans un graphe valué. Pendant longtemps, plusieurs auteurs [21, 22, 25, 65] ont tenté de résoudre le *TSP* en utilisant un algorithme de branchement et évaluation (“branch and bound”). Les implantations de Smith et Thompson [89], Volgenant et Jonker [96], et Gavish et Srikanth [31] de cet algorithme, les plus rapides qui existent, résolvent à l’optimum des instances dont la taille dépasse à peine quelques dizaines de sommets. L’utilisation d’un tel algorithme ne permet pas de résoudre des instances du *TSP* de grandes tailles.

Ces dix dernières années, les développements de la théorie des polyèdres ont permis d’envisager à nouveau l’utilisation d’un algorithme de branchement et évaluation pour la résolution d’instances du *TSP* de grandes tailles. Ce regain d’intérêt est dû principalement à l’efficacité de la procédure de coupe qu’on y intègre pour le calcul d’une borne inférieure de l’optimum. Cette procédure utilise la programmation linéaire et une certaine connaissance de la structure faciale du polytope du *TSP*. Elle est l’aboutissement de l’étude polyédrale du *TSP*. Désormais, l’algorithme est dit de branchement et de coupe et l’approche est dite polyédrale.

Plusieurs implantations de cet algorithme ont été réalisées et ont permis de résoudre des instances dont la taille est de plus en plus grande. Au début des années 90, Padberg et Rinaldi [78] ont résolu des instances de plus de 2000 sommets et, tout récemment, Applegate et al. [2] ont résolu des instances jusqu’à plus de 7000 sommets. On notera que cela ne signifie pas que de tels implantations résolvent toutes les instances de moins de 7000 ou même 2000 sommets.

L’efficacité de l’algorithme de branchement et de coupe pour la résolution d’instances du *TSP* de grandes tailles a motivé certains auteurs à proposer la même approche pour la résolution d’autres problèmes d’optimisation combinatoire difficiles de grandes tailles. Des algorithmes de branchement et de coupe ont été proposés pour la résolution de la version non symétrique du *TSP*, du problème de tournées de véhicules [62, 3, 4], du problème du voya-

geur de commerce foncier [29, 7] et bien d'autres problèmes d'optimisation combinatoire *NP*-difficiles.

Cette étude a pour objectif principal de proposer un algorithme de branchement et de coupe pour la résolution d'un autre problème d'optimisation combinatoire. Il s'agit d'un problème *NP*-difficile qui généralise le *TSP*.

Soient $G = (V, E)$ un graphe d'ordre n muni d'une fonction coût sur les arêtes et \mathcal{W} une partition en p éléments de l'ensemble de ses sommets V . Le problème consiste à trouver un cycle élémentaire de longueur minimum qui utilise un et un seul sommet de chaque élément de la partition \mathcal{W} . Il est appelé problème du voyageur de commerce international et il est noté *ITSP*. On supposera que chaque élément de la partition \mathcal{W} constitue un stable de G . Le graphe G est multiparti, il est dit complet lorsque toutes les arêtes possibles sont dans E . On notera *ITSP*(n, p) le polytope associé au *ITSP* dans le cas d'un graphe multiparti complet. Pour un graphe multiparti quelconque G , le polytope est noté *ITSP*(G).

Le *ITSP* est un problème très peu étudié. C'est pour cette raison que notre étude ne s'est pas limitée à l'approche polyédrale. On a voulu une étude générale sur le *ITSP*. Dans cette étude, on sera amené à citer des résultats connus du *TSP* et, même, à l'étudier.

Dans le chapitre 1, on présentera le *ITSP* d'une manière générale et on citera les motivations qui nous ont poussées à l'étudier. On donnera un aperçu général de la littérature qui traite le *ITSP* ainsi que le *TSP* et on décrira des applications citées dans la littérature ainsi que de nouvelles applications. On finira le chapitre en proposant certaines extensions du *ITSP*.

Dans le chapitre 2, on décrit des heuristiques et des algorithmes exacts pour la résolution du *ITSP*. On distinguera deux types d'heuristiques. Celles qui s'appliquent directement à une instance du *ITSP* et celles qui exploitent la structure du *ITSP* comme étant la composition de deux sous-problèmes : le *TSP* et la recherche d'une plus courte chaîne dans un réseau. À cet effet, nous avons développé et testé numériquement une heuristique pour le *TSP* basée sur la notion de chaîne alternée.

Un premier type d'algorithmes exacts est obtenu en utilisant un algorithme exact pour la résolution du TSP dans la réduction polynomiale du $ITSP$ au TSP qu'on décrira. Après que nous ayons montré l'inefficacité d'un tel algorithme, on décrira l'algorithme de branchement et évaluation et l'approche polyédrale pour la résolution du $ITSP$.

Le Chapitre 3 est consacré à l'étude polyédrale. Avant d'étudier le polytope $ITSP(n, p)$, on définit la relaxation graphique du $ITSP$ qui nous permettra d'obtenir des résultats sur la structure faciale de $ITSP(n, p)$. On étudiera la dimension du polytope de la relaxation graphique ainsi que celle d'autres polytopes associés. On définit la forme triangulaire serrée des inéquations induisant des facettes du polytope de la relaxation graphique et on montre que, dans le cas d'un graphe multiparti complet, une inéquation induisant une de ses facettes est soit l'inéquation de non négativité, soit l'inéquation de degré ou soit une inéquation triangulaire serrée.

Après l'étude de la dimension de $ITSP(n, p)$ et des inéquations de non négativité, on montre qu'une facette de $ITSP(n, p)$ est contenue dans $p + 1$ facettes du polytope de la relaxation graphique, dont p facettes sont induites par les inéquations de degré.

On distinguera deux classes de facettes de $ITSP(n, p)$: les facettes régulières et les facettes non régulières. On montrera qu'une inéquation induisant une facette régulière de $ITSP(n, p)$ peut être obtenu à partir d'une inéquation induisant une facette du polytope du TSP et, réciproquement, on montrera qu'à l'exception de quelques inéquations qu'on déterminera toutes les inéquations induisant des facettes du polytope du TSP déterminent des inéquations induisant des facettes régulières de $ITSP(n, p)$. On donnera également des inéquations induisant des facettes non régulières en considérant la généralisation de certaines classes d'inéquations induisant des facettes régulières.

On montrera que toutes les facettes du polytope $ITSP(n, p)$ induites par les inéquations qu'on présente dans ce travail vérifient une propriété qu'on définira et qu'on appellera propriété forte des facettes. On parlera de facettes fortes. On montrera également qu'à l'exception de quelques unes d'entre elles

toutes les les facettes fortes (y compris celles qu'on ne présente pas) vérifient une autre propriété qu'on définira et qu'on appellera propriété *tsp*-forte des facettes. On finira le chapitre en donnant deux opérations sur les inéquations qui permettent de construire de nouvelles inéquations induisant des facettes de $ITSP(n, p)$.

Dans le chapitre 4, on décrira les différentes phases d'un algorithme de branchement et de coupe pour la résolution d'un problème d'optimisation combinatoire quelconque et le TSP en particulier. Le principal objectif de ce chapitre consiste à présenter les différentes modifications qu'il faut apporter à un algorithme de branchement et de coupe pour la résolution du TSP , ainsi que les différents modules qu'il faut rajouter, pour obtenir un algorithme de branchement et de coupe pour la résolution du $ITSP$. On donnera plusieurs heuristiques et algorithmes exacts pour la séparation de certaines inéquations induisant des facettes de $ITSP(n, p)$. Les instances du $ITSP$ qu'on a utilisé pour tester numériquement notre implantation s'obtiennent à partir d'instances du TSP par une procédure de Fischetti et al. [28] qu'on décrira. On finira cette étude en donnant des résultats numériques et une analyse de notre implantation.

Chapitre 1

État de l'art et Applications

Le problème du voyageur de commerce est connu pour sa grande difficulté. Il est l'un des problèmes *NP*-difficiles les plus étudiés. La simplicité de son énoncé et les nombreuses applications qu'il présente en ont fait un problème très connu.

Il s'agit d'un voyageur qui veut visiter un certain nombre de villes une et une seule fois en partant de son domicile pour y revenir en fin de parcours. Son objectif principal est évidemment de réaliser un voyage au moindre coût. Si on suppose que les coûts sont proportionnels aux distances parcourues, le problème consiste à trouver un parcours de longueur minimum. On désignera ce problème par la notation *TSP*.

Ici, on étudie un autre problème plus général qui peut se poser à un autre voyageur aimant davantage l'aventure. Il serait tenté, de visiter, pendant son voyage, exactement une ville de chaque pays. L'objectif est toujours le même : réaliser un voyage au moindre coût.

On appellera cette nouvelle variante le problème du voyageur de commerce international et on utilisera la notation *ITSP* pour le désigner. Le *ITSP* est une généralisation du *TSP*. On retrouve le *TSP* quand chaque pays contient exactement une seule ville.

Puisque le *TSP* en est un cas particulier et qu'il est un problème *NP*-difficile, le *ITSP* est également un problème *NP*-difficile.

On a commencé à s'intéresser au *ITSP* pour la première fois au début des années 70 à travers certaines applications. Il reste cependant un problème très peu étudié. À peine une dizaine d'articles, sur ce sujet, ont été publiés jusqu'à présent. Ceci est dû essentiellement à la difficulté du problème. Il était plus intéressant d'étudier le *TSP*, déjà difficile et pas encore suffisamment traité, comme cas particulier. Par ailleurs, le *TSP* est un standard qui valide les nouvelles approches de résolution. Beaucoup de résultats théoriques et pratiques ont été établis concernant le *TSP* et une littérature riche et variée a été accumulée ces 20 dernières années. Ces connaissances, ainsi que l'apparition de puissants calculateurs et des architectures parallèles, permettent d'envisager, de nouveau, l'étude de certaines généralisations du *TSP* avec des moyens théoriques et pratiques plus performants.

Plusieurs auteurs ont ainsi commencé à étudier les méthodes exactes de résolution de la version non symétrique du *TSP*, notée *ATSP*, du problème de tournées de véhicules [62, 3, 4], noté *VRP*, du problème du voyageur de commerce foncier [29, 7], noté *PCTSP*, ainsi que d'autres problèmes *NP*-difficiles. La plupart des algorithmes exacts de résolution de ces problèmes ont été développés ces 10 dernières années. Ce travail se propose essentiellement de développer un algorithme exact pour la résolution du *ITSP*.

L'objet de ce chapitre est de présenter cette nouvelle généralisation du *TSP*. La section 1.1 est une présentation formelle du *ITSP*. On donnera un aperçu général de la littérature qui traite du *ITSP* ainsi que du *TSP* dans la section 1.2. Des applications citées dans la littérature ainsi que d'autres nouvelles applications sont décrites dans la section 1.3, et certaines extensions du *ITSP* sont envisagées dans la section 1.4. On conclura le chapitre en citant les motivations qui nous ont poussées à étudier le *ITSP*.

1.1 Présentation du problème

Soit $G = (V, E, \mathcal{W})$ un graphe non orienté ; V l'ensemble des sommets , E l'ensemble des arêtes et \mathcal{W} une partition de V . On désigne par n , m et p les cardinalités respectives de V , E et \mathcal{W} . On note W_i le $i^{\text{ème}}$ élément de \mathcal{W} , une arête e de E , d'extrémités x et y , sera notée $[x, y]$. Dans la suite, on utilisera également les termes "ville" et "pays" pour désigner respectivement

un sommet de V et un élément de \mathcal{W} . Un pays qui contient exactement une ville est dit dégénéré et \mathcal{W}^d désignera l'ensemble de ces pays. Chaque pays de \mathcal{W} constitue un stable de G . Le graphe G est un graphe multiparti, on dira aussi que le graphe G est un graphe p -parti.

Graphe international : est un graphe multiparti G muni d'une fonction coût $p : E \rightarrow \mathbb{R}^+$ qui associe un coût $p(e)$ à chaque arête e de E .

Tournée internationale : est un sous-ensemble T d'arêtes d'un graphe international G tel que le graphe partiel induit est un cycle qui utilise une et une seule ville de chaque pays. On appellera poids d'une tournée internationale T la somme des coûts $p(e)$ des arêtes appartenant à T , et on le notera $p(T)$.

La figure 1.1 représente un graphe international à 5 pays et 13 villes. L'ensemble des arêtes qui apparaissent sur la figure forme une tournée internationale.

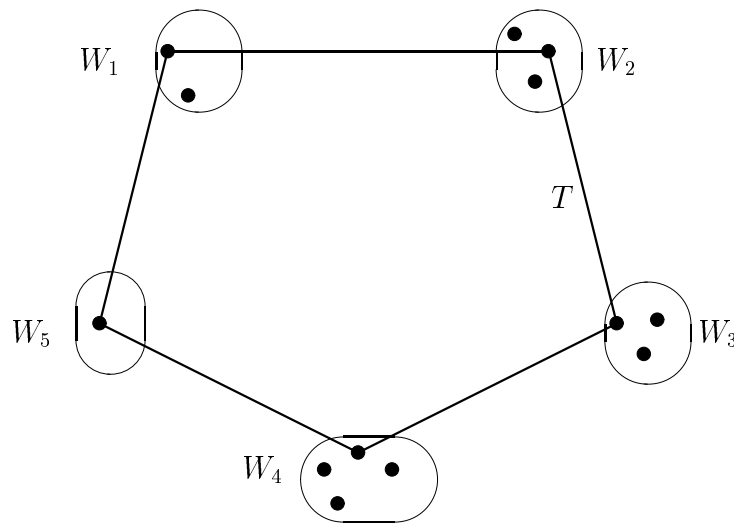


FIG. 1.1 – Une tournée internationale.

Le problème du voyageur de commerce international, sur un graphe inter-

national G , consiste à trouver une tournée internationale de poids minimum. Il s'agit, alors, d'établir un séquençement des pays, puis de choisir un représentant de chaque pays de la manière la plus économique. Ainsi, on peut percevoir le *ITSP* comme étant deux problèmes dépendants. Cette façon de voir le *ITSP* nous a permis de développer l'algorithme décrit dans la section 2.1.3 du chapitre suivant.

Le *ITSP* peut être également perçu comme un ensemble fini de *TSP* de tailles identiques, égales à p . Si on appelle ensemble légal un sous-ensemble de sommets L tel que $|L \cap W| = 1$ pour tout pays $W \in \mathcal{W}$, on notera $TSP(L)$ le *TSP* sous-jacent. On dira que le $TSP(L)$ est induit par l'ensemble légal L . Le graphe international G admet un nombre fini d'ensembles légaux, la résolution de tous les *TSP* induits est équivalente à la résolution du *ITSP*.

Le problème du voyageur de commerce international apparaît dans la littérature sous le nom du problème du voyageur de commerce généralisé et il est noté *GTSP*. Nous préférons utiliser le terme "international" pour qualifier ce problème, au lieu du terme "généralisé", pour deux raisons précises:

- 1- le terme généralisé est utilisé par certains auteurs [8] pour désigner toute une classe de problèmes qui généralisent le *TSP*, alors que le terme international qualifie d'une manière précise la nature de notre problème;
- 2- la notation *GTSP*, qui résulte de l'utilisation du terme généralisé, se confond avec celle de la relaxation graphique du *TSP* qui sera présentée dans le chapitre 3.

1.2 État de l'art

Ces dernières années, le *TSP* a été largement étudié. Ici, on ne citera pas toutes les études qui lui ont été dédiées. Dans [63] et [56], on trouvera un aperçu général de la littérature qui traite du *TSP*.

Le *ITSP* a été présenté pour la première fois au début des années 70, cependant il reste un problème très peu étudié. Laporte et al. [61] ont développé un algorithme de branchement et d'évaluation pour la résolution du

ITSP. Fischetti et al. [27, 28] ont étudié le polytope associé et ils ont implanté un algorithme de branchement et de coupe pour sa résolution. On trouve également dans les articles [86, 88, 72] certains résultats concernant le *ITSP*.

Lorsqu'on considère un graphe international G orienté, le *ITSP* correspondant est dit asymétrique. Cette nouvelle version est encore moins étudiée. Laporte et al. [60] ont développé un algorithme de branchement et d'évaluation pour sa résolution, Noon et al. [73] utilisent la relaxation lagrangienne pour le calcul de la borne inférieure de la solution optimale dans un algorithme de branchement et d'évaluation. Dans la suite, le *ITSP* sera toujours associé à un graphe non orienté.

1.3 Applications

Les premières applications du *ITSP* apparaissent dans la littérature au début des années 70. Le *ITSP* a été présenté comme un modèle mathématique d'optimisation du séquençement des fichiers en informatique [53] et de la gestion des clients dans les centres de sécurité sociale [85]. Depuis, plusieurs applications sont apparues dans la littérature. Dans [72], Noon décrit plusieurs applications dans le domaine de l'approvisionnement avec plusieurs sites de stockage, d'ordonnancement de certains ateliers flexibles, etc...

Les deux sous-sections suivantes sont réservées à la description de deux applications du *ITSP*. La première application simple est décrite dans la littérature. La seconde est une nouvelle application.

1.3.1 Collecte du courrier

Dans [84], Rousseau a modélisé le problème de ramassage du courrier dans les boîtes aux lettres publiques en un *ITSP*.

Le problème consiste à trouver l'itinéraire que doit parcourir un agent de la poste dans un véhicule pour la collecte du courrier dans un certain nombre de boîtes aux lettres. Une contrainte de sécurité est imposée par la direction de l'entreprise responsable de la collecte : un agent ne doit jamais traverser la chaussée pour aller vider une boîte aux lettres. Ceci implique

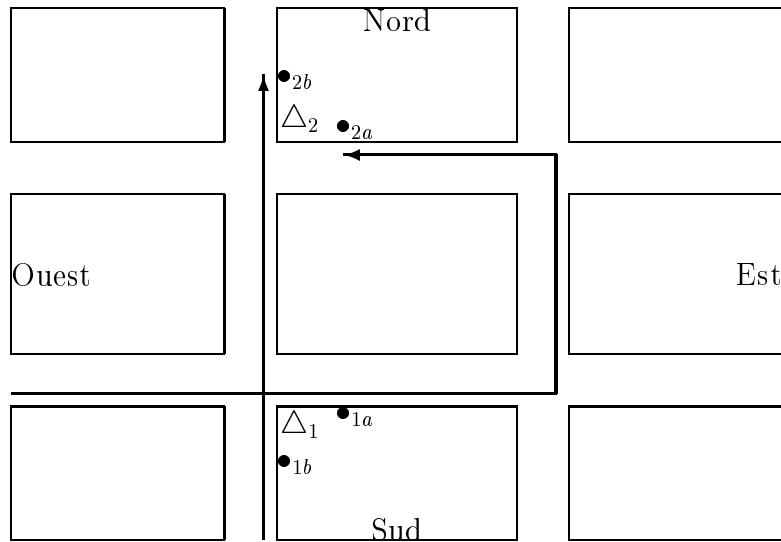


FIG. 1.2 – Collecte du courrier avec emplacements connus.

que, pour une boîte aux lettres située dans un coin de rue donné, la collecte ne peut être réalisée que si le postier est stationné sur l'un des deux côtés du coin de rue. Sur la figure 1.2, on peut facilement voir que pour aller de la boîte 1 à la boîte 2 (matérialisées sur la figure par un triangle), il existe deux chemins de distances différentes. La distance effectivement parcourue par l'agent, pour aller de la boîte 1 à la boîte 2, dépend de son itinéraire. Le problème se modélise en un *ITSP* en remplaçant chaque triangle (boîte), par deux cercles disposés selon les deux possibilités différentes d'atteindre la boîte. Dans [10], Bovet ne suppose pas connu l'emplacement exact d'une boîte aux lettres. Des emplacements potentiels lui sont associés sur les différents coins de rue d'une ou plusieurs intersections. On obtient ainsi un *ITSP* dont la taille des pays est quelconque.

1.3.2 Distribution à deux étapes

Pour arriver aux clients ou aux consommateurs, un produit emprunte un circuit de distribution souvent complexe et fait intervenir plusieurs entreprises de distribution et (/ou) plusieurs modes de distribution. Dans cette application, on s'intéresse aux circuits qui se composent de deux étapes. Ce

type de circuit peut apparaître dans plusieurs domaines. On supposera le cas particulier de la distribution du courrier pour la description de l'application.

Le problème de la distribution du courrier peut se formuler de la manière suivante. La ville est découpée en zones ou quartiers affectés à des facteurs se déplaçant à vélo. La capacité des vélos étant limitée, le facteur est obligé de faire plusieurs tournées pour distribuer tout le courrier du quartier. Une boîte disposée dans chaque quartier servira de dépôt. Un autre postier, en voiture cette fois-ci, effectue la distribution du courrier dans les différentes boîtes (dépôts) de la ville. Sachant qu'il existe, pour chaque quartier, plusieurs emplacements potentiels, le problème d'implantation des différentes boîtes de la ville peut se formuler en un *ITSP*.

1.4 Extensions

Plusieurs extensions du *ITSP* peuvent être formulées. Les plus importantes sont les suivantes :

- 1- le problème de Steiner : étant donné un sous-ensemble \mathcal{N} de pays ($\mathcal{N} \subset \mathcal{W}$), le problème consiste à trouver un cycle de longueur minimum qui utilise une et une seule ville de chaque pays de \mathcal{N} . Une ville qui n'est pas dans un des pays de \mathcal{N} peut apparaître dans la solution. On retrouve le *ITSP* quand \mathcal{N} se confond avec \mathcal{W} ;
- 2- le *ITSP* pondéré : on associe à l'ensemble \mathcal{W} un vecteur k de \mathbb{N}^p . Le problème du *ITSP* pondéré consiste à trouver une tournée de longueur minimum qui utilise exactement k_i villes de chaque pays W_i , pour $i = 1, \dots, p$. On retrouve le *ITSP* lorsque $k_i = 1$, pour $i = 1, \dots, p$ et on retrouve le *TSP* lorsque $k_i = |W_i|$, pour $i = 1, \dots, p$;
- 3- le *GITSP* : au lieu de se restreindre à visiter exactement une seule fois chaque pays, on demande à ce que chaque pays soit visité au moins une fois. En réalité, cette extension du problème est une relaxation appelée le problème du voyageur de commerce international graphique, notée *GITSP*. Dans le chapitre 3, on étudiera le polytope associé à ce problème ainsi que sa relation avec le polytope du *ITSP*.

On peut également définir la version internationale de toutes les extensions et généralisations du *TSP* [8], ainsi que du problème du voyageur de commerce foncier [6], et du problème de tournées de véhicules.

1.5 Conclusion

Le *ITSP* est un problème qui présente plusieurs applications dont quelques unes sont citées dans la section 1.3 (dans [72], Noon décrit un large éventail d'applications du *ITSP*). Il constitue une première étape pour l'étude d'autres généralisations du *TSP*. Cependant, il reste un problème très peu étudié et très peu d'algorithmes se proposent pour le résoudre. C'est pour ces raisons que nous nous proposons d'étudier les différentes approches de résolution du *ITSP* dans le chapitre 2. Un algorithme de résolution exact, basé sur l'étude polyédrale présentée au chapitre 3, sera développé dans le chapitre 4.

Chapitre 2

Heuristiques et Algorithmes Exactes

Les méthodes de résolution exactes ou approchées proposées au cours de ces dernières années sont nombreuses, et sont le reflet de l'éventail des méthodes dont nous disposons pour traiter les problèmes d'optimisation combinatoire. Nous nous proposons, dans ce chapitre, d'étudier les principales approches heuristiques et exactes pour la résolution du *ITSP*. On donnera, chaque fois qu'il est possible, la description générale des méthodes, suivie de l'algorithme adapté au *ITSP*. Généralement, une méthode de résolution est supposée être exacte. Dans ce chapitre, une méthode de résolution désigne également une méthode approchée et on suppose que la fonction coût sur les arêtes vérifie l'inégalité triangulaire.

L'objectif de ce chapitre consiste à réaliser une classification des principales approches de résolution du *ITSP*. On ne comparera pas ces méthodes en fonction de leurs performances numériques, mais on indiquera, chaque fois qu'il est possible, les avantages ou les inconvénients de l'utilisation d'une méthode pour la résolution du *ITSP*.

Dans cette classification, on sera amené, à plusieurs reprises, à citer les approches de résolution du *TSP*. Ces approches sont très nombreuses, le *TSP* a inspiré de nombreux auteurs. Il constitue le terrain d'essai privilégié pour le développement et le test de nouvelles méthodes de résolution en optimisation discrète. Beaucoup de méthodes ont d'abord été dédiées au *TSP*, avant de

devenir des méthodes classiques de l'optimisation en nombres entiers. Dans leur algorithme du *TSP* [65], Little et al. utilisent pour la première fois la méthode "Branch and Bound" qui, par la suite, deviendra la seule méthode généraliste pour la résolution exacte des problèmes *NP*-difficiles.

Dans ce chapitre, on discutera les principales approches de résolution du *ITSP*, selon l'organisation suivante. La première section sera consacrée à l'approche heuristique. On distinguera deux types d'algorithmes approximatifs : les algorithmes appliqués directement à une instance du *ITSP*, dit algorithmes directs, et les algorithmes approximatifs qui utilisent la décomposition du *ITSP* en deux sous-problèmes : le *TSP* et le problème de la recherche d'une plus courte chaîne dans un réseau. À cet effet, on développera une nouvelle heuristique pour la résolution du *TSP*.

La deuxième section, composée de deux parties, traitera l'approche exacte. Dans la première partie, on décrira une réduction polynomiale du *ITSP* au *TSP*. À partir d'un algorithme exact pour la résolution du *TSP*, la réduction permet de développer un premier type d'algorithmes exacts pour la résolution du *ITSP*. Dans la deuxième partie, on présentera un deuxième type d'algorithmes basé sur l'approche classique par branchement et évaluation et on décrira l'approche polyédrale. On finira le chapitre en donnant une nouvelle formulation du *ITSP* en un programme linéaire en nombres entiers.

2.1 Heuristiques

Comme pour tous les problèmes *NP*-difficiles, la recherche d'heuristiques performantes est l'une des principales préoccupations des spécialistes du domaine. Le principe d'une méthode heuristique est de trouver, en un temps raisonnable, une solution réalisable, la meilleure possible.

On distinguera deux types d'algorithmes : les algorithmes appliqués directement à un problème du *ITSP*, dits algorithmes directs, et les algorithmes basés sur la décomposition du *ITSP* en sous-problèmes dépendants, exploitant au mieux la structure du *ITSP*.

2.1.1 Algorithmes directs

Les algorithmes directs consistent en l'adaptation au *ITSP* des différentes méthodes connues. On distinguera les heuristiques constructives et les heuristiques d'amélioration.

2.1.1.1 Heuristiques constructives

L'objectif de ces méthodes, comme il est indiqué dans le titre, est de construire une bonne solution en utilisant des principes souvent très simples. Une panoplie très large de ce type de méthodes existe dans la littérature avec beaucoup de variantes [39, 55]. On distinguera, principalement, trois types de méthodes : celles qui utilisent des principes intuitifs appelées méthodes de base, celles qui utilisent les propriétés d'une tournée internationale et enfin celles qui dérivent de certaines méthodes de résolution du problème de tournées de véhicules [4], appelées méthodes d'économie ("savings"). Dans ce qui suit, on décrira chacune de ces trois méthodes.

a) Méthodes de base

Ce type de méthodes regroupe essentiellement la méthode du plus proche voisin et les méthodes d'insertion.

La méthode du plus proche voisin consiste à construire une tournée internationale selon le schéma suivant. Initialement, on dispose d'une chaîne qui se réduit généralement à un seul sommet. À chaque itération, on augmente la taille de la chaîne courante du sommet le plus proche de l'une de ses extrémités. L'inconvénient de cette méthode est que souvent des pays sont oubliés pour être considérés, après plusieurs itérations, avec des coûts très élevés.

La méthode d'insertion consiste à considérer, à chaque itération, un cycle pour l'étendre d'un seul sommet. Le choix du sommet à insérer dans le cycle peut être effectué selon plusieurs critères plus ou moins gloutons localement.

La complexité de ce type de méthodes est en général en $O(np)$. Souvent, pour améliorer cette complexité, on considère des graphes partiels susceptibles de contenir la solution optimale. Des considérations géométriques permettent le choix de tels graphes partiels.

b) Méthodes d'arbre

Ce type de méthodes a été initialement développé pour la résolution du *TSP* [79, 13] et s'adapte également pour le *ITSP*.

Étant donné une famille d'arêtes T , on définit un raccourci sur un sommet u , de degré supérieur ou égal à 4 dans le graphe partiel $G[T]$, l'opération qui consiste à éliminer de T deux arêtes $[u, v]$ et $[u, w]$ incidentes au sommet u et à ajouter à T l'arête $[v, w]$ (voir figure 2.1).

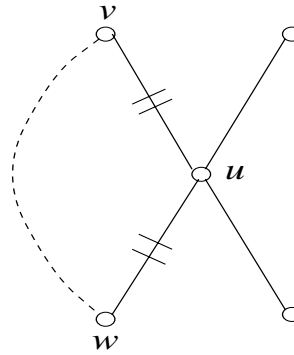


FIG. 2.1 – *Raccourci*.

Un résultat connu établit que, pour toute famille d'arêtes eulérienne T , il existe une séquence de raccourcis telle que, si on l'applique à T , on obtient un cycle hamiltonien H . De plus, le poids de H est, au plus, égal à deux fois celui de T (voir [69]).

La méthode consiste à obtenir un cycle hamiltonien en appliquant le résultat précédent à une famille d'arêtes eulérienne obtenue en dédoublant toutes les arêtes d'un arbre couvrant de coût minimum.

Pour obtenir une famille d'arêtes eulérienne, on peut également considérer l'union de l'arbre de coût minimum et d'un couplage parfait sur l'ensemble des sommets de degré impair dans l'arbre. On peut considérer un couplage de coût minimum [26], mais dans ce cas là, on peut se contenter d'un "bon" couplage obtenu en utilisant une heuristique rapide.

Pour adapter ce principe au *ITSP*, il suffit de considérer un arbre partiel

de poids minimum sur un ensemble légal L , susceptible d'induire la solution optimale du problème. On peut choisir l'ensemble L selon plusieurs critères. On se contentera de citer le critère qui nous semble être le mieux adapté. Dans un arbre couvrant de poids minimum, on choisit, pour chaque pays et parmi les sommets qui le constituent, le sommet qui a le degré le plus élevé. Si le degré maximum est atteint en plusieurs sommets, on choisira parmi ces sommets celui qui maximise la somme des distances des arêtes qui lui sont incidentes. On obtient ainsi un ensemble légal L . Si on calcule l'arbre de coût minimum dans le sous-graphe engendré par l'ensemble légal L , et qu'on applique le résultat précédent, on trouve la solution recherchée.

c) Méthodes d'économie ("savings")

Ce type de méthodes a été initialement développé pour le problème de tournées de véhicules [15] et s'adapte parfaitement au *TSP*.

Pour le cas du *ITSP*, on choisit, dans une première phase, un ensemble légal L dont le sous-graphe engendré est susceptible de contenir la solution optimale ; on utilise par exemple la procédure décrite dans la méthode d'arbre.

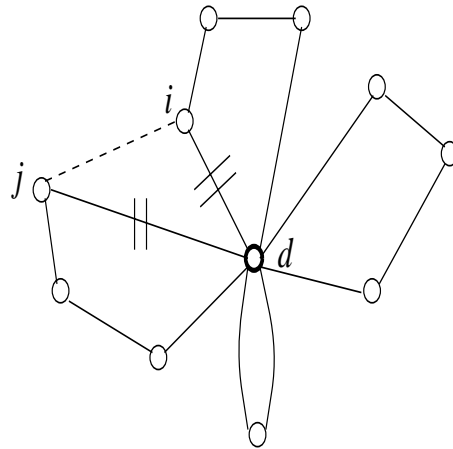


FIG. 2.2 – *Fusion de cycles.*

Initialement, on dispose d'un ensemble T d'arêtes eulérien sur L , dont le degré de tous les sommets vaut 2, à l'exception d'un sommet particulier

d dont le degré est quelconque. Ainsi, l'ensemble T est l'union de cycles arêtes-disjoints qui se rencontrent au sommet d . On définit la fusion de cycles comme étant l'opération qui consiste à éliminer deux arêtes $[d, i]$ et $[d, j]$ appartenant à deux cycles différents et à ajouter l'arête $[j, i]$ à l'ensemble T (voir figure 2.2). On associe à une fusion la quantité égale à $p([d, i]) + p([d, j]) - p([j, i])$, appelée économie ("saving"), p étant la fonction poids sur les arêtes. À chaque itération on applique une fusion ayant une économie maximum. Ainsi, le nombre de cycles dans T diminue d'une unité à chaque itération. L'algorithme s'arrête lorsqu'on obtient un cycle hamiltonien sur L . En général, on choisit initialement un ensemble T tel que tous les cycles correspondent à une arête double.

Ce type de méthodes se caractérise par une certaine robustesse de la solution obtenue, contrairement aux méthodes précédentes, qui peuvent donner de très bonnes mais aussi de très mauvaises solutions.

2.1.1.2 Heuristiques amélioratrices

Contrairement aux méthodes constructives dont l'objectif est de construire une solution, les méthodes d'amélioration modifient une solution initiale, en vue d'améliorer sa valeur. Cette solution initiale est souvent le résultat d'une méthode constructive ; un algorithme général sera composé de deux phases : une méthode constructive suivie d'une méthode d'amélioration.

La plupart de ces méthodes utilisent la notion de voisinage. Il s'agit de trouver, à chaque itération, une "bonne" solution parmi l'ensemble des solutions qui définissent un voisinage d'une solution courante. Parmi ces méthodes, on distingue celles qui améliorent, à chaque itération, la valeur de la fonction économique, dites méthodes de descente, et celles qui permettent de choisir une solution qui n'améliore pas forcément la valeur de la fonction économique. D'une manière générale, le principal inconvénient des méthodes de descente est qu'elles donnent souvent des solutions correspondant à des optima locaux qui ne sont pas de très bonne qualité, alors que, en choisissant une solution qui n'est pas de descente, on peut sortir des minima locaux.

Le voisinage d'une solution peut être défini de plusieurs manières différentes. Ici, on retiendra le voisinage le plus connu et le plus utilisé : celui qui

représente l'ensemble des solutions que l'on peut atteindre, à partir d'une solution donnée, en effectuant une opération de type k -opt. Celle-ci consiste à éliminer k arêtes pour obtenir k chaînes, puis à ajouter k autres arêtes pour obtenir un cycle hamiltonien. La figure 2.3 illustre une opération 2-opt ($k = 2$).

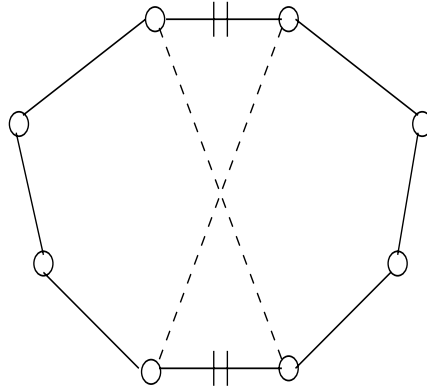


FIG. 2.3 – *Opération 2-opt.*

Le nombre d'opérations k -opt augmente d'une manière exponentielle avec k . Dans la pratique, seules les opérations 2-opt et 3-opt sont considérées.

Les méthodes de type k -opt consistent simplement à rechercher, à chaque itération, la meilleure solution du voisinage k -opt d'une solution courante. La solution trouvée est alors solution courante de la prochaine itération. Le processus s'arrête lorsque aucune amélioration n'est constatée. La méthode est de descente.

Dans ce qui suit, on présentera brièvement la méthode tabou et le recuit simulé qui explorent également un voisinage, en général un k -opt, mais pas en descente. Dans ce cas, on parle de méta-heuristiques. On présentera également, et un plus longuement, un algorithme génétique pour le *ITSP*.

a) La méthode tabou

Cette méthode a été élaborée par Glover [33, 34, 37], elle est basée sur la notion de mouvements interdits (tabou). Chaque itération consiste à trouver le mouvement qui nous donne la meilleure solution dans le voisinage de la

solution courante ; sachant que certains mouvements sont interdits. Parfois, on choisit une solution qui détériore légèrement la solution courante pour sortir des minima locaux. L'inverse du mouvement effectué à chaque itération est rajouté dans une liste, appelée liste tabou, qui contient les mouvements interdits. Initialement, la liste tabou est vide.

b) Le recuit simulé

Le recuit simulé est basé sur un algorithme de simulation de recuit de métaux et s'inspire de modèles de la physique statistique [66]. Il a été développé par S.Kirkpatrick [59].

L'idée principale consiste à autoriser, à chaque itération, le choix d'une solution dont la qualité décroît. À chaque itération, on considère aléatoirement une solution dans le voisinage de la solution courante. La solution considérée est choisie comme solution courante de la prochaine itération si la fonction économique décroît. Sinon, on lui affecte une probabilité de sélection donnée par l'expression $e^{\frac{\Delta H}{T}}$ dérivée de la physique statistique, avec ΔH la différence entre la solution considérée et la solution courante, et T un nombre positif qui représente une température. Comme dans le phénomène physique, on génère des températures qui décroissent. En général, on considère une décroissance linéaire ou en escaliers.

Des résultats récents sur cette technique ainsi que des comparaisons avec d'autres méthodes de voisinage comme la méthode tabou sont donnés dans [94, 95, 92, 1].

c) Les algorithmes génétiques

Ce type d'algorithmes a été proposé, pour la première fois, au milieu des années 70 par Holland [54] et, dans [93], Aarts et al. présentent un algorithme génétique pour la résolution du *TSP*. Une collection d'articles et une bibliographie très complète se trouvent dans Davis [23, 24].

Ces algorithmes se proposent d'imiter la sélection naturelle et les méthodes d'amélioration génétiques de la théorie de l'évolution. La terminologie de la génétique a été également empruntée pour la description de tels algorithmes.

Contrairement à la méthode tabou et au recuit simulé qui manipulent une seule solution, les algorithmes génétiques considèrent un ensemble de solutions (individus) appelé population. Le but de la méthode est de faire évoluer la population en effectuant des mutations et des croisements suivis de sélection d'individus. La mutation est une opération unaire qui modifie la structure d'un individu (semblable à la mutation génétique). Le croisement est une opération binaire, qui à partir de deux individus, en produit deux nouveaux ("cross-over" pour le croisement des gènes). Pour conserver une population de taille raisonnable, généralement entre une dizaine et une centaine de solutions, après un nombre donné de mutations et de croisements, on éliminera certaines solutions aléatoirement, avec une plus grande probabilité pour les moins bonnes. C'est la phase de sélection.

Pour le cas du *ITSP*, l'opération de mutation peut être réalisée de deux manières différentes. On peut conserver l'ensemble légal définissant la solution et ne modifier que l'ordre de parcours des sommets en utilisant par exemple des opérations 2-opt ou 3-opt. On peut également conserver l'ordre de parcours des pays et ne modifier que les représentants de certains pays.

L'opération de croisement fait intervenir deux solutions dont les ensembles légaux induits sont identiques ou différents. On peut construire une nouvelle solution à partir de ces deux solutions de plusieurs manières. Par exemple, on peut considérer le sens de parcours des pays de la première solution et les représentants des pays de la deuxième solution ; ce qui détermine une nouvelle solution. L'algorithme génétique s'exprime de manière générique comme suit :

Procédure Génétique;

Début

Choisir p solutions initiales H_0, \dots, H_p ;

Tant que (critère d'arrêt non vérifié) faire

 Début

 Évolution (q individus supplémentaires);

 Sélection (revenir à p individus);

 Fin

Fin.

Quelques aménagements de l'algorithme sont possibles et, en particulier,

l'utilisation de méthodes amélioratrices (tabou, recuit simulé, etc.) pour faire évoluer les individus, après la phase de mutation et de croisement, en vue d'avoir des solutions dans des minima locaux.

L'idée de base de ces algorithmes réside dans le fait que travailler avec une population permet d'identifier et d'explorer les propriétés qu'ont en commun les bonnes solutions.

Les problèmes liés à cette méthode sont extrêmement nombreux. Le premier est, bien entendu, la taille de la population et sa création. Prendre une population nombreuse ralentirait considérablement la phase de mutation et de croisement alors qu'une taille réduite entraînerait un manque de diversité, ce qui est contraire à la philosophie de la méthode. Le deuxième problème apparaît quand la taille de l'instance traitée devient importante. Dans ce cas, deux solutions qui admettent un petit nombre de sommets visités en commun peuvent avoir des valeurs de fonction économique très proches alors qu'elles sont très différentes. Il est alors difficile de trouver des propriétés fortes caractérisant les bonnes solutions.

L'utilisation du calcul parallèle permettra certainement d'améliorer l'efficacité de tous les algorithmes décrits précédemment. Sur chaque processeur d'une machine parallèle, on résout un *TSP* induit par un ensemble légal L convenablement choisi. On développe ainsi un algorithme parallèle à granularité constante (la répartition des tâches sur les différents processeurs est équilibrée) qu'on peut facilement implanter sur une machine parallèle synchrone (processeur de même performance).

Dans la suite, on présentera une nouvelle méthode basée sur un nouveau voisinage qui généralise le principe de Lin et Kernighan [64]. Les développements et l'expérience numérique sont dédiés au *TSP* pour des raisons toutes simples. D'abord, ce type de méthodes est relativement nouveau. Glover est le seul à étudier ce nouveau voisinage et, à notre connaissance, aucune expérience numérique n'a été réalisée. De plus, cela nous a permis d'obtenir un algorithme pour le *TSP* que l'on a intégré dans l'algorithme de décomposition du *ITSP* qui fera l'objet de la section 2.1.3.

2.1.2 Méthode des chaînes alternées pour le TSP

Ce type de méthodes se base sur une opération élémentaire, dite opération de base. Celle-ci consiste à ajouter à un cycle hamiltonien une arête $[t, x]$, et lui enlever une arête $[x, y]$ (voir figure 2.4).

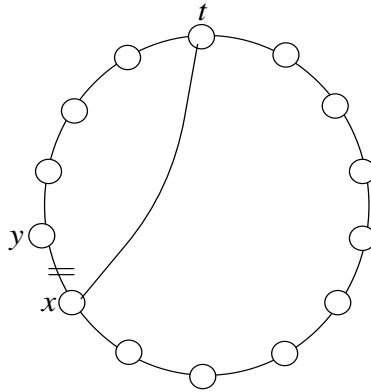


FIG. 2.4 – *Opération de base.*

On peut appliquer l'opération de base à un cycle hamiltonien, un certain nombre de fois, selon des principes différents, chacun d'eux définissant un algorithme. Lin et Kernighan [64] sont les premiers à avoir étudié ce type de méthodes et ont développé un algorithme très efficace qui porte leur noms. Par la suite, Glover a repris les idées de base de Lin et Kernighan, dans [35], et avec Punnen dans [36], pour les développer et les approfondir notamment sur le plan combinatoire. Le développement qui suit se base sur des notions fondamentales introduites par Glover [35].

En appliquant l'opération de base à un ensemble initial d'arêtes, un certain nombre de fois, l'ensemble des arêtes ajoutées et éliminées peut être de nature très différente. Ici, on s'intéresse aux méthodes dont l'ensemble des arêtes ajoutées et éliminées constitue une chaîne à chaque itération de l'algorithme. La chaîne est une séquence d'arêtes qui sont alternativement ajoutées et éliminées.

Le cycle hamiltonien n'est pas conservé après application d'une opération de base. On préfère appliquer ces opérations à des ensembles d'arêtes dont la

structure diffère très légèrement du cycle hamiltonien et reste conservée après application d'une opération de base. De ce fait, l'estimation du voisinage, ainsi que la recherche locale, deviennent plus simple.

Dans une première phase, on décrit les structures qui sont à la base de l'algorithme que nous avons développé. Par la suite, on définit la notion de chaîne alternée ordonnée qui va guider notre algorithme dans la recherche locale, et on finit par donner des résultats numériques comparant notre méthode avec d'autres méthodes classiques.

2.1.2.1 Structures

On distinguera deux structures qui diffèrent entre elles par une seule arête. Pour chaque structure, on donnera les règles d'application de l'opération de base.

Définition 2.1.1 *Un lasso est un ensemble d'arêtes constitué par l'union d'une chaîne hamiltonienne d'extrémités s et t et d'une arête $[s, r]$. On retrouve le cycle hamiltonien lorsque le sommet r s'identifie au sommet t . On dit que le cycle hamiltonien est un lasso dégénéré.*

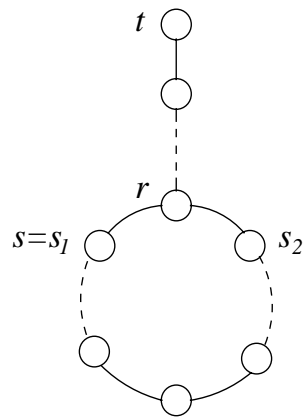


FIG. 2.5 – Un lasso.

Ainsi défini, un lasso peut être considéré comme l'union d'une chaîne élémentaire et d'un cycle comme l'indique la figure 2.5. La chaîne et le cycle

se rencontrent au sommet r .

Les sommets t et r sont respectivement appelés terminaison et racine du lasso, et les sommets s_1 et s_2 , adjacents à r et appartenant au cycle du lasso, sont appelés sous-racines du lasso.

Un lasso engendre deux solutions, dites solutions triviales du lasso, obtenues en enlevant l'arête $[s_i, r]$ et en ajoutant l'arête $[t, s_i]$ respectivement pour i égal à 1 et à 2.

Les deux règles suivantes définissent les différentes manières d'appliquer l'opération de base à un lasso (voir figure 2.6).

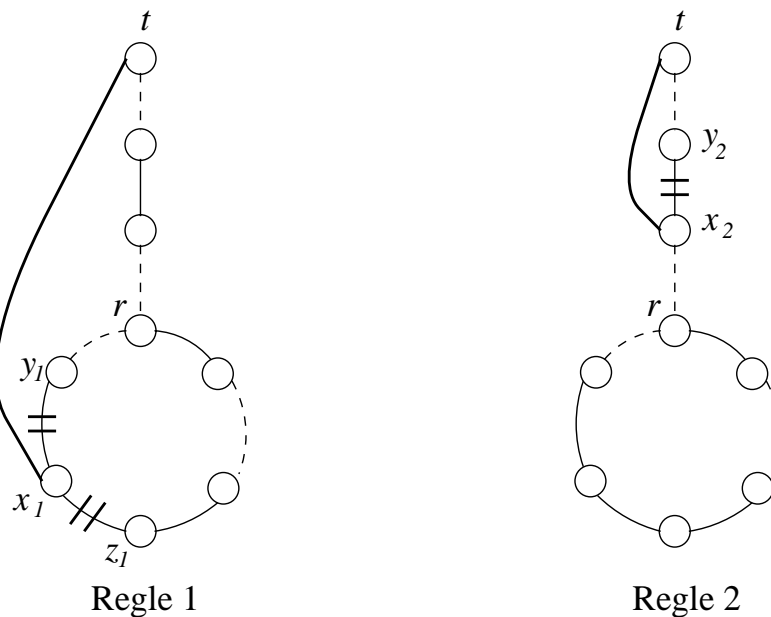


FIG. 2.6 – Opérations sur un lasso.

Règle 1: on ajoute l'arête $[t, x_1]$ et on élimine l'arête $[x_1, y_1]$ ou $[x_1, z_1]$; x_1 étant un sommet appartenant au cycle du lasso, et y_1, z_1 étant ses deux voisins dans le cycle.

Règle 2: on ajoute l'arête $[t, x_2]$ et on élimine l'arête $[x_2, y_2]$ avec x_2 un sommet appartenant à la chaîne du lasso, et y_2 son successeur dans la chaîne le plus proche du sommet t .

Il faut remarquer que la structure de lasso est préservée après l'application de la règle 1 ou de la règle 2. Si n est le nombre de sommets, ces deux règles définissent un voisinage dont la taille est égale à $2n$. La recherche locale s'effectue en $O(n)$ opérations.

Les mouvements décrits dans les deux règles précédentes sont très proches du mouvement élémentaire de Lin et Kernighan [64], noté LK , qui s'applique également à un lasso. Dans les deux règles 1 et 2, on ajoute une arête et on élimine une autre pour faire varier la terminaison (la racine reste invariable), tandis qu'un mouvement LK consiste à éliminer une arête $[r, s]$, avec s une sous-racine, et ajouter une arête quelconque pour faire varier la racine (la terminaison reste invariable). L'idée de base de l'algorithme de Lin et Kernighan consiste à construire, à partir d'un cycle hamiltonien initial (lasso dégénéré), une suite de lassos, en effectuant, à chaque fois, un mouvement LK , tel que le coût de l'arête ajoutée est plus petit que celui de l'arête éliminée. On retiendra la meilleure solution parmi toutes les solutions triviales associées aux lassos générés. Souvent, on impose que les arêtes éliminées n'aient pas été ajoutées auparavant.

L'algorithme que nous avons développé se base sur une structure qui diffère très légèrement du lasso et qu'on appellera lasso fermé; sa gestion ne demande pratiquement aucun effort supplémentaire. Cette nouvelle structure nous permettra d'élargir le voisinage.

Définition 2.1.2 *Une généralisation d'un lasso de terminaison t et de racine r_1 consiste à lui ajouter une arête de type $[t, r_2]$. La structure ainsi obtenue est appelée lasso fermé.*

On distinguera deux types de lassos fermés selon que le sommet r_2 appartient à la chaîne ou au cycle du lasso. Si le sommet r_2 appartient à la chaîne du lasso, on obtient un bicycle et si le sommet r_2 appartient au cycle

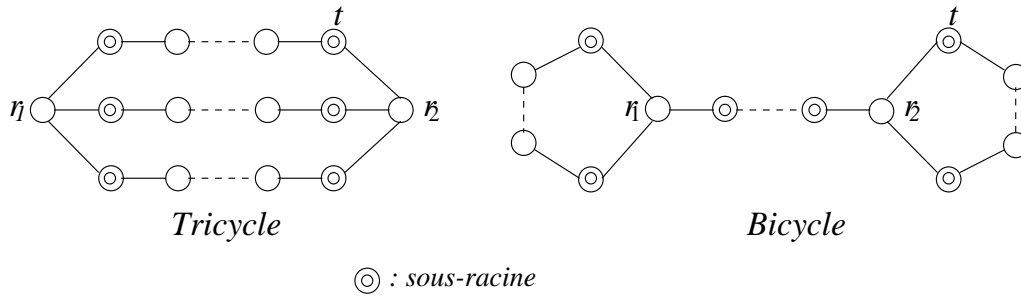


FIG. 2.7 – Les deux types de lasso fermé.

du lasso, on obtient un tricycle. La figure 2.7 représente ces deux types de lassos fermés.

Le lasso fermé est appelé également structure à double racines. Les sommets r_1 et r_2 sont appelés racines du lasso fermé, et les sommets qui leurs sont adjacents sont appelés sous-racines du lasso fermé (voir figure 2.7). On dira qu'une sous-racine est de cycle si elle appartient à un cycle.

Un lasso fermé engendre plusieurs lassos en enlevant à chaque fois une arête entre une racine et une sous-racine de cycle. En considérant toutes les solutions triviales de ces lassos, on peut en déduire jusqu'à 6 solutions différentes, dites solutions triviales du lasso fermé.

Les deux règles suivantes, illustrées dans la figure 2.8, définissent les différentes manières d'appliquer l'opération de base à un lasso fermé selon qu'il s'agisse d'un tricycle ou d'un bicycle.

Règle 3 : à un tricycle, on élimine l'arête $[r, s]$, avec r une racine et s une sous-racine adjacente à la racine r , et on ajoute l'arête $[s, i]$ avec i un sommet quelconque.

Règle 4 : à un bicycle, on enlève l'arête $[r, s]$, avec r une racine et s une sous-racine n'appartenant pas à un cycle, et on ajoute l'arête $[s, i]$ avec i un sommet quelconque appartenant au même cycle que la racine considérée.

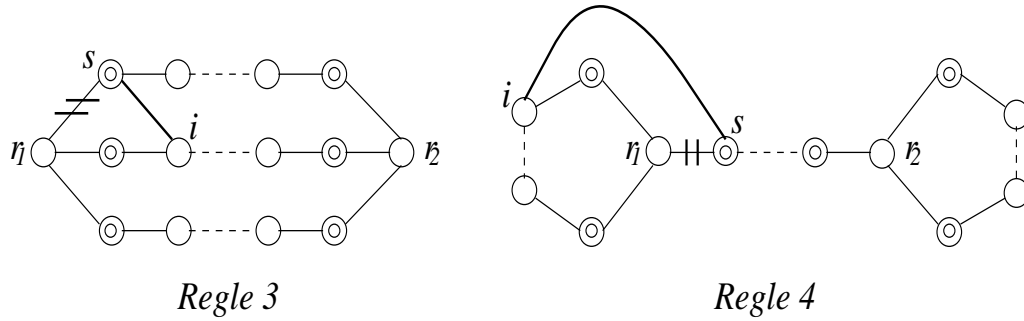


FIG. 2.8 – Opérations sur un lasso fermé.

On remarque que, dans ce cas là également, la structure du lasso fermé est conservée après l'application de la règle 3 ou de la règle 4, et qu'une racine varie tandis que l'autre reste invariable. La taille du voisinage est toujours en $O(n)$, et un effort en $O(n)$ opérations suffit pour effectuer la recherche locale. Dans la suite, on montre qu'on peut élargir le voisinage d'une manière exponentielle tout en maintenant une recherche locale polynomiale.

L'idée de base de notre algorithme consiste à appliquer à une structure de lasso fermé un certain nombre de fois l'opération de base en chaîne alternée. Il reste alors à définir le processus qui va guider notre algorithme dans cette tâche. On définira des restrictions qui vont nous permettre de maîtriser le processus et d'évaluer l'algorithme, notamment sur le plan combinatoire.

2.1.2.2 Algorithme

Au début de chaque itération, on suppose que l'on dispose d'un cycle hamiltonien tel que les sommets sont numérotés de 1 à n en parcourant le cycle dans le sens contraire des aiguilles d'une montre, comme l'indique la figure 2.9. Ceci n'engendre aucun effort supplémentaire au niveau de l'implantation informatique. Il ne s'agit pas de renuméroter les sommets, il suffit simplement de considérer le rang des sommets dans le tableau unidimensionnel où est mémorisé le cycle hamiltonien.

Dans cet algorithme, on considère des chaînes alternées particulières. Ces chaînes sont dites ordonnées et sont notées *CAO*. Leur définition est la suivante : une *CAO* est une chaîne alternée d'extrémités i et j ($i < j$) telle que

la première et la dernière arête sont des arêtes ajoutées, et telle que si on oriente toutes les arêtes de la chaîne de i vers j , chaque arête $[k, l]$ ajoutée est orientée de k à l si $k < l$, et au plus une arête éliminée $[k, k + 1]$ est orientée de k à $k + 1$. Les figures 2.9, 2.10 et 2.11 représentent des *CAO*. On remarquera que toute *CAO* d'extrémités i et j engendre un lasso fermé de racines i et j .

Le voisinage que l'on considère est l'ensemble des solutions triviales associées aux structures en lasso fermé qu'on peut générer par des *CAO* d'extrémités i et j , pour i allant de 1 à $n - 2$, et j allant de $i + 2$ à n . L'objectif de l'algorithme est de trouver un bon cycle hamiltonien dans ce voisinage.

Il est clair que, pour deux sommets quelconques i et j , il existe plusieurs *CAO* ayant pour extrémités les mêmes sommets i et j . À chaque *CAO*, on associe une longueur égale à la somme algébrique des distances de ces arêtes, avec des distances positives pour les arêtes ajoutées, et des distances négatives pour les arêtes éliminées.

Les *CAO* qui ont exactement une arête éliminée $[k, k + 1]$, orientée de k à $k + 1$, génèrent un bicycle et sont dites brisées (voir figure 2.11). Les autres *CAO* génèrent un tricycle. Parmi ces dernières, on distingue celles qui ont un nombre pair d'arêtes ajoutées, dites paires, et celles qui ont un nombre impair d'arêtes ajoutées, dites impaires (voir respectivement les figures 2.9 et 2.10).

Pour obtenir une bonne solution du voisinage du cycle hamiltonien courant, on s'intéressera aux solutions triviales associées aux lassos fermés engendrés par les plus courtes *CAO* de chaque type et pour toute paire de sommets i et j ($j \geq i + 2$).

Ainsi, pour toute paire de sommet i et j ($j \geq i + 2$), on calculera les plus courtes *CAO* impaires, paires et brisées, d'extrémités i et j . Le calcul des plus courtes chaînes s'effectue d'une manière progressive et dynamique sachant que les sous-chaînes d'une plus courte chaîne sont également des plus courtes chaînes.

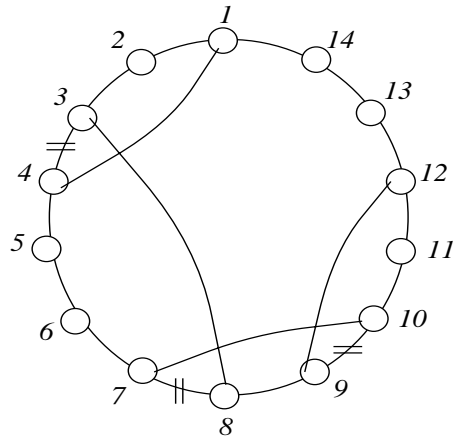


FIG. 2.9 – Une chaîne alternée paire.

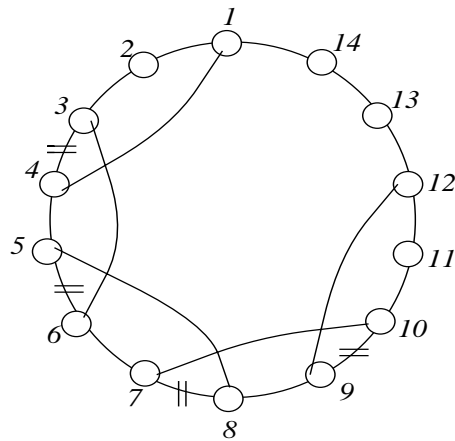


FIG. 2.10 – Une chaîne alternée impaire.

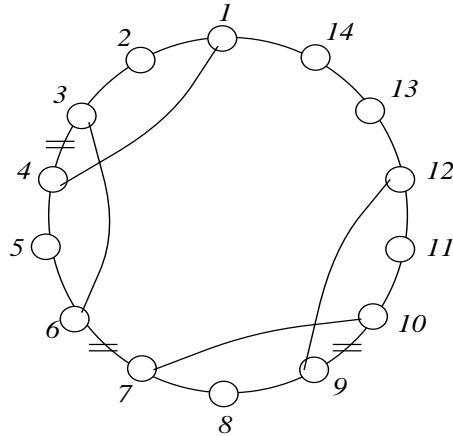


FIG. 2.11 – Une chaîne alternée brisée.

Pratiquement, dans une itération, on fixe le sommet i au sommet 1, et on laisse variable le sommet j . Le sommet 1 est, alors, racine de tous les lassos fermés engendrés par les *CAO* d'extrémités les sommets 1 et j .

Pour implanter la procédure qui génère les plus courtes *CAO* de chaque type, dans le cas d'une racine fixe, on définit trois tableaux *CAO_impaire*, *CAO_paire* et *CAO_brisée* de taille n chacun associés respectivement aux types impaire, paire et brisée des *CAO*. Un élément i quelconque d'un tableau donné est composé de deux champs *Long* et *Pred* représentant respectivement la longueur de la plus courte chaîne de la racine fixe au sommet i du type correspondant au tableau et le prédécesseur du sommet i dans cette plus courte chaîne. On peut exprimer cette procédure d'une manière générique comme suit :

Procédure Génération_de_Chaînes;

Début

Pour i allant de 3 à $n - 1$ faire /* Initialisation */

CAOimpaire[i].*Long* = $p(1, i)$;

CAOimpaire[i].*Pred* = 1;

Fin Pour

Pour i allant de 2 à $n - 1$ faire

 /* Extension des *CAO* impaires */

 Pour j allant de $i + 2$ à n faire

```

        Etendre(1, Impaire, Paire , i, j);
        Etendre(1, Impaire, Brisée , i, j);
    Fin Pour
    /* Extension des CAO impaires */
    Pour j allant de i + 2 à n faire
        Etendre(0, Paire, Impaire , i, j);
        Etendre(1, Paire, Brisée , i, j);
    Fin Pour
    /* Extension des CAO brisées */
    Pour j allant de i + 2 à n faire
        Etendre(0, Brisée, Brisée , i, j);
    Fin Pour
Fin Pour
Fin.

```

Les arguments de la fonction *Etendre* qui apparaît dans la procédure *Génération_de_chânes* sont, dans l'ordre, *Sense*, *Type*, *Nouveau_Type*, *Som1* et *Som2*. L'objectif de cette fonction consiste à comparer la longueur de la *CAO* courante d'extrémités 1 et *Som2* de type *Nouveau_Type* à la longueur de la *CAO* de mêmes extrémités et de même type, obtenu en étendant la *CAO* d'extrémités 1 et *Som1* de type *Type* par élimination de l'arête [*Som1*, *Som1* + 1] et l'ajout de l'arête [*Som1* + 1, *Som2*] si la valeur de l'argument *Sense* est 0, ou par élimination l'arête [*Som1*, *Som1* - 1] et l'ajout de l'arête [*Som1* - 1, *Som2*] si la valeur de l'argument *Sense* est 1. On retiendra cette dernière chaîne comme étant la chaîne courante d'extrémités 1 et *Som2* de type *Nouveau_Type* si sa longueur est plus petite.

La procédure *Génération_de_chânes* calcule les plus courtes *CAO* de chaque type, et ceci pour toute paire de sommets 1 et *j* ($j > 2$). On détermine la meilleure solution triviale des lassos fermés associés à ces plus courtes *CAO*, et on réitère la procédure en considérant comme solution initiale la meilleure solution de l'itération précédente. Lorsqu'aucune amélioration n'est constatée avec la racine fixe courante, on considèrera comme racine fixe le sommet suivant dans la solution courante.

Si aucune amélioration n'est constatée en faisant varier la racine fixe de 1 à

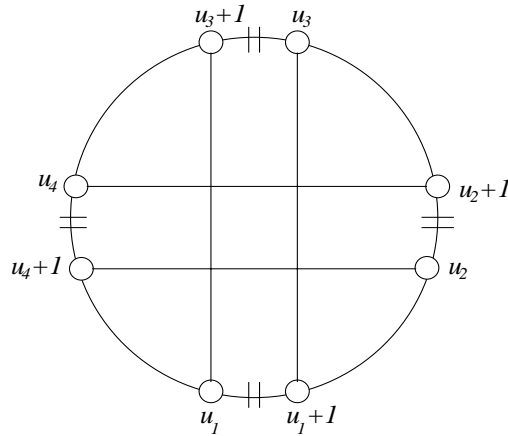


FIG. 2.12 – Opération 4-opt.

$n-2$, on perturbe aléatoirement la solution courante et on réitère. La solution finale est la meilleure solution trouvée au bout d'un certain nombre de perturbations. La perturbation qu'on a adoptée est l'opération 4-opt suivante : on choisit aléatoirement 4 sommets u_1, u_2, u_3 et u_4 , avec $u_1 < u_2 < u_3 < u_4$; on ajoute les arêtes $[u_1, u_3 + 1]$, $[u_1 + 1, u_3]$, $[u_2, u_4 + 1]$ et $[u_2 + 1, u_4]$ et on élimine les arêtes $[u_1, u_1 + 1]$, $[u_2, u_2 + 1]$, $[u_3, u_3 + 1]$ et $[u_4, u_4 + 1]$ (voir figure 2.12) sachant que le cycle hamiltonien est numéroté de 1 à n .

La proposition ci-dessous indique la complexité de la procédure *Génération_de_chaînes* ainsi que la taille du voisinage qu'elle explore.

Proposition 2.1.1 *Une itération de la procédure Génération_de_chaînes explore en $O(n^2)$ un voisinage dont la taille est en $O(n2^n)$.*

Preuve : Il est clair que la complexité de la procédure est en $O(n^2)$.

On calcule la taille du voisinage qu'on considère, à chaque itération de la procédure, en estimant d'abord le nombre de *CAO* paires, impaires et brisées considérées en calculant les plus courtes *CAO* de chaque type. Puis on montre que chaque *CAO* engendre au moins une solution différente de toutes les autres.

Soient P_i , I_i et B_i le nombre de *CAO* respectivement paires, impaires et brisées d'extrémités les sommets 1 et j , pour $j = 2, \dots, n$. On peut

facilement vérifier que $P_2 = P_3 = 0$, $I_2 = 0$, $I_3 = 1$, $B_j = 0$ pour $j = 2, 3, 4, 5$ et $B_6 = 1$.

Pour les *CAO* paires et impaires, les relations de récurrence définissant ces nombres à l'ordre n sont données par :

$$P_n = \sum_{j=2}^{n-1} I_j \text{ et } I_n = 1 + \sum_{j=2}^{n-1} P_j.$$

Soit $U_n = P_n + I_n$, pour tout $n \geq 2$. Le nombre total de *CAO* paires et impaires qui nous intéresse est égal à $U_{n+1} = \sum_{j=2}^n P_j + I_j$. Il suffit pour cela de calculer le terme général de la suite U_n en fonction de n uniquement. Après un calcul classique sur la suite, on trouve que

$$U_n = \begin{cases} 2^{n-3} & \text{si } n \geq 3 \\ 1 & \text{si } n = 3 \end{cases}$$

Pour les *CAO* brisées, la relation de récurrence définissant leur nombre à l'ordre n est donnée par :

$$B_n = \sum_{j=2}^{n-1} B_j + \sum_{j=2}^{n-3} U_j.$$

De même, après un calcul classique sur la suite, le terme général en fonction de n , uniquement, est donné par la relation

$$B_n = \begin{cases} (n-4)2^{n-7} & \text{si } n \geq 7 \\ 1 & \text{si } n = 6 \end{cases}$$

Finalement, le nombre total de chaînes alternées est de l'ordre de $B_{n+1} + U_{n+1} = (n+13)2^{n-6}$ ($n \geq 6$).

Par ailleurs, il est clair que deux *CAO* différentes ne peuvent pas engendrer une même structure en lasso fermé (car un ensemble d'arêtes ajoutées détermine d'une manière unique une *CAO*). Ce qui nous donne $(n+13)2^{n-6}$ structures en lasso fermé différentes.

On peut remarquer que l'arête $[1, 2]$ appartient à tous ces lassos fermés et en l'enlevant, on obtient autant de lassos, deux à deux distincts, dont les terminaisons sont toutes égales au sommet 2.

Pour faciliter la suite de la preuve, considérons uniquement les lassos engendrés par des *CAO* brisées.

Si le sommet j est la deuxième extrémité d'une *CAO* brisée, alors la racine du lasso engendré est j , et $j - 1$ est une sous-racine. Soit l'ensemble des solutions obtenues en enlevant l'arête $[j, j - 1]$, et en ajoutant l'arête $[j - 1, 2]$ à tous les lasso lorsque j représente la racine du lasso. La figure 2.13 montre une chaîne alternée brisée et la solution correspondante.

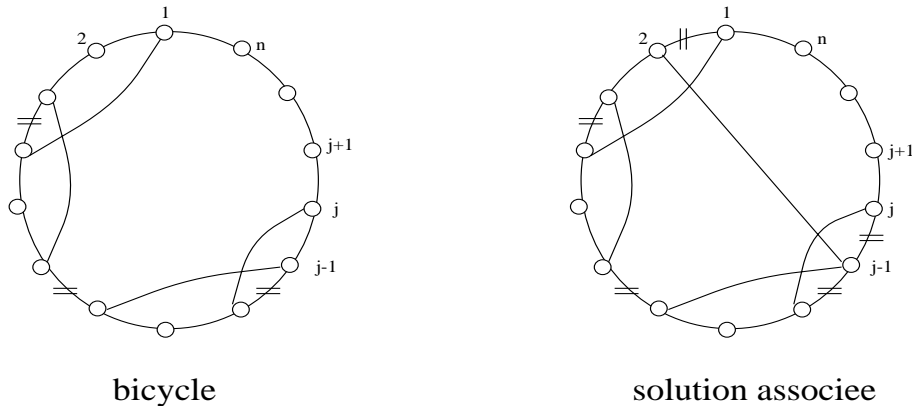


FIG. 2.13 – *Bicycle et solution associée.*

En procédant de la sorte, supposons que deux *CAO* brisées P_1 et P_2 différentes engendrent la même solution. Soient j_1 et j_2 les extrémités finales respectives de P_1 et P_2 . Le sommet 1 est l'extrémité initiale commune. Dans la solution, la chaîne, du sommet 1 au sommet j_i , est définie par la séquence de sommets $(1, n, n - 1, \dots, j_i + 2, j_i + 1, j_i)$, et le sommet j_i n'est pas adjacent au sommet $j_i - 1$, pour $i = 1, 2$. Alors les sommets j_1 et j_2 sont identiques.

On peut remarquer également que toutes les arêtes ajoutées de P_1 et P_2 appartiennent à la solution ; donc les deux *CAO* P_1 et P_2 sont définies par le même sous-ensemble d'arêtes ajoutées. Or, un ensemble d'arêtes ajoutées définit d'une manière unique une *CAO* brisée. Les *CAO* P_1 et P_2 sont identiques.

Finalement, le nombre de solutions, deux à deux différentes, considérées dans ce voisinage est au moins égal à $(n - 4) 2^{n-7}$ ($n \geq 7$). \square

On notera que, dans [35], Glover propose des procédures pour le *TSP* qui utilisent la structure en lasso fermé et montrent des résultats similaires au précédent pour ses procédures en utilisant des preuves différentes de celle

Problème	Opt	2-opt	3-opt	LK1	LK2	ILK	CAO
lin105	14379	8.42	0.00	0.77	0.00	0.00	0.00
pr107	44303	3.79	2.05	1.53	0.81	0.00	0.00
pr124	59030	2.68	1.15	2.54	0.39	0.00	0.50
pr136	96772	10.71	6.14	0.55	0.72	0.38	1.50
pr144	58537	3.79	0.39	0.56	0.06	0.00	0.00
pr152	73682	2.93	1.85	0.00	0.19	0.00	0.40
u159	42080	14.00	11.49	2.20	1.59	0.00	0.01
rat195	2323	6.46	3.01	1.55	1.55	0.47	1.40
d198	15780	3.85	6.12	0.63	1.51	0.16	0.70
pr226	80369	13.17	1.72	0.72	0.49	0.00	1.10
gil262	2378	10.26	3.07	1.18	2.44	0.55	2.00
pr264	49135	4.39	6.04	0.12	0.01	0.49	1.30
pr299	48194	10.46	4.37	1.55	1.36	0.15	1.20
lin318	42029	9.54	2.67	1.87	1.17	0.53	2.30
rd400	15281	5.01	3.42	2.34	1.41	0.75	2.10

TAB. 2.1 – Résultats numériques.

que nous donnons.

Dans la suite, on compare numériquement notre algorithme à quelques unes des méthodes classiques

2.1.2.3 Résultats numériques

On a choisi de confronter numériquement notre méthode avec les résultats d'une étude réalisée par Jünger et al. [56] sur 5 méthodes classiques. Nous avons retenu un jeu d'essais de 15 instances. Les colonnes 1 et 2 de la table 2.1 représentent, respectivement, le nom et la valeur de la solution optimale de l'instance. Dans cette étude, les méthodes retenues sont les méthodes 2-opt et 3-opt dans la classe des méthodes k -opt. On trouve également 2 variantes $LK1$ et $LK2$ de la méthode de Lin et Kernighan [64]; chacune d'elles est définie par des restrictions sur les mouvements à effectuer. Il ne nous semble pas important de les détailler ici, toutefois leurs définitions peuvent être trouvées dans [56]. Les résultats de ces 4 méthodes ont été obtenus sans perturbation et représentent les premiers minima locaux.

Problème	2-opt	3-opt	LK1	LK2	ILK	CAO
lin105	0.02	4.10	1.15	4.09	168.40	44.03
pr107	0.01	2.73	0.71	2.25	121.34	43.30
pr124	0.02	3.82	1.03	3.08	219.52	58.91
pr136	0.03	3.76	1.05	2.98	221.96	89.31
pr144	0.02	6.37	1.20	3.85	304.52	74.11
pr152	0.02	3.44	1.03	2.85	260.19	103.88
u159	0.02	6.03	1.46	4.26	314.53	104.27
rat195	0.02	4.41	1.93	4.86	409.74	173.22
d198	0.03	7.22	5.27	6.04	520.23	220.72
pr226	0.02	12.85	2.64	7.16	488.87	215.11
gil262	0.04	7.84	2.84	8.37	575.52	476.04
pr264	0.03	9.83	3.53	8.29	455.71	405.12
pr299	0.04	10.27	3.47	10.97	750.62	620.39
lin318	0.04	12.56	5.30	11.98	825.53	703.08
rd400	0.05	13.19	4.57	13.33	1153.41	1228.75

TAB. 2.2 – *Temps en CPU.*

La méthode *ILK* représente une variante de *LK2* avec un certain nombre de perturbations. On construit initialement une solution en utilisant une procédure du plus proche voisin, et on applique *LK2*. La solution obtenue est perturbée en utilisant un mouvement aléatoire de type 4-opt, et on réitère en appliquant la méthode *LK2*. Le nombre de perturbations choisi est égal à 20 et la solution retenue est la meilleure solution générée pendant les 20 itérations.

Dans cette comparaison, notre souci majeur est de présenter des résultats réalisés dans des conditions très proches. C'est ainsi qu'on a choisi d'utiliser une opération 4-opt pour perturber les solutions et cela durant 20 itérations.

Le tableau 2.1 donne la qualité des solutions obtenues, pour chaque instance, exprimée par l'écart de la solution obtenue par rapport à la solution optimale en pourcentage. Les colonnes 2-opt, 3-opt, *LK1*, *LK2* et *ILK* représentent les résultats obtenus pour ces 5 méthodes par Jünger et al. [56]. La dernière colonne représente les résultats que nous avons obtenus. On notera que nos résultats représentent des moyennes sur 30 exécutions. La table 2.2

donne les temps de calcul exprimés en secondes de *CPU* pour une station SUN SPARC 10 – 20 pour les résultats de Jünger et al. [56] et pour une station SUN SPARC 10 – 51 pour nos résultats.

La qualité moyenne des solutions que nous avons obtenu n'est pas très bonne. L'implantation de Jünger et al. [56] de la méthode *ILK* est très renforcée. Cela se manifeste dans les temps d'exécutions élevés de la méthode. On pourrait obtenir une implantation de notre méthode qui générera de meilleures solutions en un temps d'exécution plus élevé en lui ajoutant certaines procédures décrites dans Jünger et al. [56].

Dans la section suivante, on revient au *ITSP* pour proposer une décomposition du *ITSP* en sous-problèmes dépendants et l'algorithme de résolution qu'elle induit.

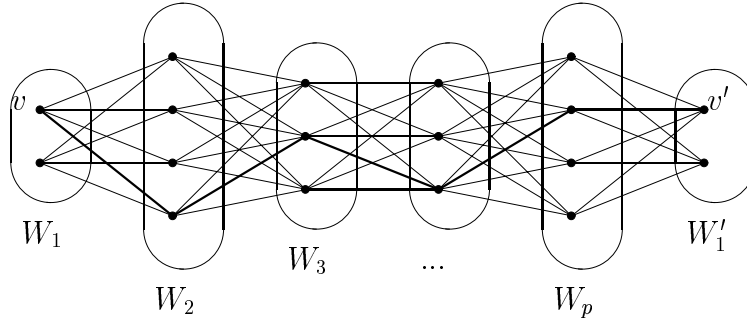
2.1.3 Algorithme de décomposition

L'étude de la structure du *ITSP* permet de développer un algorithme qui exploite au mieux la nature du *ITSP*. Si on fixe la séquence de parcours des pays, le *ITSP* se réduit à un problème de calcul de plus courte chaîne. Si on fixe les sommets qui représentent les pays dans la solution, le *ITSP* se réduit à un *TSP*. Le *ITSP* est ainsi l'association de deux problèmes dépendants. Le point suivant sur la complexité du *ITSP* permet de dégager un algorithme de résolution du *ITSP* qu'on appelle algorithme de décomposition.

2.1.3.1 Complexité du *ITSP*

Dans le cas général, le *ITSP* est un problème *NP*-difficile. Ici, on s'intéresse à la complexité du problème dans le cas où le nombre de pays est fixé.

Soit P une séquence donnée de pays, disons W_1, \dots, W_p . On définit un réseau $R = (V_R, E_R)$ associé à la séquence P de la manière suivante. L'ensemble des sommets est $V_R = W_1 \cup \dots \cup W_p \cup W'_1 = V \cup W'_1$ avec W'_1 une copie du pays W_1 . L'ensemble des arêtes est $E_R = E_1 \cup E_2$ avec $E_1 = \{[u, v] \in E : u \in W_i, v \in W_{i+1} \text{ et } i = 1, \dots, p-1\}$ et $E_2 =$

FIG. 2.14 – Réseau R .

$\{[u, v'] : u \in W_p, v' \in W'_1 \text{ et } [u, v] \in E\}$, avec $v' \in W'_1$ la copie du sommet v de W_1 (voir figure 2.14).

En utilisant le réseau R , ci-dessus défini, on montre, dans ce qui suit, que la complexité du *ITSP* est polynomiale dans le cas où le nombre de pays est fixé. On notera que ce résultat est connu dans la littérature.

Théorème 2.1.1 *Le ITSP est polynomial en nombre de villes n pour un nombre de pays fixé. Sa complexité est $O(n^3)$.*

Preuve : Soient P une séquence de pays quelconque et R le réseau associé. Sans perte de généralité, on peut supposer que $P = (W_1, \dots, W_p)$. Soit P_v la plus courte chaîne reliant $v \in W_1$ à $v' \in W'_1$, avec v' la copie du sommet v . La plus courte chaîne parmi toutes les chaînes $P_v, v \in W_1$, donne le tour optimal correspondant à la séquence P de pays. Pour obtenir le tour optimum, il suffit de considérer toutes les séquences de pays possibles.

La complexité de cet algorithme est $O((p-1)!|W_1|n^2)$. Pratiquement, on choisit W_1 comme étant le pays qui a le moins de sommets pour minimiser cette complexité. \square

2.1.3.2 Algorithme

La preuve précédente suggère l'algorithme approximatif de décomposition suivant :

Procédure Décomposition;
Début

```

Initialisation;
Tant que (critère d'arrêt non vérifié) faire
  Début
    Recherche d'une plus courte chaîne;
    Résolution d'un TSP;
  Fin
Fin.

```

Dans le module “Initialisation”, on détermine un réseau initial R à partir d’une séquence initiale de pays. Cependant, il faut savoir que les deux modules qui suivent dans la procédure peuvent apparaître dans un ordre inversé, et dans ce cas, un ensemble légal L de sommets est donné en initialisation. Dans le module “Recherche d’une plus courte chaîne”, on intègre une procédure de recherche de plus courte chaîne adaptée au réseau R . On calcule la plus courte chaîne parmi toutes les chaînes reliant $v \in W_1$ à $v' \in W'_1$, avec v' la copie du sommet v dans W_1 le premier pays de la séquence de pays associée au réseau R . En sortie, on récupère un ensemble légal L de sommets. Enfin, dans le module “Résolution de *TSP*”, on intègre une procédure de résolution de *TSP* sur l’ensemble légal L des sommets. En sortie, on récupère un réseau R . L’algorithme s’arrête lorsque aucune amélioration n’est constatée dans le module “Résolution de *TSP*”, ou lorsqu’on atteint un nombre maximum d’itérations.

On pourrait penser que cet algorithme est une réduction polynomiale du *ITSP* au *TSP* et donc, un algorithme exact pour la résolution du *ITSP*. Ceci n’est pas vrai ; même si on dispose d’un algorithme polynomial pour la résolution du *TSP*, l’algorithme précédent ne donne pas toujours une tournée internationale de longueur minimum.

La figure 2.15 représente un graphe international à 5 pays et 6 sommets sur lequel l’algorithme de décomposition “échoue” si on l’initialise avec la tournée T indiquée sur la figure. En effet, dans ce cas, la solution générée par l’algorithme est la tournée internationale T , alors que l’optimum est atteint dans la tournée internationale T^* (dans cet exemple les distances sont euclidiennes et le graphe est dessiné sur le plan).

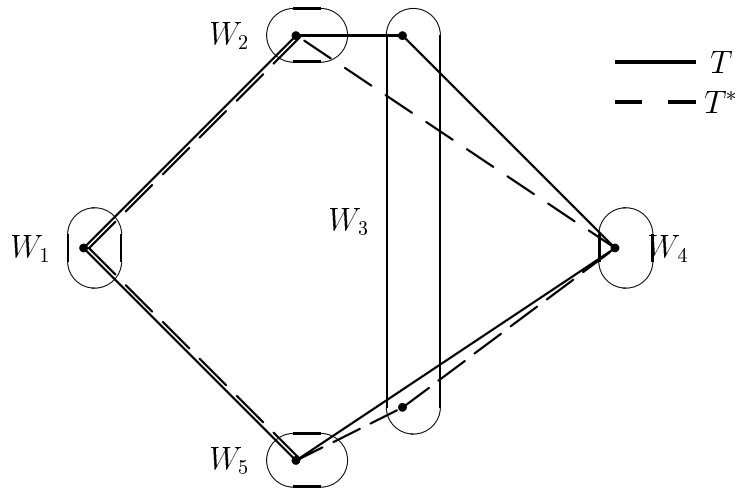


FIG. 2.15 – *Un contre-exemple de l’algorithme de décomposition.*

Bien que l’algorithme de décomposition du *ITSP* soit approximatif, il reste très intéressant dans la mesure où il est simple, structuré et facile à implanter, mais aussi parce qu’il exploite au mieux la structure du *ITSP*.

On peut implémenté cet algorithme en intégrant dans le module “Résolution de *TSP*” une procédure quelconque pour le *TSP*. Pour notre part, on a utilisé la procédure décrite dans 2.1.2.2. Pour la recherche d’une plus courte chaîne dans le réseau R , on utilise simplement l’algorithme de Bellman [9]. Le réseau R est sans circuit et sa décomposition en stables est évidente. La complexité de cette recherche est en $O(|W_1|n^2)$, avec W_1 le premier pays dans la séquence de pays associée au réseau R .

2.2 Algorithmes exacts

L’objectif des méthodes exactes est de trouver, en un temps de calcul, le plus court possible, une solution garantie optimale du problème considéré. Pour tous les problèmes *NP*-difficiles de l’optimisation combinatoire, ceci reste un défi souvent difficile à relever.

Le principe général des méthodes exactes pour la résolution des problèmes NP -difficiles de l'optimisation combinatoire consiste à énumérer implicitement, et le plus efficacement possible, toutes les solutions du problème pour en extraire une solution optimale.

Dans la suite, on présentera une réduction polynomiale du $ITSP$ au TSP qui permet d'envisager l'utilisation d'une méthode exacte pour la résolution du TSP , via la réduction, pour obtenir ainsi un premier algorithme exact pour la résolution du $ITSP$. On montrera les principaux inconvénients d'une telle approche. Dans la section 2.2.3, on présente la méthode classique de branchement et d'évaluation dans le cas général, et on cite les différentes tentatives de résolution du $ITSP$ par cette même méthode. Une extension très efficace de cette méthode, appelée méthode de branchement et de coupe ("branch and cut"), utilise la programmation linéaire combinée à une certaine connaissance polyédrale du problème. On donnera une description détaillée du principal module de cette méthode et on montrera la nécessité d'une étude polyédrale pour sa mise en œuvre. Fischetti et al. [28] sont les seuls à avoir proposé un algorithme de ce type pour la résolution du $ITSP$ en utilisant une première formulation du problème en nombres entiers. On finira cette section en proposant une nouvelle formulation du problème en nombres entiers qui nous permettra de développer un nouvel algorithme de branchement et de coupe dans le chapitre 4.

Avant de présenter toutes ces approches, on donne quelques notations utilisées dans la suite de ce chapitre.

2.2.1 Quelques notations

Soit $\mathbb{R}^{|E|}$ l'espace de tous les vecteurs réels dont les composantes sont indexées par l'ensemble E des arêtes. Pour tout vecteur $x \in \mathbb{R}^{|E|}$, on note x_e , ou $x[u, v]$, la composante du vecteur x indexée par l'arête $e = [u, v]$, d'extrémités u et v . Pour un sous-ensemble F d'arêtes, on note $x(F)$ la somme des composantes de x indexées par les éléments de F . De la même manière, soit $\mathbb{R}^{|V|}$ l'espace de tous les vecteurs réels dont les composantes sont indexées par l'ensemble V des sommets. Pour tout vecteur $y \in \mathbb{R}^{|V|}$, on note y_v la composante du vecteur y indexée par le sommet v . Pour un sous-ensemble S de sommets, on note $y(S)$ la somme des composantes de y

indexées par les éléments de S .

Pour un sous-ensemble de sommets S , on note $\delta(S)$ l'ensemble des arêtes ayant exactement une extrémité dans S et $E(S)$ l'ensemble des arêtes ayant les deux extrémités dans S .

2.2.2 Approche par réduction au TSP

L'idée de résoudre le *ITSP* en utilisant une réduction polynomiale au *TSP* est naturelle et semble, à priori, très intéressante. Notre première tentative de résolution du *ITSP* par un algorithme exact consistait à réduire le *ITSP* au *TSP*, et à résoudre ce dernier en utilisant une approche polyédrale (voir section 2.2.4). D'une manière générale, on peut utiliser l'une des nombreuses méthodes de résolution du *TSP*.

2.2.2.1 Description de la réduction

La réduction consiste à construire, à partir du graphe international G , un nouveau graphe G_R muni d'une nouvelle fonction coût sur ses arêtes, tel que le *ITSP* sur le graphe G soit équivalent au *TSP* sur le graphe G_R . On définit le graphe $G_R = (V_R, E_R)$ de la façon suivante :

- a) V_R : à un sommet u de V appartenant à un pays non dégénéré, on associe deux sommets u_a et u_b de V_R . On dira que (u_a, u_b) est un couple de sommets jumeaux. On notera ce sous-ensemble de sommets par V'_R . Si \mathcal{W}^d est l'ensemble des pays dégénérés, alors $V_R = V'_R \cup \mathcal{W}^d$. Sa taille est égale à $2n - d$, avec $d = |\mathcal{W}^d|$. On définit également une partition \mathcal{W}_R de V_R en p sous-ensembles, telle qu'un élément W_{R_i} de \mathcal{W}_R est le sous-ensemble de sommets de V_R induit par les sommets de V qui définissent W_i , pour $i = 1, \dots, p$;
- b) E_R : à une arête $e = [u, v]$ de E , reliant les deux sommets u et v de V , on associe : les arêtes $[u_a, v_a]$, $[u_a, v_b]$, $[u_b, v_a]$ et $[u_b, v_b]$ si u et v appartiennent à deux pays non dégénérés, ou les arêtes $[u, v_a]$ et $[u, v_b]$ si u appartient à un pays dégénéré, et v à un pays non dégénéré, ou l'arête $[u, v]$ si les deux sommets u et v appartiennent à deux pays dégénérés. L'ensemble d'arêtes ainsi obtenu est noté E'_R . Des arêtes de E'_R seront dites jumelles si elles sont associées à une même arête

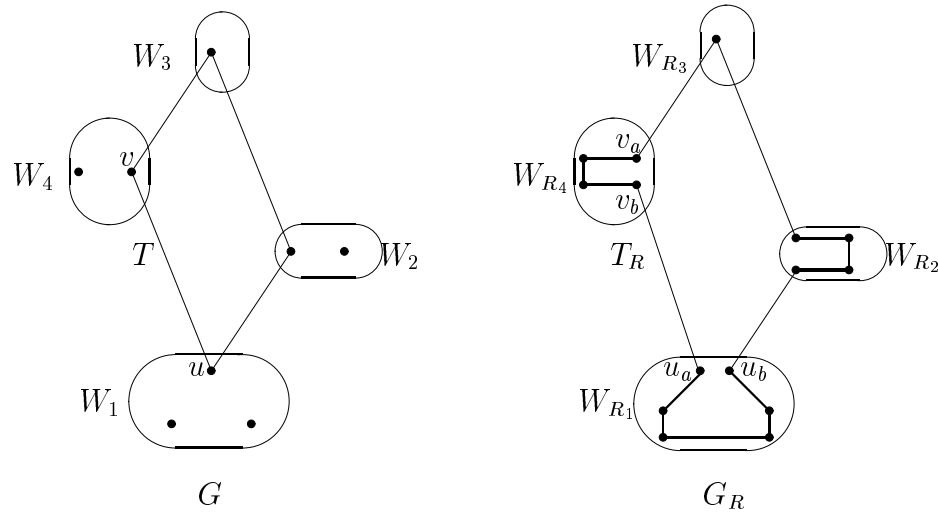


FIG. 2.16 – Graphes de la réduction.

de E . On associe à chaque élément W_{R_i} de \mathcal{W}_R un cycle élémentaire quelconque C_i , tel que chaque paire de sommets jumeaux représente une arête du cycle; C_i est alors l'union de deux ensembles F_i et L_i , avec F_i l'ensemble des arêtes reliant les paires de sommets jumeaux. Pour compléter F_i et obtenir un cycle, on peut choisir L_i de plusieurs manières différentes; le cycle C_i n'est pas unique. Soit l'ensemble $E''_R = \cup_{i=1}^p C_i$ (avec $C_i = \emptyset$ pour un pays dégénéré). L'ensemble E_R est égal à l'union des ensembles E'_R et E''_R , sa taille est égale à $4m - 3\frac{d(d-1)}{2} - 2d(d-1) + 2(n-d) = 4m - (d-1)(2n-d/2)$. La figure 2.16 représente un couple de Graphes G et G_R . Les arêtes qui y figurent représentent une tournée internationale dans G et le cycle hamiltonien correspondant dans G_R .

Soient la fonctions coût $c_R : E_R \rightarrow R^+$ qui associe à chaque arête $e = [u, v]$ de E_R le coût $c_R(e)$ égal à la distance séparant les sommets de G associés à

u et v , et la fonction coût $c'_R : E_R \rightarrow R^+$ telle que :

$$c'_R(e) = \begin{cases} c_R(e) & \text{si } e \in E'_R, \\ M & \text{si } e \in F_i, i = 1, \dots, p, \\ 2M & \text{si } e \in C_i \setminus F_i, i = 1, \dots, p, \end{cases} \quad (2.1)$$

avec M un nombre suffisamment grand. On notera un graphe G muni d'une fonction coût p par le couple (G, p) .

Un cycle hamiltonien H^* de longueur minimum dans (G_R, c'_R) vérifie les propriétés décrites dans les deux observations suivantes.

Observation 2.2.1 *Au plus une arête parmi un ensemble d'arêtes jumelles est utilisée dans H^* . Dans le cas contraire, l'ensemble des arêtes de H^* , ayant les deux extrémités dans l'ensemble de sommets formé par l'union des deux pays associés à ces arêtes jumelles, est un cycle.*

Observation 2.2.2 *Le nombre d'arêtes de H^* incidentes à un pays donné est exactement égal à 2, et de plus elles sont incidentes à deux sommets u_a et u_b formant un couple de sommets jumeaux. Sinon, on peut facilement montrer que H^* n'est pas optimum.*

Le théorème suivant établit le principal résultat de la réduction ; sa preuve utilise les deux observations précédentes.

Théorème 2.2.1 *Le ITSP sur G est équivalent au TSP sur G_R muni de la fonction coût c'_R .*

Preuve : Les deux observations précédentes établissent une bijection entre les tournées internationales de G et les cycles hamiltoniens de G_R qui les vérifient. De plus, l'image par cette bijection d'un optimum H^* du TSP sur (G_R, c'_R) est une tournée internationale optimale du ITSP, et réciproquement. \square

Ainsi, la résolution du TSP, défini sur le graphe G_R muni de la fonction c'_R , donne la solution optimale du ITSP défini sur le graphe G . Dans la suite, on étudie la résolution du TSP sur (G_R, c'_R) par un algorithme de branchement et de coupe et on discutera les limites de cette approche.

Le lecteur non familiarisé avec l'algorithme de branchement et de coupe, et la notion de polyèdres est vivement conseillé de sauter la section suivante, et de reprendre la lecture à partir de la section 2.2.3, jusqu'à la fin du chapitre, où les éléments de base d'un algorithme de branchement et de coupe sont développés, et où l'on peut trouver les différentes définitions et notations concernant les polyèdres. On pourra ensuite revenir à la section en question sans aucune difficulté.

2.2.2.2 Approche polyédrale via la réduction

Avant de développer cette approche polyédrale, on montre que le choix de la fonction coût c'_R a une interprétation polyédrale. En effet, la fonction coût c'_R impose que l'optimum appartienne toujours à une face particulière de l'enveloppe convexe $TSP(G_R)$ des cycles hamiltoniens dans G_R .

Cette face que l'on désignera par $CTSP(G_R)$ est définie par :

$$CTSP(G_R) = TSP(G_R) \cap P(W_R) \cap P(F), \quad (2.2)$$

avec

$$P(F) = \{x \in R^{|E|} : x_e = 1, e \in F = \cup_{i=1}^p F_i\} \quad (2.3)$$

et

$$P(W_R) = \{x \in R^{|E|} : x(E(W_{R_i})) = |W_{R_i}| - 1, W_{R_i} \in \mathcal{W}_R\}. \quad (2.4)$$

Il faut remarquer qu'on obtient la face $CTSP(G_R)$ en serrant des inéquations qui induisent des facettes pour le polytope $TSP(G_R)$; les sommets de $CTSP(G_R)$ sont exactement les cycles hamiltoniens qui vérifient les propriétés décrites dans les deux observations 2.2.1 et 2.2.2.

Soit $CTSP$ le problème d'optimisation sur $CTSP(G_R)$ si on munit le graphe G_R de la fonction coût c_R . Le TSP sur (G_R, c'_R) est ainsi équivalent au $CTSP$ sur (G_R, c_R) .

Dans la suite, on considère le $CTSP$ et on munit le graphe G_R de la fonction c_R . La fonction c_R représente une distance euclidienne si les coordonnées de 2 sommets jumeaux sont égales aux coordonnées du sommet de G qui les a induits; contrairement à la fonction c'_R qui ne représente pas une distance.

De plus, en considérant la face $CTSP(G_R)$, l'ensemble des solutions potentielles est ainsi clairement défini sur le plan polyédral.

Pour résoudre le $CTSP$ en utilisant un algorithme de branchement et de coupe, il suffit alors de considérer un algorithme de branchement et de coupe pour la résolution du TSP , et de rajouter au programme linéaire initial les équations exprimées dans $P(\mathcal{W}_R)$ et $P(F)$. La disponibilité d'un code de branchement et de coupe pour la résolution du TSP [17] nous a permis d'obtenir rapidement un code de branchement et de coupe pour la résolution du $CTSP$.

Après avoir réalisé cette adaptation, on s'est aperçu, après expérience, que cette manière d'approcher le problème est inefficace. Il est vrai que la taille du problème augmente sensiblement. Mais on pense que, dans ce cas là, cet argument ne détermine pas à lui seul l'inefficacité de cette approche. En effet, des difficultés autres se sont mêlées à notre tentative, et plus particulièrement les suivantes :

- a) **Coupes non profondes** : une description linéaire de $CTSP(G_R)$ est obtenue à partir de celle de $TSP(G_R)$ en ajoutant les équations exprimées dans $P(\mathcal{W}_R)$ et $P(F)$.

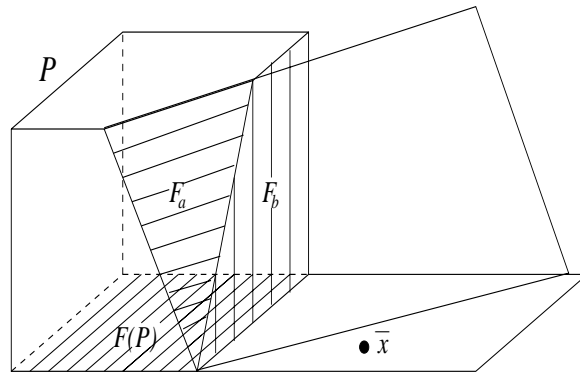


FIG. 2.17 –

Cependant, une description linéaire minimale de $TSP(G_R)$ ne donne pas de cette manière une description linéaire minimale de $CTSP(G_R)$. Une inéquation induisant une facette de $TSP(G_R)$ n'induit pas nécessairement une facette de $CTSP(G_R)$. L'exemple de la figure 2.17

illustre clairement ce point : considérons le polytope P (le cube coupé) dans l'espace R^3 , et $F(P)$ la face du bas de P . Soient $ax \leq a_0$ et $bx \leq b_0$ les inéquations de R^3 qui induisent respectivement les facettes F_a et F_b de P . Sur la figure 2.17, on peut voir que, dans R^3 , l'inéquation $ax \leq a_0$ n'induit pas une facette de $F(P)$, et que le point \bar{x} qui appartient au plan induisant $F(P)$ est tel que $a\bar{x} > a_0$ et $b\bar{x} > b_0$. En effet, pour couper le point \bar{x} dans un algorithme de coupe pour l'optimisation sur la face $F(P)$, on risque de générer la contrainte $ax \leq a_0$, au lieu de $bx \leq b_0$, si on dispose d'une description minimale du polytope P seulement.

Il n'est pas facile de trouver un exemple semblable sur le polytope $TSP(G_R)$. La description linéaire connue de $TSP(G_R)$ n'est que partielle, et les inéquations qui la constituent sont souvent difficiles à manipuler. Cependant, on peut trouver, parmi les facettes connues de $TSP(G_R)$, des facettes qui sont dominées par d'autres quand on considère une face de $TSP(G_R)$. Par exemple, si on fixe toutes les arêtes d'une coupe à zéro, sauf trois arêtes e_1 , e_2 et e_3 , alors $x_{e_1} \geq 0$ est dominée par $x_{e_1} + x_{e_2} \geq 1$ et $x_{e_1} + x_{e_3} \geq 1$.

- b) **Fonction objective particulière :** la fonction coût c_R à optimiser est très particulière. Des arêtes jumelles ont des coûts identiques et certaines arêtes ont des coûts nuls. Ceci crée une dégénérescence dans la résolution des programmes linéaires.

En conclusion, cette réduction nous a permis de montrer une certaine relation qui existe entre le $ITSP$ et le TSP , malheureusement son utilisation dans une approche de résolution du $ITSP$ n'est pas efficace. Dans la suite, on ne s'intéressera qu'à des méthodes qui approchent directement le $ITSP$.

2.2.3 Méthode de branchement et d'évaluation

La méthode de branchement et d'évaluation, connue sous le nom de "branch and bound", a été pendant longtemps la méthode la plus utilisée pour la résolution à l'optimum des problèmes NP -difficiles.

Le principe de base de cette méthode consiste, dans une première phase,

à partitionner l'ensemble des solutions S_0 en plusieurs sous-ensembles. Cette première phase est appelée phase de branchement et les critères sur lesquels est basée la subdivision de l'ensemble des solutions sont dits stratégies de branchement. La deuxième phase consiste à évaluer chaque sous-ensemble de solutions : on calcule, pour chaque sous-ensemble, une borne inférieure de la valeur de sa meilleure solution. Cette deuxième phase est appelée phase d'évaluation.

En appliquant itérativement ces deux opérations de branchement et d'évaluation, on construit un arbre, dit arbre d'énumération. Si, à chaque itération, on subdivise un ensemble de solutions en 2 sous-ensembles, l'arbre d'énumération sera dit binaire. L'ensemble S_0 des solutions du problème est la racine de l'arbre, et les sous-ensembles S_i des solutions sont les nœuds de l'arbre. Un nœud S_i quelconque est adjacent aux nœuds qui représentent sa subdivision, appelés fils de S_i , et au nœud dont S_i est élément de sa subdivision, appelé père de S_i . La racine S_0 n'admet pas de père (voir figure 2.18).

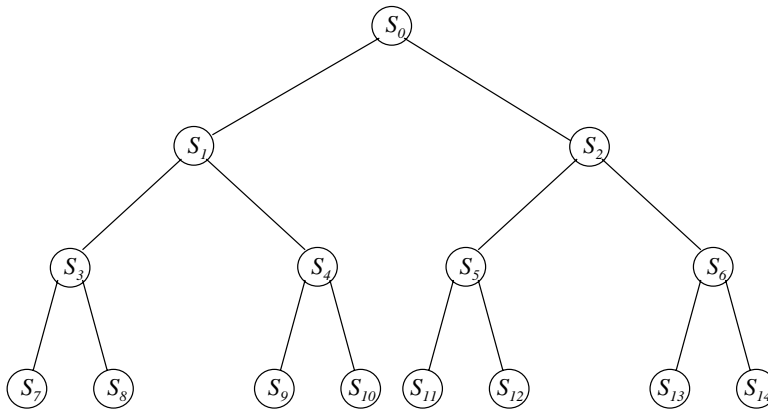


FIG. 2.18 – Arbre d'énumération binaire.

La méthode consiste à construire un arbre d'énumération tel que l'on puisse affirmer, avec preuve, que l'optimum appartient ou n'appartient pas à une feuille quelconque de l'arbre d'énumération.

Le problème majeur qui se pose en utilisant ce type de méthodes est la taille de l'arbre d'énumération qui augmente d'une manière exponentielle. On

voudrait alors affirmer, chaque fois qu'il est possible, que la solution optimale du problème n'appartient pas à une branche donnée de l'arbre d'énumération; ce qui nous permettra d'abandonner la recherche dans cette branche.

Si pour un nœud quelconque S_i , la borne inférieure de la meilleure solution dans S_i est strictement supérieure à la valeur de la solution optimale du problème, alors celle-ci n'appartient pas au sous-ensemble S_i . On abandonne la recherche dans la branche définie par le nœud S_i et on dira que la branche est stérile. Cela suppose connaître la solution optimale du problème considéré. Or, dans la pratique, l'optimum est souvent inconnu et représente l'objectif à atteindre. Cependant, le raisonnement précédent reste valide si on considère une borne supérieure de la solution optimale du problème au lieu de la valeur de la solution optimale elle-même.

Finalement, la taille de l'arbre d'énumération, et par conséquent l'efficacité d'un algorithme de branchement et d'évaluation, dépend en grande partie de la valeur de la borne supérieure et des valeurs des bornes inférieures dans chaque nœud.

Une borne supérieure est déterminée initialement (au niveau de la racine S_0) en utilisant une procédure de recherche locale et, par la suite, peut être améliorée dans l'arbre d'énumération. La section 2.1 présente toute une collection d'algorithmes pour le calcul d'une borne supérieure pour le *ITSP*. En général, ce calcul est une opération classique et ne présente pas de difficultés majeures. Pour la plupart des problèmes d'optimisation combinatoire *NP*-difficiles, on dispose d'algorithmes efficaces pour la détermination d'une borne supérieure.

Une borne inférieure de l'optimum, quant à elle, est calculée en optimisant la même fonction économique sur un espace de solutions plus grand qu'on appelle une relaxation. La valeur de la solution obtenue est au plus égale à l'optimum; de ce fait, elle constitue une borne inférieure de l'optimum du problème initial. Le problème associé doit être facile pour pouvoir calculer efficacement la borne inférieure. D'une manière générale, on distingue deux types de relaxations: les relaxations discrètes et les relaxations continues.

Une relaxation discrète consiste à relâcher certaines propriétés combinatoires que vérifient les solutions du problème et considérer ainsi un ensemble d'objets combinatoires plus grand, contenant l'ensemble des solutions initiales.

Exemple 2.2.1 *Un 2-pays-couplage parfait est un ensemble d'arêtes tel que le degré de chaque pays est égal à 2. Le problème d'optimisation sur l'ensemble des 2-pays-couplages parfaits est une relaxation discrète du ITSP, puisque une tournée internationale est un 2-pays-couplage parfait (la réciproque est évidemment fausse). Le problème est facile. En effet, seule l'arête de plus petit poids, parmi celles qui séparent deux pays quelconques, peut appartenir à un 2-pays-couplage parfait de poids minimum. De telles arêtes sont dites potentielles. Si on élimine les arêtes non potentielles et on contracte chaque pays en un seul sommet, le problème classique du 2-couplage parfait de poids minimum, dans le graphe obtenu, est équivalent au problème du 2-pays-couplage parfait de poids minimum. Puisque le problème de 2-couplage parfait de poids minimum est polynomial dans un graphe quelconque [26], on peut calculer une borne inférieure de l'optimum du ITSP en utilisant la relaxation de 2-pays-couplage parfait.*

Une relaxation continue consiste à considérer un programme linéaire P_r avec un nombre polynomial de contraintes, tel que toute solution du problème initial est solution réalisable de P_r . La programmation linéaire étant polynomiale en nombres de contraintes et de variables ; le calcul de l'optimum de P_r est polynomial. Pratiquement, pour résoudre rapidement le programme linéaire P_r , il faut qu'il y ait un nombre réduit de contraintes.

Exemple 2.2.2 *Si on associe, à chaque tournée internationale T , un vecteur d'incidence $x^T \in \mathbb{R}^E$, tel que $x_e = 1$ si $e \in T$, et $x_e = 0$ si $e \notin T$, alors tout vecteur d'incidence x^T d'une tournée internationale T est solution réalisable du programme linéaire*

$$P_r = \begin{cases} \text{Min} \sum_{e \in E} c_e x_e \\ \text{t.q.} \\ x(\delta(W_i)) = 2, \forall i \in \{1, \dots, p\} & (1) \\ 0 \leq x_e \leq 1, \forall e \in E & (2) \end{cases}$$

par conséquent la solution optimale de P_r est une borne inférieure de la solution optimale du *ITSP*. Les contraintes (1) représentent les contraintes de degré sur les pays.

Dans ce dernier cas, le problème qui peut se poser est de considérer un programme linéaire avec un nombre exponentiel de contraintes. Un temps exponentiel serait nécessaire pour la saisie et la résolution du problème d'une part, et d'autre part un espace mémoire exponentiel serait également nécessaire pour mémoriser toutes les contraintes. La méthode classique qui consiste à résoudre le problème dans sa totalité en utilisant un algorithme de la programmation linéaire (algorithme du simplexe, méthode ellipsoïdale,...) est inefficace ; on résout ce type de programme linéaire en utilisant la méthode de coupe qui est présentée dans la section 2.2.4.2. Un algorithme de branchement et d'évaluation qui utilise une relaxation continue dont la solution est donnée par une méthode de coupe est dit algorithme de branchement et de coupe.

Les algorithmes de branchement et d'évaluation pour la résolution du *ITSP* sont peu nombreux. La seule étude significative portant sur la résolution du *ITSP* par un algorithme de branchement et d'évaluation a été publiée en 1983 par Laporte et Nobert [61]. Les résultats de cette étude sont très limités. En effet, la taille des problèmes résolus à l'optimum n'excède guère 50 sommets. D'une manière générale, et pour la plupart des problèmes d'optimisation combinatoire *NP*-difficiles, la méthode de branchement et d'évaluation devient inefficace dès que la taille des instances est de l'ordre de quelques dizaines.

Pour la résolution des instances de grandes tailles, les algorithmes de branchement et de coupe s'avèrent très efficaces. Si on considère l'exemple typique du *TSP*, la taille des instances résolues à l'optimum par un algorithme de branchement et d'évaluation n'excède pas quelques dizaines de sommets, tandis qu'en utilisant un algorithme de branchement et de coupe, des instances de quelques milliers de sommets ont été résolues à l'optimum [78, 44, 2]. Dans la section suivante, on décrit la procédure de coupe, dans le cas général, comme module de base d'un algorithme de branchement et de coupe. On montre par la même occasion la nécessité d'une étude polyédrale pour la

mise en œuvre d'un tel algorithme : l'approche est dite polyédrale.

2.2.4 Approche polyédrale

Le développement d'un algorithme de coupe fait appel à différentes notions de la théorie des polyèdres, que nous utilisons également dans les chapitres suivants. Dans [5, 50, 83, 91], on trouve les fondements et les développements de cette théorie et dans [87], Schrijver donne les éléments de la théorie des polyèdres nécessaires dans l'étude des problèmes d'optimisation combinatoire. Dans ce qui suit, on donnera les définitions des principales notions ainsi que les résultats connus qui nous intéressent.

2.2.4.1 Notions polyédrales de base

Dans ce qui suit S désigne un ensemble non vide de k éléments x_1, \dots, x_k de \mathbb{R}^n . La plupart des définitions sont suivies par les propriétés fondamentales des notions introduites.

Indépendance affine : *l'ensemble S est dit affinement indépendant si les relations $\sum_{i=1}^k \lambda_i x_i = 0$ et $\sum_{i=1}^k \lambda_i = 0$ impliquent $\lambda_i = 0$, pour tout $i = 1, \dots, k$, sinon S est dit affinement dépendant.*

Rang affine : *le rang affine d'un ensemble S est le nombre maximum d'éléments de S affinement indépendants.*

Indépendance linéaire : *l'ensemble S est dit linéairement indépendant si la relation $\sum_{i=1}^k \lambda_i x_i = 0$ implique $\lambda_i = 0$, pour tout $i = 1, \dots, k$, sinon S est dit linéairement dépendant.*

Rang : *le rang d'un ensemble S est le nombre maximum d'éléments de S linéairement indépendants.*

Enveloppe affine : *l'enveloppe affine de S , notée $\text{aff}(S)$, est l'ensemble*

$$\left\{ \sum_{i=1}^k \lambda_i x_i : \sum_{i=1}^k \lambda_i = 1 \text{ et } \lambda_i \in \mathbb{R}, \text{ pour } i = 1, \dots, k \right\}.$$

Enveloppe conique : *l'enveloppe conique de S , notée cône (S), est l'ensemble $\left\{ \sum_{i=1}^k \lambda_i x_i : \lambda_i \in \mathbb{R}^+, \text{ pour } i = 1, \dots, k \right\}$. Le plus souvent, on utilise le terme cône pour désigner l'enveloppe conique.*

Enveloppe convexe : *l'enveloppe convexe de S , notée conv (S), est l'ensemble $\left\{ \sum_{i=1}^k \lambda_i x_i : \sum_{i=1}^k \lambda_i = 1 \text{ et } \lambda_i \in \mathbb{R}^+, \text{ pour } i = 1, \dots, k \right\}$.*

Enveloppe : *une enveloppe P est définie par $P = Q + C$, avec Q une enveloppe convexe de \mathbb{R}^n , et C une enveloppe conique de \mathbb{R}^n ; i.e. $P = \{x + y : x \in Q \text{ et } y \in C\}$.*

Dimension : *la dimension d'un ensemble P de \mathbb{R}^n , notée $\dim (P)$, est le nombre maximum d'éléments de P affinement indépendants diminué d'une unité. Si $\dim (P) = n$, alors l'ensemble P est dit de pleine dimension. Si $\dim (P) \leq n - 1$ et le vecteur 0 n'appartient pas à un hyperplan contenant P , alors l'indépendance linéaire est équivalente à l'indépendance affine.*

Demi-espace : *un demi-espace est un ensemble $D = \{x \in \mathbb{R}^n : a^t x \leq b\}$ avec a un vecteur de \mathbb{R}^n et b un réel. Si a est le vecteur nul et $b \geq 0$, alors $D = \mathbb{R}^n$.*

Hyperplan : *un hyperplan est un ensemble $H = \{x \in \mathbb{R}^n : a^t x = b\}$ avec a un vecteur de \mathbb{R}^n non nul et b un réel.*

Polyèdre : *un polyèdre est un ensemble $P = \{x \in \mathbb{R}^n : a_i^t x \leq b_i, i \in I\}$ avec a_i un vecteur de \mathbb{R}^n et b_i un réel, pour tout i appartenant à un ensemble I fini. Un polyèdre est l'intersection d'un nombre fini de demi-espaces.*

Polytope : *un polytope P est un polyèdre borné; i.e. $\exists B \in \mathbb{R}^+$, tel que $\forall x \in P, \|x\| \leq B$, avec $\|\cdot\|$ une norme quelconque de \mathbb{R}^n (toutes les normes sont équivalentes dans \mathbb{R}^n).*

Inéquation valide : une inéquation $f^t x \leq f^0$ est dite valide pour un polyèdre P , si pour tout \bar{x} élément de P , on a $f^t \bar{x} \leq f^0$. De plus, elle est dite support de P s'il existe x^* , élément de P , tel que $f^t x^* = f^0$.

Face : une face F d'un polyèdre P est l'intersection du polyèdre P avec l'hyperplan défini par une inéquation valide $f^t x \leq f^0$ de P ; i.e. $F = P \cap \{x \in \mathbb{R}^n : f^t x = f^0\}$. On dit alors que la face F est induite par l'inéquation $f^t x \leq f^0$. Si f est le vecteur nul de \mathbb{R}^n , alors la face $F = P$. Dans ce cas F est dite non propre. Si $f^t x \leq f^0$ n'est pas support de P , alors la face $F = \emptyset$.

Si F est une face non vide et $P = \{x \in \mathbb{R}^n : a_i^t x \leq b_i, i \in I\}$, alors il existe $I' \subset I$ tel que $F = \{x \in P : a_i^t x = b_i, i \in I'\}$.

Sommet : un sommet d'un polyèdre P est une face de dimension nulle; donc une face réduite à un seul sommet. Un point x^* de P est un sommet de P , si et seulement si, il existe $I' \subset I$ tel que x^* soit la solution unique du système d'équations $a_i^t x = b_i$, pour tout $i \in I'$.

Facette : une facette d'un polyèdre P est une face maximale par inclusion. La dimension d'une facette est égale à $\dim(P) - 1$.

Inéquations équivalentes : deux inéquations $f_1^t x \leq f_1^0$ et $f_2^t x \leq f_2^0$ valides pour un polyèdre P , sont dites équivalentes, si elles induisent la même face de P .

Inéquations dominées : on dit qu'une inéquation $f_1^t x \leq f_1^0$, valide pour P , induisant une face F_1 de P , domine une inéquation $f_2^t x \leq f_2^0$, également valide pour P , et induisant une face F_2 de P , si F_2 est strictement incluse dans F_1 . Lorsque les deux inéquations $f_1^t x \leq f_1^0$ et $f_2^t x \leq f_2^0$ apparaissent dans la description linéaire du polyèdre P , on peut omettre l'inéquation $f_2^t x \leq f_2^0$ dans la description de P . L'inéquation $f_2^t x \leq f_2^0$ est dite redondante.

Description minimale d'un polyèdre : une description linéaire d'un polyèdre est dite minimale si aucune des inéquations de la description n'est redondante.

Toute inéquation d'une description minimale induit une facette du polyèdre. Si le polyèdre est de pleine dimension, alors la description minimale

de P est unique modulo une multiplication des inéquations par un scalaire positif.

Le principal résultat qui nous intéresse et qui associe la plupart des notions définies ci-dessus est le théorème de Minkowski et Weyl [87] suivant :

Théorème 2.2.2 *Soit P un sous-ensemble de \mathbb{R}^n . P est un polyèdre si et seulement si P est une enveloppe.*

Pour la plupart des cas, les ensembles que l'on considère représentent des enveloppes convexes. Pour notre étude, la conséquence principale de ce théorème est qu'une enveloppe convexe est un polytope.

2.2.4.2 Algorithme de coupe

Soient une famille finie \mathcal{L} d'inéquations linéaires dans l'espace \mathbb{R}^m et c un vecteur de \mathbb{R}^m . Pour une sous-famille \mathcal{L}' de \mathcal{L} , on note $Q_{\mathcal{L}'}$ le polyèdre qu'elle induit et on définit le programme linéaire

$$P(\mathcal{L}') = \begin{cases} \text{Min } \sum_{i=1}^m c_i x_i \\ \text{t.q.} \\ x \in Q_{\mathcal{L}'} \end{cases}$$

et on note P le programme $P(\mathcal{L})$.

Si le nombre de contraintes dans \mathcal{L} est raisonnable, la résolution de P est facile. Il suffit de lister explicitement toutes les inéquations et résoudre P par un algorithme de programmation linéaire.

L'algorithme de coupe se propose de résoudre le programme linéaire P lorsque la famille \mathcal{L} est de très grande taille (exponentielle en nombre de variables), et qu'on ne peut pas lister toutes les contraintes de \mathcal{L} . Il s'exprime de manière générique comme suit :

Procédure Coupe;

Début

$\mathcal{L}' \leftarrow \mathcal{L}_0$; $\{\mathcal{L}_0$ est un sous-ensemble initial de contraintes $\}$
 Répéter
 Résoudre $P_{\mathcal{L}'}$ et soit $x_{\mathcal{L}'}$ la solution optimale ;
 Trouver un ensemble d'inéquations $\mathcal{L}'' \subset \mathcal{L} \setminus \mathcal{L}'$ violées par $x_{\mathcal{L}'}$;
 $\mathcal{L}' \leftarrow \mathcal{L}' \cup \mathcal{L}''$;
 Jusqu'à ce que $\mathcal{L}'' = \emptyset$;
 Fin.

Dans la procédure ci-dessus, la recherche d'une ou plusieurs contraintes dans \mathcal{L} violées par $x_{\mathcal{L}'}$ est appelée problème de séparation. Il est énoncé comme suit :

Problème de séparation : *Étant donné un point $x \in \mathbb{R}^m$ et une famille \mathcal{L} d'inéquations dans \mathbb{R}^m , trouver une ou plusieurs inéquations violées par x , ou prouver qu'aucune des inéquations de \mathcal{L} n'est violée par x .*

Grötschel, Lovàsz et Schrijver [45] ont montré que l'on peut résoudre le programme linéaire $P(\mathcal{L})$ polynomialement si et seulement si le problème de séparation associé est polynomial. L'algorithme de coupe est d'autant plus rapide que la résolution du problème de séparation l'est aussi.

Tout problème d'optimisation combinatoire *NP*-difficile peut s'écrire sous la forme d'un programme linéaire de type $P(\mathcal{L})$, avec un nombre, certes exponentiel, mais fini de contraintes. Lorsque \mathcal{L} est complètement déterminée, le problème de séparation sur la famille \mathcal{L} est évidemment *NP*-difficile. Malheureusement, pour la plupart des problèmes *NP*-difficiles, la famille \mathcal{L} n'est connue que partiellement et, dans ce cas également, le problème de séparation reste souvent difficile.

Finalement, pour résoudre un problème d'optimisation combinatoire *NP*-difficile en utilisant un algorithme de coupe, la démarche à suivre consiste, dans une première étape, à étudier le polyèdre associé pour obtenir une description partielle en inéquations et, dans une deuxième phase, à développer des procédures pour la résolution du problème de séparation sur l'ensemble, ou même un sous-ensemble, des inéquations de la description partielle connue.

On s'intéresse à des inéquations induisant des facettes pour obtenir une description partielle minimum en nombre d'inéquations et surtout obtenir des coupes profondes.

Pour le problème du *ITSP*, Fischetti et al. [28] sont les seuls à avoir développé un algorithme de branchement et de coupe pour le résoudre. Pour toute tournée internationale T , définie sur un ensemble légal L , ils associent un vecteur d'incidence $z^T = (x^T, y^T) \in \mathbb{R}^{|E|} \times \mathbb{R}^{|V|}$ tel que :

$$x_e = \begin{cases} 1 & \text{si } e \in T \\ 0 & \text{sinon} \end{cases} \quad \text{et} \quad y_v = \begin{cases} 1 & \text{si } v \in L \\ 0 & \text{sinon} \end{cases}$$

Ils considèrent le polytope associé à l'enveloppe convexe de l'ensemble des vecteurs d'incidence des tournées internationales ainsi définis. Une première description partielle du polytope est obtenue en relâchant toutes les contraintes d'intégralité de la formulation en programme linéaire en nombres entiers du *ITSP* suivante :

$$\begin{aligned} & \text{Min} \sum_{e \in E} c_e x_e \\ & \text{tel que} \\ & \begin{cases} x(\delta(\{v\})) = 2y_v, \forall v \in V & (3) \\ y(W_i) = 1, \forall i \in \{1, \dots, p\} & (4) \\ x(\delta(S)) \geq 2(y_i + y_j - 1), \forall S \subset V, i \in S, j \in V \setminus S & (5) \\ x_e \in \{0, 1\}, \forall e \in E & (6) \\ y_v \in \{0, 1\}, \forall v \in V & (7) \end{cases} \end{aligned}$$

Les contraintes (3) imposent que le nombre des arêtes incidentes à un sommet v quelconque est égal à 2 ou 0. Les contraintes (4) garantissent qu'un et un seul sommet est visité dans chaque pays. La connexité des solutions est assurée par les contraintes (5), et enfin les contraintes d'intégralité (6) et (7) traduisent l'intégralité sur l'ensemble des variables.

On notera que cette formulation a été reprise par tous les auteurs et qu'elle est à la base des algorithmes de branchement et d'évaluation pour la résolution du *ITSP*, mentionnés précédemment. Dans la suite, on donnera une nouvelle formulation du *ITSP*.

2.2.4.3 Une nouvelle formulation du ITSP

La formulation du *ITSP* en un programme linéaire en nombres entiers, présentée précédemment, associe des variables binaires aux arêtes ainsi qu'aux sommets du graphe. La formulation que nous présentons ici, n'associe des variables binaires qu'aux arêtes du graphe. À une tournée internationale T , on associe un vecteur d'incidence $x^T \in \mathbb{R}^{|E|}$ tel que :

$$x_e = \begin{cases} 1 & \text{si } e \in T \\ 0 & \text{sinon} \end{cases}$$

Le programme linéaire en nombres entiers obtenu est :

$$\begin{aligned} & \text{Min } \sum_{e \in E} c_e x_e \\ & \text{tel que} \\ & \begin{cases} x(\delta(W_i)) = 2, \forall i \in \{1, \dots, p\} & (8) \\ x(\delta(S)) \geq 2, \forall S = \cup_{i \in I} W_i, I \subset \{1, \dots, p\} & (9) \\ x(\delta(S)) \geq 2, \forall S = W \cup \{v\}, W \in \mathcal{W} \text{ et } v \in V \setminus W & (10) \\ x_e \in \{0, 1\}, \forall e \in E & (11) \end{cases} \end{aligned}$$

Les contraintes (8) imposent que le cocycle de chaque pays soit exactement égal à deux. Dans une tournée internationale, le cocycle d'un ensemble S quelconque ($\emptyset \neq S \neq V$) égal à l'union d'un certain nombre de pays est au moins égal à 2, d'où les contraintes (9). Pour imposer que chaque pays soit visité en exactement une seule ville, et ainsi éviter des discontinuités dans les solutions (voir figure 2.19), on introduit les contraintes (10). Finalement, on a les contraintes d'intégralité (11) sur les variables. On obtient ainsi un système qui décrit l'ensemble des tournées internationales de G .

Lorsque tous les pays sont dégénérés, cette nouvelle formulation donne la formulation classique du *TSP*. On notera qu'on peut également obtenir cette nouvelle formulation par projection sur l'espace des arêtes de la première formulation.

Si on considère le polytope associé à l'enveloppe convexe des vecteurs d'incidence de l'ensemble des tournées internationales, cette nouvelle formulation constitue le point de départ de son étude. En relâchant les contraintes

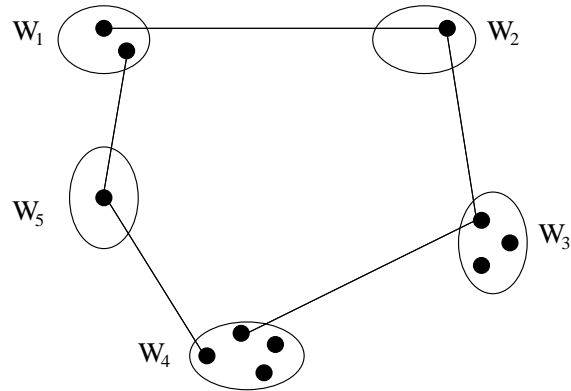


FIG. 2.19 – Une tournée discontinue.

d'intégralités (11), on obtient une première description linéaire partielle du polytope.

Dans le prochain chapitre, on étudie le polytope du *ITSP* dans l'espace $\mathbb{R}^{|E|}$ ainsi que les polytopes de certains problèmes associés et notamment sa relaxation graphique. On donnera une description partielle du polytope *ITSP* et on décrira certaines opérations sur les inéquations qui permettent d'obtenir des inéquations induisant de nouvelles facettes. Des procédures de séparation sont décrites dans le chapitre 4, ainsi que l'ensemble des éléments qui permettent la mise en œuvre d'un algorithme de branchement et de coupe.

Chapitre 3

Étude Polyédrale

Nous avons montré, dans la section 2.2.4 du chapitre précédent, que l'étude polyédrale est une phase nécessaire pour le développement d'un algorithme de branchement et de coupe. L'objectif principal de ce chapitre est d'obtenir une description linéaire partielle du polytope $ITSP(n, p)$ par des inéquations induisant des facettes.

Dans la première section, on donnera les différentes définitions et notations qu'on utilisera dans ce chapitre. Dans la section 3.2, on définit la relaxation graphique du $ITSP$ et on étudie certains polyèdres associés. Cette approche classique permet d'obtenir certaines propriétés de la structure faciale du polytope $ITSP(n, p)$ dont l'étude fera l'objet de la section 3.3. Après l'étude de la dimension du polytope $ITSP(n, p)$ et des inéquations de non négativité, on détermine certaines propriétés de $ITSP(n, p)$ en étudiant la relation polyédrale qui existe entre le $ITSP$ et sa relaxation graphique. On distinguera deux classes de facettes de $ITSP(n, p)$: les facettes régulières et les facettes non régulières. Pour chaque classe, on donnera des inéquations induisant des facettes et on étudiera certaines propriétés que vérifient les facettes qu'on détermine. On finira la section en décrivant deux opérations sur les inéquations en vue d'obtenir des inéquations induisant des facettes de $ITSP(n + 1, p)$ à partir d'inéquations induisant des facettes de $ITSP(n, p)$.

3.1 Définitions et notations

Soient $G = (V, E, \mathcal{W})$ un graphe international et S un sous-ensemble de sommets. On note $G[S]$ le sous-graphe engendré par l'ensemble S et on rappelle que $E(S)$ et $\delta(S)$ désignent respectivement l'ensemble des arêtes ayant les deux extrémités dans S et l'ensemble des arêtes ayant exactement une extrémité dans S . L'ensemble $\delta(S)$ est dit cocycle associé à S . Pour S_1 et S_2 deux sous-ensembles de sommets disjoints, on désigne par $(S_1 : S_2)$ l'ensembles des arêtes ayant une extrémité dans S_1 et l'autre dans S_2 .

Pour un sous-ensemble S de sommets, on définit également les nombres:

$$\mu(S) = |\{h : W_h \subseteq S\}|$$

et

$$\eta(S) = |\{h : W_h \cap S = \emptyset\}|$$

qui représentent le nombre de pays respectivement à l'intérieur et à l'extérieur d'un ensemble S de sommets. Selon les valeurs des ces nombres, on distingue deux notions de sous-ensembles de sommets : les sous-ensembles valides et les sous-ensembles réguliers.

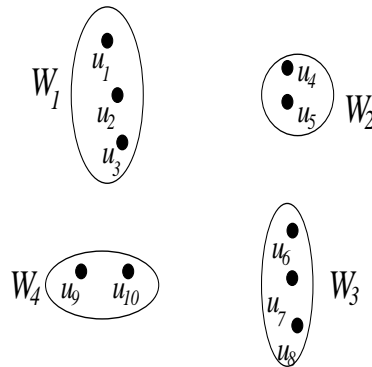


FIG. 3.1 – Graphe international.

Définition 3.1.1 *Un sous-ensemble S de sommets est dit valide si au moins un pays est contenu dans S et un autre dans $V \setminus S$; i.e. $\mu(S) > 0$ et $\eta(S) > 0$.*

Définition 3.1.2 *Un sous-ensemble S de sommets est dit régulier si tout pays est soit dans S soit dans $V \setminus S$; i.e. $S = \bigcup_{i \in I} W_i$ pour $I \subset \{1, \dots, p\}$ ou bien $\mu(S) + \eta(S) = p$.*

Ainsi, par définition, tout ensemble régulier est valide. La réciproque n'est pas vraie. Le sous-ensemble $S_1 = \{u_1, u_2, u_3, u_9, u_{10}\}$ de la figure 3.1 est un ensemble régulier, donc valide et le sous-ensemble $S_2 = \{u_4, u_5, u_6, u_7\}$ de la même figure est valide sans être régulier. Notons que les arêtes du graphe ne sont pas représentées sur la figure.

Dans la section 3.3.6, on s'intéressera aux sous-ensembles de sommets valides et non réguliers minimaux par inclusion qu'on appelle ensembles minimaux. Un ensemble minimal est égal à l'union d'un pays W et d'un sommet u appartenant à un pays W_u non dégénéré et différent de W . L'ensemble S_2 défini plus haut est valide et non régulier mais il n'est pas minimal par inclusion. Le sous-ensemble $S_3 = \{u_4, u_5, u_6\}$ est un ensemble minimal.

Dans la section 2.2.4.3 du chapitre précédent, les inéquations (9) et (10) de la formulation du *ITSP* en un programme linéaire en nombres entiers sont associées à des ensembles respectivement réguliers et minimaux.

On rappelle qu'un sous-ensemble L de sommets est dit légal s'il rencontre exactement une seule fois chaque pays; i.e. $|L \cap W_i| = 1$, pour $i = 1, \dots, p$. Toute tournée internationale T induit un ensemble légal unique défini par les sommets visités par T , et noté L^T . Le sous-ensemble $L = \{u_2, u_4, u_7, u_9\}$ de la figure 3.1 est un ensemble légal.

Soit $\mathbb{R}^{|E|}$ l'espace de tous les vecteurs réels dont les composantes sont indexés par l'ensemble E . Pour tout vecteur $x \in \mathbb{R}^{|E|}$, on note x_e , ou $x_{u,v}$, la composante du vecteur x indexée par l'arête $e = [u, v]$. Pour tout sous-ensemble F d'arêtes de G , on note $G(F)$ le sous-graphe partiel induit par F et on lui associe un vecteur d'incidence unique x^F dans $\mathbb{R}^{|E|}$ défini par :

$$x_e = \begin{cases} 1 & e \in F, \\ 0 & \text{sinon.} \end{cases}$$

Soient a un vecteur de $\mathbb{R}^{|E|}$ et S un sous-ensemble de sommets, on associe

au vecteur a un vecteur a^* de $\mathbb{R}^{|E(S)|}$ tel que :

$$a_e^* = a_e \quad \forall e \in E(S).$$

On appelle a^* la restriction du vecteur a à l'ensemble S .

Étant donné une inéquation $cx \geq c_0$ de $\mathbb{R}^{|E|}$ et un sous-ensemble de sommets S , on considère l'inéquation $c^*x \geq c_0$ de $\mathbb{R}^{|E(S)|}$ déterminée par la restriction c^* du vecteur c à l'ensemble S . L'inéquation $c^*x \geq c_0$ est appelé restriction de l'inéquation $cx \geq c_0$ à l'ensemble S .

On désigne par A la matrice d'incidence sommet-arête classique définie par

$$A_{kj} = \begin{cases} 1 & \text{si } e_j \in \delta(v_k) \\ 0 & \text{sinon} \end{cases}$$

pour $k = 1, \dots, n$ et $j = 1, \dots, m$ et on définit la matrice d'incidence pays-arête du graphe G , notée A^* , par :

$$A_{ij}^* = \begin{cases} 1 & \text{si } e_j \in \delta(W_i) \\ 0 & \text{sinon} \end{cases}$$

pour $i = 1, \dots, p$ et $j = 1, \dots, m$. On désigne par $\mathbb{1}$ le vecteur dont chacune des composantes est égale à 1.

Soit \mathcal{I} l'ensemble des tournées internationales dans le graphe G . Le polytope du *ITSP* associé au graphe G , noté $ITSP(G)$, est l'enveloppe convexe de l'ensemble des vecteurs d'incidence des tournées internationales appartenant à \mathcal{I} ; i.e. $ITSP(G) = \text{conv} \{x^I \in \mathbb{R}^{|E|} : I \in \mathcal{I}\}$. Le polytope est noté $ITSP(n, p)$ lorsque G est un graphe multiparti complet de n villes et p pays.

Soit \mathcal{H} l'ensemble des cycles hamiltoniens dans le graphe G . Le polytope du *TSP* associé au graphe G , noté $TSP(G)$, est l'enveloppe convexe de l'ensemble des vecteurs d'incidence des cycles hamiltoniens appartenant à \mathcal{H} ; i.e. $TSP(G) = \text{conv} \{x^H \in \mathbb{R}^{|E|} : H \in \mathcal{H}\}$. Lorsque G est le graphe complet K_n , le polytope est noté $TSP(n)$. Dans la suite, on s'intéressera au polytope du *TSP* sous-jacent au graphe, d'ordre p , engendré par un ensemble légal L de V .

L'approche classique pour l'étude de la description linéaire des polytopes des problèmes d'optimisation combinatoire *NP*-difficiles qui ne sont pas de pleines dimensions consiste à considérer des relaxations dont les polytopes associés sont généralement de pleines dimensions. L'étude de ces polytopes est relativement plus simple. Leurs structures faciales sont généralement très riches en informations sur la structure faciale du polytope initial.

Le *TSP* n'a pas échappé à cette approche. Une grande partie des résultats connus sur le polytope du *TSP* ont été obtenus en étudiant de nombreuses relaxations. On a longtemps étudié la monotonisation du problème [63], la relaxation du 2-couplage [26, 42, 41] et la relaxation du 1-arbre [52, 32]. On notera que le polytope associé à cette dernière relaxation n'est pas de pleine dimension. Ces dernières années, c'est la relaxation graphique [18] qui a eu le plus de "succès" ; elle permet une meilleure compréhension de la structure faciale du polytope du *TSP*.

Dans la section suivante, on étudie une relaxation du *ITSP* inspirée de la relaxation graphique du *TSP*, ou encore le problème du voyageur de commerce graphique, noté *GTSP*. Cette relaxation du *TSP* est relativement récente ; elle a été introduite pour la première fois en 1985 par Cornuéjols, Fonlupt et Naddef [18]. Les solutions considérées, appelées tours, sont les familles d'arêtes dont le graphe partiel est un graphe eulérien couvrant. Le *GTSP* consiste à trouver un tour de longueur minimum.

Pour tout tour T , on associe un vecteur d'incidence unique x^T dans $\mathbb{R}^{|E|}$ tel que $x_e = d_e$, avec d_e le nombre de fois où l'arête e est dans le tour T . Soit \mathcal{K} l'ensemble des tours dans G . Le polyèdre associé au *GTSP*, noté $GTSP(G)$, est l'enveloppe convexe de l'ensemble des vecteurs d'incidence des tours appartenant à \mathcal{K} ; i.e. $GTSP(G) = \text{conv} \{x^K \in \mathbb{R}^{|E|} : K \in \mathcal{K}\}$.

3.2 La relaxation graphique

La relaxation graphique du *TSP* a permis d'obtenir des résultats très intéressants sur la structure faciale du polytope du *TSP*. Ceci nous a poussé à étudier la généralisation de cette relaxation dans le cas du *ITSP*. Dans cette section, on définit la relaxation graphique du *ITSP* et on montre que la plu-

part des résultats obtenus sur le polytope du *TSP* en utilisant la relaxation graphique restent vérifiés pour le polytope du *ITSP*.

Dans cette relaxation, on considère les solutions qui permettent de visiter un pays plusieurs fois dans une ou plusieurs villes. Plus précisément, on considère les solutions dont le graphe partiel est eulérien et incident à chaque pays en au moins un sommet. On appelle une telle solution tour international.

Un tour international T est caractérisé comme un sous-ensemble d'arêtes qui vérifie les propriétés suivantes :

- (1) *le degré de chaque sommet de V est pair ou nul;*
- (2) *$G(T)$ est connexe;*
- (3) *le cocycle de chaque pays est supérieur ou égal à deux.*

Une tournée internationale vérifie les propriétés (1) et (2), et la propriété (3) avec “exactement ” au lieu de “supérieur ou égal”, i.e. la propriété suivante :

- (3') *le cocycle de chaque pays est exactement égal à deux.*

Le problème du voyageur de commerce international graphique, noté *GITSP*, consiste à trouver un tour international de poids minimum. Soit \mathcal{T} l'ensemble des tours internationaux dans le graphe G . On note $P_0(G)$ l'enveloppe convexe des tours internationaux appartenant à l'ensemble \mathcal{T} .

Le *GITSP* n'a de solution finie que si toutes les arêtes de E ont des coûts positifs ou nuls (sachant que le problème est une minimisation). En outre, dans ce cas là, le problème d'optimisation sur le polytope $P_0(G)$ admet une solution dont le vecteur d'incidence vérifie la propriété suivante :

- (4) *les composantes sont égales à 0, 1 ou 2.*

En effet, soit x^* une solution optimum du *GITSP* dont une des composantes, disons x_e^* , est égale à k strictement supérieur à 2. On construit une

autre solution dont le vecteur d'incidence \bar{x} est égale à x^* sur l'ensemble des composantes sauf sur la composante x_e^* où

$$\bar{x}_e = \begin{cases} 1 & \text{si } k \text{ est impair} \\ 2 & \text{sinon} \end{cases}$$

La solution \bar{x} à un poids au plus égal à celui de x^* ; \bar{x} est une solution optimum du *GITSP*.

On appellera points extrêmes les tours internationaux qui vérifient la propriété (4) et on notera $P_1(G)$ l'enveloppe convexe des points extrêmes dans un graphe multiparti G .

Dans le cas où la fonction coût p vérifie l'inégalité triangulaire, on montre dans la section 3.2.2.3 que l'on peut trouver une solution optimale qui vérifie la propriété (4) ainsi que la propriété suivante :

(5) *pour chaque pays, exactement un seul sommet est atteint.*

On appellera points extrêmes minimaux les points extrêmes qui vérifient la propriété (5) et on notera $P_2(G)$ l'enveloppe convexe des points extrêmes minimaux sur un graphe multiparti G .

les polyèdres précédemment définis sont liés par la relation d'inclusion suivante:

$$P_2(G) \subset P_1(G) \subset P_0(G).$$

On notera $P_0(n, p)$, $P_1(n, p)$ et $P_2(n, p)$ respectivement les polyèdres $P_0(G)$, $P_1(G)$ et $P_2(G)$ quand le graphe G est le graphe multiparti complet K_n^p .

3.2.1 Étude de la dimension

Dans cette section, on étudie la dimension des polyèdres précédemment définis pour des graphes multipartis complets et, pour certains polyèdres,

pour des graphes multipartis quelconques. Les démonstrations de la plupart des résultats de cette section sont inspirées de [18].

Théorème 3.2.1 *Si le graphe G est connexe, le polyèdre $P_0(G)$ est de pleine dimension.*

Preuve : Considérons un tour international x obtenue en dédoublant chaque arête d'un arbre couvrant et les tours internationaux $x + 2y^i$, pour $i = 1, \dots, m$, où y^i est le vecteur unité tel que $y_i^i = 1$ et $y_j^i = 0$ pour $j \neq i$. Ces $m + 1$ tours internationaux sont affinements indépendants. \square

Dans la suite, on étudie la structure faciale du polytope $P_0(G)$ pour la cas d'un graphe multiparti complet. La dimension dans ce cas là est donnée par le corollaire suivant :

Corollaire 3.2.1 *Le polytope $P_0(n, p)$ est de pleine dimension.*

Pour l'étude de la dimension du polytope $P_1(G)$ dans le cas d'un graphe multiparti quelconque, on définit la notion d'isthme valide.

Définition 3.2.1 *Une arête e est appelé isthme valide d'un graphe G connexe si e est un isthme ; i.e. le graphe partiel $G(E \setminus \{e\})$ a deux composantes connexes, et les ensembles de sommets qui induisent les deux composantes connexes de $G(E \setminus \{e\})$ sont valides.*

Théorème 3.2.2 *Si le graphe G est connexe avec k isthmes valides, le polytope $P_1(G)$ est de dimension $m - k$.*

Preuve : Si une arête e est un isthme valide, alors $x_e = 2$ pour tout point extrême de G . La dimension de $P_1(G)$ est au plus égale à $m - k$, avec k le nombre d'isthmes valides de G .

Réciproquement, on montre par récurrence sur le nombre des arêtes de G que $\dim(P_1(G)) \geq m - k$. L'inégalité est vérifiée pour $|E| = 1$. Soit un graphe G connexe admettant plus d'une arête. Considérons une arête e de G telle que le graphe $G(E \setminus \{e\})$ est connexe avec, éventuellement, un sommet isolé et supposons que les arêtes de $E \setminus \{e\}$ sont numérotées de 1 à $m - 1$.

Si l'arête e est un isthme, alors elle est forcément pendante et le graphe $G(E \setminus \{e\})$ n'admet pas de nouveaux isthmes valides. On distingue deux cas selon que l'arête e est un isthme valide ou non valide. Dans le premier cas, le graphe $G(E \setminus \{e\})$ a $k - 1$ isthmes valides. Par hypothèse de récurrence, il admet $(m - 1) - (k - 1) + 1$ points extrêmes affinements indépendants. On peut étendre chacun de ces points en ajoutant la composante $x_e = 2$ pour obtenir $m - k + 1$ points extrêmes de G affinements indépendants. Dans le deuxième cas, le graphe $G(E \setminus \{e\})$ a k isthmes valides. Par hypothèse de récurrence, il admet $(m - 1) - k + 1$ points extrêmes affinements indépendants. On peut étendre chacun de ces points en ajoutant la composante $x_e = 0$ pour obtenir $m - k$ points extrêmes de G affinements indépendants. Si on considère un point extrême quelconque admettant une composante x_e non nulle et l'ensemble des $m - k$ points extrêmes précédemment considérés, on obtient un ensemble $m - k + 1$ points extrêmes de G affinements indépendants.

Dans le cas où l'arête e n'est pas un isthme, le graphe $G(E \setminus \{e\})$ a les k isthmes valides de G et, éventuellement, p nouveaux isthmes valides, disons e_1, \dots, e_p . Par hypothèse de récurrence, la dimension de $P_1(G(E \setminus \{e\}))$ est $(m - 1) - (k + p)$. Considérons un ensemble X de $m - (k + p)$ points extrêmes affinements indépendants de $P_1(G(E \setminus \{e\}))$ tels que les composantes d'un point extrême de X , disons x^* , sont $x_{e_i}^* = 2$ pour toute arête e_i d'un arbre couvrant A et $x_{e_i}^* = 0$ pour toute arête e_i n'appartenant pas à A . Un tel ensemble de points extrêmes affinements indépendants existe ; il suffit de considérer une ensemble X' quelconque de $m - (k + p)$ points extrêmes affinements indépendants. Le point extrême x^* , associé à un arbre quelconque A , s'écrit comme une combinaison convexe unique des points de X' . Soit \bar{x} un élément de X' ayant un coefficient strictement positif dans la combinaison convexe des éléments de X' donnant x^* . L'ensemble $X = X' \cup \{x^*\} \setminus \{\bar{x}\}$ est un ensemble de $m - (k + p)$ points extrêmes affinements indépendants.

On peut, d'une part, étendre les éléments de X en points extrêmes de G en ajoutant simplement la composante $x_e = 0$. D'autre part, on construit $p + 1$ nouveaux points extrêmes à partir de x^* en considérant une chaîne simple P de $G(E \setminus \{e\})$ joignant les extrémités de l'arête e et n'utilisant que des arêtes de l'arbre A . Les isthmes valides e_1, \dots, e_p , nouvellement créés, appartiennent à la chaîne P . Sans perdre de généralité, on peut supposer que

les arêtes de A sont les r premières arêtes de $E \setminus \{e\}$ et que les arêtes de P sont les q premières arêtes de $E \setminus \{e\}$ ($q \leq r$). Soient le point extrême x^0 tel que $x_{e_i}^0 = x_{e_i}^*$ pour $i \notin P \cup \{e\}$, $x_{e_i}^0 = x_{e_i}^* - 1$ pour $i \in P$ et $x_e^0 = 1$, et les points extrêmes x^j , pour $j = 1, \dots, p$, tels que $x_{e_i}^j = x_{e_i}^0$ pour $e_i \notin P \cup \{e\}$, $x_{e_i}^j = 2$ pour $e_i \in P \cup \{e\} \setminus \{e_j\}$ et $x_{e_j}^j = 0$. Ainsi, au total, on a défini $m - k + 1$ points extrêmes.

Soit M la matrice $m \times (m - k + 1)$, ci-dessous représentée, dont les lignes représentent les arêtes de G et les colonnes représentent les vecteurs d'incidence des points extrêmes précédemment définis. On vérifie que l'ensemble des colonnes de la matrice M forment un ensemble de vecteurs affinements indépendants.

		x^*	x^0	x^1	x^2	\dots	x^p			
e	0	0	...	0	0	1	2	2	2	2
e_1	2	2	...	2	2	1	0	2	2	2
e_2	2	2	...	2	2	1	2	0	2	2
\vdots	2	2	...	2	2	1	2	2	0	2
e_p	2	2	...	2	2	1	2	2	2	0
e_{p+1}	$m - (k + p) - 1$ colonnes				2	1	2	2	2	2
\vdots					\vdots	\vdots	\vdots	\vdots	\vdots	
e_q					2	1	2	2	2	2
e_{q+1}					2	2	2	2	2	2
\vdots					\vdots	\vdots	\vdots	\vdots	\vdots	
e_r					2	2	2	2	2	2
e_{r+1}					0	0	0	0	0	0
\vdots					\vdots	\vdots	\vdots	\vdots	\vdots	
e_{m-1}					0	0	0	0	0	0

X

Ainsi, on a $m - k + 1$ points extrêmes affinements indépendants et, par conséquent, la dimension de $P_1(G)$ est supérieure ou égale à $m - k$. \square

Corollaire 3.2.2 *Le polytope $P_1(n, p)$ est de pleine dimension.*

Ce corollaire est un cas particulier du résultat précédant. Le graphe K_n^p n'a pas d'isthme valide. Par ailleurs, c'est également une conséquence directe du résultat suivant. En effet, le polytope $P_2(n, p)$ est contenu dans le polytope $P_1(n, p)$.

Théorème 3.2.3 *Le polytope $P_2(n, p)$ est de pleine dimension.*

Preuve : Lorsque le nombre de sommets n est égal au nombre de pays p , i.e. tout les pays sont dégénérés, $P_2(n, p)$ est le polytope du problème du voyageur de commerce graphique. Dans [18], Cornuéjols, Fonlupt et Naddef montrent que, dans ce cas, $P_2(n, p)$ est de pleine dimension. Pour le cas où n est quelconque, on procède par récurrence sur le nombre de sommets n sachant un nombre de pays fixé. Soit u un sommet quelconque du graphe K_n^p appartenant à un pays non dégénéré, disons W_p . Le graphe engendré par l'ensemble $V \setminus \{u\}$ est le graphe K_{n-1}^p . Par hypothèse de récurrence, le graphe K_{n-1}^p contient $m - |\delta\{u\}| + 1$ points extrêmes minimaux affinements indépendants.

On construit $|\delta\{u\}|$ nouveaux points extrêmes en considérant, pour toute arête $e_i = [u, i] \in \delta\{u\}$, le point extrême minimal $x^i + 2y^i$, où x^i est le vecteur d'incidence d'un cycle incident au sommet i et qui visite un et un seul sommet de tout pays W différent de W_p , et y^i est le vecteur unité tel que $y_{e_i}^i = 1$ et $y_{e_j}^i = 0$ pour $e_j \neq e_i$. Les points extrêmes minimaux ainsi définis et les $m - |\delta\{u\}| + 1$ points extrêmes minimaux précédents non incidents au sommet u constituent un ensemble de $m + 1$ points extrêmes minimaux affinements indépendants de K_n^p . \square

La section suivante est réservée à l'étude de la structure faciale du polytope $P_0(G)$. On déterminera sous quelles conditions les inéquations de non négativité et les inéquations de coupe induisent une facette du polytope $P_0(G)$ et on montrera certaines propriétés que vérifient les inéquations de non négativité. On finira la section en étudiant une propriété générale que vérifient les inéquations qui induisent des facettes du polytope $P_0(G)$ autres que les inéquations de non négativité et les inéquations de coupe.

3.2.2 Structure faciale de $P_0(G)$

La propriété principale de la description linéaire du polytope $P_0(G)$, et d'une manière générale des polytopes de pleine dimension, est qu'elle est

unique modulo une multiplication des deux membres d'une ou plusieurs de ces inéquations par un scalaire positif. Cette propriété permet d'établir les résultats avec une certaine facilité.

Soit L un sous-ensemble légal de sommets. On note $\mathcal{I}(L)$ et $\mathcal{T}(L)$ les sous-ensembles respectifs de \mathcal{I} et \mathcal{T} dont les éléments ne visitent que les sommets de l'ensemble L . Soit $fx \geq f_0$ une inéquation de $\mathbb{R}^{|E|}$, on note \mathcal{I}_f^- et \mathcal{T}_f^- les sous-ensembles respectifs de \mathcal{I} et \mathcal{T} dont les éléments saturent l'inéquation ; i.e. $\mathcal{I}_f^- = \{I \in \mathcal{I} : fx^I = f_0\}$ et $\mathcal{T}_f^- = \{T \in \mathcal{T} : fx^T = f_0\}$, et $\mathcal{I}_f^-(L) = \mathcal{I}_f^- \cap \mathcal{I}(L)$ et $\mathcal{T}_f^-(L) = \mathcal{T}_f^- \cap \mathcal{T}(L)$.

Avant d'étudier les inéquations induisant des facettes du $P_0(G)$ et leurs propriétés, on donnera le remarque suivante.

Remarque 3.2.1 *Soit $cx \geq c_0$ une inéquation valide pour $P_0(G)$. Pour toute arête e , le coefficient $c_e \geq 0$. En effet, s'il existe une arête admettant un coefficient négatif, alors un tour international qui l'utilise un nombre élevé de fois ne vérifie pas l'inéquation $cx \geq c_0$.*

3.2.2.1 Facettes triviales

Pour une arête e quelconque, l'inéquation 3.1 de non négativité est valide ; elle est appelée inéquation triviale.

$$x_e \geq 0 \tag{3.1}$$

Théorème 3.2.4 *L'inéquation 3.1 définit une facette du polytope $P_0(G)$ si et seulement si l'arête e n'est pas un isthme valide.*

Preuve : Si l'arête e est un isthme valide, alors $x_e \geq 2$ dans chaque tour international de G .

Réciproquement, supposons que l'arête e n'est pas un isthme valide. Soit x le vecteur d'incidence d'un tour international tel que $x_e = 0$. Un tel tour international existe puisque au moins une des composantes connexes du graphe partiel $G(E \setminus \{e\})$ est induite par un ensemble de sommets contenant un sous-ensemble légal ; sinon l'arête e serait un isthme valide. Le tour international x et les $(m - 1)$ tours internationaux $x + 2y^i$ sont affinements indé-

pendants, où y^i est le vecteur unité tel que $y_{e_i}^i = 1$ et $y_{e_j}^i = 0$ pour $e_j \neq e_i$, avec $e_i \neq e$. \square

Corollaire 3.2.3 *l'inéquation $x_e \geq 0$ définit une facette du polytope $P_0(n, p)$.*

Ce corollaire est une conséquence directe du théorème précédent puisque K_n^p n'a pas d'isthmes valides. Une caractérisation de ces inéquations triviales est donnée par le lemme suivant.

Lemme 3.2.1 *Soit $cx \geq c_0$ une inéquation induisant une facette du polytope $P_0(G)$. Si $c_0 = 0$ alors $cx \geq c_0$ est une inéquation de non négativité.*

Preuve : Soient $cx \geq 0$ une inéquation valide pour $P_0(G)$ et e une arête de G admettant un coefficient c_e strictement positif. Un tour international qui sature l'inéquation $cx \geq 0$ n'utilise pas l'arête e puisque $c \geq 0$, alors il sature également l'inéquation $x_e \geq 0$. Les inéquations $cx \geq 0$ et $x_e \geq 0$ induisent la même facette de $P_0(G)$. Or une facette de $P_0(G)$ est induite par une inéquation unique modulo une multiplication des deux membres de l'inéquation par un scalaire positif. L'inéquation $cx \geq 0$ est l'inéquation $x_e \geq 0$. \square

Ce lemme nous permet de montrer que les facettes du polytope $P_0(G)$ induites par les inéquations de non négativité sont les seules à vérifier la propriété énoncée dans le théorème suivant.

Théorème 3.2.5 *Soit $cx \geq c_0$ une inéquation induisant une facette de $P_0(G)$. S'il existe un sous-ensemble de sommets légal L tel que, pour tout tour international T de $\mathcal{T}(L)$, $cx^T = c_0$ alors $cx \geq c_0$ est une inéquation de non négativité.*

Preuve : Soient L un ensemble légal et M la matrice dont les colonnes sont les points extrêmes de $GTSP(L)$. Si c^* est la restriction du vecteur c à l'ensemble L , alors on a la relation $c^*M = c_0 \mathbb{I}$. Le polytope de $GTSP(L)$ est de pleine dimension, alors c^* est le vecteur nul et c_0 est égal à zéro. D'après le lemme 3.2.1, $cx \geq c_0$ est une inéquation de non négativité. \square

Les inéquations triviales vérifient également une autre propriété simple mais fréquemment utilisée dans les démonstrations qui suivent.

Remarque 3.2.2 Soit $cx \geq c_0$ une inéquation induisant une facette du polytope $ITSP(n, p)$. S'il existe une arête e telle qu'aucune tournée internationale de \mathcal{I}_c^- n'utilise l'arête e , alors $cx \geq c_0$ est l'inéquation triviale associée à l'arête e . Cette remarque est vérifiée pour tout polytope admettant des facettes induites par les inéquations triviales.

Dans la section 3.3.3, on montrera que, dans le cas de $ITSP(n, p)$, la propriété décrite dans le théorème 3.2.5 n'est plus vérifiée.

3.2.2.2 Inéquations de coupe

Pour un sous-ensemble S de sommets quelconque, l'inéquation 3.2 est appelée inéquation de coupe. Elle est dite inéquation de degré, lorsque le sous-ensemble S se réduit à un seul pays.

$$x(\delta(S)) \geq 2 \tag{3.2}$$

Soit S un sous-ensemble de sommets. Pour un sommet u , on désigne par S_u l'ensemble de sommets de la composante connexe de $G[S]$ ou de $G[V \setminus S]$ qui contient le sommet u et, pour une arête $f = [u, v]$ appartenant à $\delta(S)$, on note $S_{u,v} = S_u \cup S_v$.

Théorème 3.2.6 *l'inéquation 3.2 définit une facette du polytope $P_0(G)$ si et seulement si l'ensemble S est valide et pour toute arête $f = [u, v]$ appartenant à $\delta(S)$, $S_{u,v}$ contient un ensemble légal de sommets.*

Preuve : Montrons que la condition est nécessaire. D'une part, si l'ensemble S n'est pas valide, alors il existe un tour international T tel que $x^T(\delta(S)) = 0$. D'autre part, si $\delta(S) = \{f\}$, avec $f = [u, v]$, alors f est un isthme. Puisque G est connexe, les graphes $G[S]$ et $G[V \setminus S]$ le sont aussi et, par conséquent, $S_{u,v} = V$. Supposons que $|\delta(S)| \geq 2$ et qu'il existe une arête $f = [u, v]$ appartenant à $\delta(S)$ telle que $S_{u,v}$ ne contient pas un ensemble légal de sommets. L'arête f n'est pas un isthme valide. En effet, si f est un isthme valide alors l'inéquation $x_f \geq 2$ est valide et domine l'inéquation $x(\delta(S)) \geq 2$. Soit T un tour international tel que $x^T(\delta(S)) = 2$ et qui utilise l'arête f . Un tel tour international existe sinon l'inéquation est dominée par l'inéquation triviale $x_f \geq 0$. Si l'ensemble $S_{u,v}$ ne contient pas d'ensemble

légal et le graphe partiel associé au tour international T est connexe, alors $x^T(\delta(S_{u,v})) \geq 2$. Or $\delta(S_{u,v}) + f \subset \delta(S)$ et $x_f^T \geq 1$, alors $x^T(\delta(S)) \geq x^T(\delta(S_{u,v})) + x_f^T \geq 3$. Ceci contredit l'hypothèse que $x^T(\delta(S)) = 2$.

Réciproquement, pour toute arête $f = [u, v]$ de $\delta(S)$, soit x^f le vecteur d'incidence associé à la famille d'arêtes obtenue en considérant deux fois l'arête f et deux fois l'ensemble des arêtes dans $G[S_u]$ et $G[S_v]$. Le vecteur x^f est le vecteur d'incidence d'un tour international puisque $S_{u,v}$ contient un ensemble légal.

D'autre part, pour toute arête $i \in E \setminus \delta(S)$, considérons le tour international $x^{f(i)} + 2y^i$, où y^i est le vecteur unité tel que $y_i^i = 1$ et $y_j^i = 0$ pour $j \neq i$ et $x^{f(i)}$ est le vecteur d'incidence du tour international précédemment défini pour une arête $f(i) = [u, v]$ appartenant à $\delta(S)$ telle que les deux extrémités de l'arête i sont dans $S_{u,v}$. Les $|E|$ tours internationaux $\{x^f : f \in \delta(S)\} \cap \{x^{f(i)} + 2y^i : i \in E \setminus \delta(S)\}$ sont linéairement indépendants et saturent l'inéquation $x(\delta(S)) \geq 2$. \square

Lemme 3.2.2 *l'inéquation 3.2 définit une facette du polytope $P_0(n, p)$ pour tout sous-ensemble S valide de V .*

En effet, dans le cas d'un graphe multiparti complet K_n^p , $S_{u,v}$ est l'ensemble des sommets V pour toute arête $f = [u, v]$ appartenant à $\delta(S)$.

Dans la section suivante, on introduit la notion d'inégalité triangulaire serrée qui nous permet de distinguer trois classes de facettes du polytope $P_0(n, p)$.

3.2.2.3 Forme triangulaire serrée

Dans [69], Naddef et Rinaldi définissent la forme triangulaire serrée d'une inéquation pour le cas du *GTSP*. Ici, on généralise la définition pour le cas du *GITSP*. On définit l'inégalité triangulaire classique avant d'introduire la notion d'inégalité triangulaire serrée.

Définition 3.2.2 *Une inéquation $fx \geq f_0$ est dite triangulaire si ses coefficients vérifient l'inégalité triangulaire, i.e. :*

$$(i) \quad f(u, v) \leq f(u, w) + f(w, v), \quad \forall (u, v, w) \in V^3, \quad w \neq u \neq v.$$

Définition 3.2.3 Soit $fx \geq f_0$ une inéquation triangulaire dans $\mathbb{R}^{|E|}$. Pour tout sommet w de V , on définit un sous-ensemble d'arêtes non incidentes au sommet w , noté $\Delta_f(w)$, de la manière suivante :

$$\Delta_f(w) = \{[u, v] \in E : w \neq u, w \neq v, f(u, v) = f(u, w) + f(w, v)\}.$$

Définition 3.2.4 Une inéquation triangulaire $fx \geq f_0$ valide pour le polytope $P_0(n, p)$ est dite serrée si elle vérifie la condition suivante :

$$(ii) \Delta_f(W) \neq \emptyset, \forall W \in \mathcal{W}, \text{ avec } \Delta_f(W) = \bigcup_{w \in W} \Delta_f(w).$$

On dira que l'inéquation $fx \geq f_0$ est FTS.

Lemme 3.2.3 Si le cocycle d'un pays W dans un tour international T est supérieur à deux, alors il existe deux arêtes $[w, u]$ et $[w, v]$ appartenant à T , avec $w \in W$, tel que $T' = T - [w, u] - [w, v] + [u, v]$ est un tour international.

Preuve : Soient T un tour international et W un pays tel que le cocycle associé dans T est strictement supérieur à 2. Supposons qu'il existe un sommet $w \in W$ tel que le degré de w dans T soit égal à deux, alors $T' = T - [w, u] - [w, v] + [u, v]$, avec $[w, u]$ et $[w, v]$ appartenant à T , est un tour international.

Si le degré de chaque sommet de W est nul ou strictement supérieur à 2, alors on considère l'ensemble S des sommets induit par T et le graphe $G[S]$ engendré par S . Le tour international T est un tour au sens du *GTSP* dans le graphe $G[S]$. Soit w un sommet appartenant à $S \cap W$, le degré de w dans T est strictement supérieur à 2. Dans le paragraphe 2.1.1.1 du chapitre précédent, nous avons vu qu'il existe deux arêtes $[w, u]$ et $[w, v]$ appartenant à T telles que $T' = T - [w, u] - [w, v] + [u, v]$ soit un tour dans $G[S]$. Le tour T' est un tour international dans G . \square

Une conséquence de ce lemme est que le *GITSP* admet une tournée internationale comme solution optimale lorsque la fonction coût p est triangulaire. Aussi, il permet d'établir le théorème suivant qui donne le résultat principal sur la structure faciale du polytope $P_0(n, p)$.

Théorème 3.2.7 Une inéquation $cx \geq c_0$ induisant une facette du polyèdre $P_0(n, p)$ appartient à l'une des trois classes suivantes :

$$(a) \text{ inéquations triviales } x_e \geq 0, \text{ pour toute arête } e \in E;$$

- (b) inéquations de degré $x(\delta(W)) \geq 2$, pour tout pays $W \in \mathcal{W}$;
- (c) inéquations FTS .

Preuve : Soit $cx \geq c_0$ une inéquation induisant une facette du polyèdre $P_0(n, p)$. Supposons qu'elle n'est pas triangulaire, alors il existe trois sommets u, v et w , appartenant à des pays deux à deux disjoints, tels que $c(u, v) > c(u, w) + c(w, v)$. Soit T une tour internationale appartenant à \mathcal{T}_c^- et qui utilise l'arête $[u, v]$. Le tour international $T' = T - [u, v] + [u, w] + [w, v]$ est tel que $cx^{T'} < c_0$. Ceci contredit l'hypothèse que $cx \geq c_0$ est une inéquation valide, alors il n'existe pas de tour international qui appartient à \mathcal{T}_c^- et qui utilise l'arête $[u, v]$. L'inéquation $cx \geq c_0$ est l'inéquation triviale $x_{u,v} \geq 0$.

Supposons que $cx \geq c_0$ est une inéquation triangulaire qui n'est pas serrée. Soient W un pays tel que $\Delta_c(W) = \emptyset$ et T un tour international appartenant à \mathcal{T}_c^- tel que le cocycle de W dans T soit strictement supérieur à deux. D'après le lemme précédent, il existe un sommet w appartenant à W et deux arêtes $[w, u]$ et $[w, v]$ appartenant à T tels que $T' = T - [w, u] - [w, v] + [u, v]$ soit un tour international. On a que $cx^{T'} = c_0 + (c(u, v) - c(u, w) - c(w, v))$. L'ensemble $\Delta_c(W)$ est vide, alors $cx^{T'} < c_0$. Ceci contredit l'hypothèse que $cx \geq c_0$ est une inéquation valide, alors il n'existe pas de tour international appartenant à \mathcal{T}_c^- tel que le cocycle du pays W soit strictement supérieur à 2. L'inéquation $cx \geq c_0$ est l'inéquation de degré $x(\delta(W)) \geq 2$. \square

Ainsi, toutes les inéquations qui induisent des facettes du polytope $P_0(n, p)$ différentes de celles induites par les inéquations triviales ou les inéquations de degré sont des inéquations FTS . Dans la section suivante, réservée à l'étude du polytope $ITSP(n, p)$, on étudiera la relation qui existe entre les structures faciales des polytopes $P_0(n, p)$ et $ITSP(n, p)$.

3.3 Le polytope $ITSP(n, p)$

Le principal objectif de cette section consiste à obtenir une description linéaire partielle du polytope $ITSP(n, p)$ en inéquations induisant des facettes. La dimension du polytope $ITSP(G)$ n'est pas connue dans le cas d'un graphe multiparti G quelconque. Dans la section 3.3.2, on étudie la dimension du polytope $ITSP(n, p)$ et, dans la section 3.3.4, on détermine

certaines propriétés de la structure faciale de $ITSP(n, p)$ en étudiant la relation qui existe entre le polytope $ITSP(n, p)$ et le polytope de pleine dimension $P_0(n, p)$.

Après l'étude des inégalités de non négativité et leurs propriétés dans la section 3.3.3, on distinguera deux classes de facettes de $ITSP(n, p)$: les facettes régulières et les facettes non régulières. Dans la section 3.3.5, on montre qu'à chaque facette régulière est associée une facette de $TSP(p)$ et, dans la section 3.3.6, on présente des facettes non régulières généralisant certaines facettes régulières. Avant de conclure la section, on donne deux résultats décrivant deux opérations sur les inégalités qui permettent d'obtenir des inéquations induisant des facettes de $ITSP(n+1, p)$ à partir d'inéquations induisant des facettes de $ITSP(n, p)$. Pour simplifier les énoncés des résultats qui suivent, on supposera que le nombre de pays p est supérieur ou égal à 5.

Avant d'étudier la dimension du polytope $ITSP(n, p)$, on décrira, dans la section qui suit, une manière qui permet de générer un ensemble de tournées affinements indépendantes, et qu'on utilisera dans la preuve de nombreux résultats.

3.3.1 Ensemble (W, u) -maximal

Dans la suite de ce chapitre, il sera souvent question d'exhiber un certain nombre de tournées internationales affinements indépendantes qui vérifient une certaine propriété. Si on considère l'union de deux ensembles disjoints de tournées internationales affinements indépendantes, on n'obtient pas nécessairement un ensemble affinement indépendant. Dans ce qui suit, on définit la notion d'ensemble de tournées internationales (W, u) -maximal qui permet, sous certaines conditions, d'obtenir une union affinement indépendante.

Soient W un pays et u un sommet appartenant à un pays différent de W . On suppose que les pays sont indexés par l'ensemble $\{1, \dots, p\}$ et que le pays W est W_1 et que u appartient au pays W_p . On suppose également que les sommets sont indexés par l'ensemble $\{1, \dots, n\}$ tels que deux sommets quelconques v_i et v_j appartenant respectivement aux pays W_l et W_k vérifient

la propriété $[l < k \Rightarrow i < j]$ et on note n_k l'indice le plus élevé des sommets du pays W_k , pour $k = 1, \dots, p$. Sans perte de généralité, on peut supposer que le sommet u est le sommet v_n . Dans ces conditions, on définit un ensemble de tournées internationales, noté $\mathcal{S}_{W,u}$, dans les trois points suivants :

- (i) $T_{1,i}$ une tournée internationale qui utilise les arêtes $[u, v_1]$ et $[u, v_i]$, pour $i = n_1 + 1, \dots, n_{p-1}$;
- (ii) T_{i,n_1+1} une tournée internationale qui utilise les arêtes $[u, v_{n_1+1}]$ et $[u, v_i]$, pour $i = 2, \dots, n_1$;
- (iii) T_{n_2,n_3} une tournée internationale qui utilise les arêtes $[u, v_{n_2}]$ et $[u, v_{n_3}]$.

Le graphe est multiparti complet, alors un ensemble $\mathcal{S}_{W,u}$ existe et il est appelé (W, u) -maximal. Sa cardinalité est égale au nombre d'arêtes incidentes au sommet u . On notera qu'à l'exception de la tournée internationale T_{n_2,n_3} , toutes les tournées internationales d'un ensemble (W, u) -maximal utilisent une arête $[u, w]$, avec w un sommet de W . Le lemme suivant montre qu'un ensemble (W, u) -maximal est linéairement indépendant.

Lemme 3.3.1 *Soient W un pays et u un sommet n'appartenant pas à W . Un ensemble (W, u) -maximal $\mathcal{S}_{W,u}$ des tournées internationales est un ensemble linéairement indépendant.*

Preuve : les tournées internationales T_{1,n_2} , T_{1,n_3} et T_{n_2,n_3} sont linéairement indépendantes (voir figure 3.2). Soit T une tournée internationale de $\mathcal{S}_{W,u}$ différente des trois précédentes. Il existe une arête incidente au sommet u telle que T est la seule tournée internationale de $\mathcal{S}_{W,u}$ à l'utiliser. Ainsi, l'ensemble $\mathcal{S}_{W,u}$ est linéairement indépendant. \square

Dans le lemme suivant, on montre que l'union d'un ensemble (W, u) -maximal et d'un ensemble de tournées internationales affinement indépendantes qui n'utilise pas le sommet u est un ensemble affinement indépendant.

Lemme 3.3.2 *Soient W un pays et u un sommet appartenant à un pays non dégénéré différent de W . Si $\mathcal{S}_{W,u}$ est un ensemble (W, u) -maximal de tournées internationales et \mathcal{S} est un ensemble de tournées internationales affinement indépendantes qui n'utilisent pas le sommet u , alors l'ensemble $\mathcal{S}_{W,u} \cup \mathcal{S}$ est affinement indépendant.*

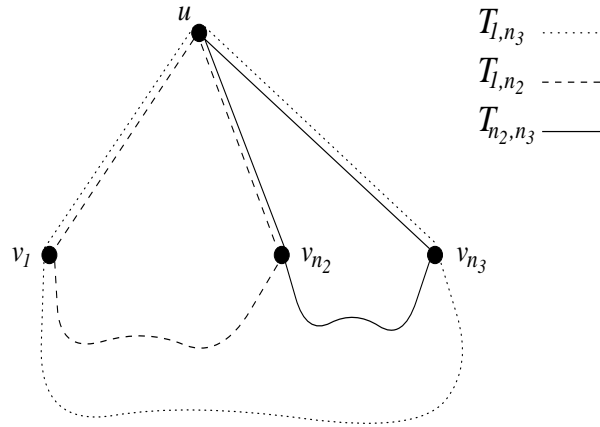


FIG. 3.2 – Trois tournées internationales linéairement indépendantes.

Preuve : Soient $\mathcal{S}_{W,u}$ un ensemble (W, u) -maximal et \mathcal{S} est un ensemble affinement indépendant de tournées internationales qui n'utilisent pas le sommet u . On note c_1 et c_2 les cardinalités respectives de \mathcal{S} et $\mathcal{S}_{W,u}$ et $c = c_1 + c_2$. Soit M la matrice de m lignes et de c colonnes dont les lignes représentent l'ensemble des arêtes et les colonnes représentent les tournées internationales de $\mathcal{S} \cup \mathcal{S}_{W,u}$. On supposera que les c_1 premières colonnes représentent les vecteurs d'incidence des éléments de \mathcal{S} et les c_2 dernières colonnes représentent les vecteurs d'incidence des éléments de $\mathcal{S}_{W,u}$, et que les $|\delta(\{u\})|$ premières lignes représentent les arêtes incidentes au sommet u . La matrice M est de la forme :

$$\begin{pmatrix} 0 & A \\ B & C \end{pmatrix}$$

avec A une sous-matrice carrée d'ordre c_2 . L'ensemble des colonnes de la sous-matrice B est affinement indépendant et, d'après le lemme 3.3.1, la sous-matrice A est de plein rang, alors l'ensemble des colonnes de la matrice M est affinement indépendant. \square

3.3.2 Dimension du polytope ITSP(n, p)

Dans une tournée internationale, chaque pays à un degré égal à deux. Ainsi, le polytope $ITSP(n, p)$ est contenu dans l'intersection des hyperplans associés aux équations de degré. Le rang de la matrice A^* permet de borner

supérieurement la dimension du polytope $ITSP(n, p)$.

Proposition 3.3.1 *Le rang de la matrice d'incidence pays-arête A^* est égal à p .*

La matrice A^* peut être obtenue en considérant la matrice d'incidence sommet-arête du graphe K_p à laquelle on dédouble un certain nombre de fois certaines colonnes. Cette opération donne une matrice de rang égal à celui de la matrice initiale. Ainsi, le rang de la matrice A^* est égal à celui de la matrice d'incidence sommet-arête du graphe K_p , donc à p .

Théorème 3.3.1 $\dim (ITSP(n, p)) = m - p$.

Preuve : La dimension de l'ensemble $P = \{x \in \mathbb{R}^{|E|} : A^*x = 2\mathbb{1}\}$ est égale à $m - p$ puisque le rang de la matrice A^* est égal à p . Le polytope $ITSP(n, p)$ est contenu dans l'ensemble P ; sa dimension est inférieure ou égale à $m - p$. On montre l'inéquation dans l'autre sens par induction sur le nombre n des sommets du graphe avec un nombre fixé de pays. Si $n = p$, alors le polytope $ITSP(n, p)$ est $TSP(p)$. Dans ce cas, la dimension est connue. Supposons que l'inéquation est vérifiée pour $n - 1$ sommets et montrons qu'elle l'est également pour n sommets, avec $n > p$. Soient u un sommet du graphe K_n^p appartenant à un pays non dégénéré W_u et $\mathcal{S}_{W,u}$ un ensemble (W, u) -maximal de tournées internationales associé au sommet u et à un pays W différent de W_u . Le sous-graphe de K_n^p induit par l'ensemble $V \setminus \{u\}$ est le graphe K_{n-1}^p . Par hypothèse de récurrence, il existe un ensemble \mathcal{S} de $m - |\delta(u)| - p + 1$ tournées internationales affinement indépendantes dans le graphe K_{n-1}^p et non incidentes au sommet u . D'après le lemme 3.3.2, l'ensemble $\mathcal{S}_{W,u} \cup \mathcal{S}$ représente un ensemble de $m - p + 1$ tournées internationales affinement indépendantes de K_n^p . \square

Une autre manière de démontrer ce résultat consisterait à montrer que les contraintes de degré sont les seules équations satisfaites par tous les points du polytope $ITSP(n, p)$.

Ainsi, le polytope $ITSP(n, p)$ n'est pas de pleine dimension. Plusieurs inéquations différentes induisent la même facette; ce qui explique une certaine

difficulté dans l'étude de sa structure faciale. Avant de commencer cette étude, on définit la propriété forte d'une facette.

Définition 3.3.1 *Une facette du polytope $ITSP(n, p)$ est dite forte si elle est induite par une inéquation $cx \geq c_0$ telle que, pour tout ensemble légal L la restriction $c^*x \geq c_0$ de $cx \geq c_0$ à l'ensemble L induit une face F de $TSP(p)$ maximale ou non propre; i.e. F est une facette du polytope $TSP(p)$ ou F est le polytope $TSP(p)$.*

Dans la suite, on étudie les inéquations de non négativité et les propriétés que vérifient les facettes qu'elles induisent.

3.3.3 Facettes triviales

Théorème 3.3.2 *Les inéquations $x_e \geq 0$, pour $e \in E$, induisent des facettes du polytope $ITSP(n, p)$.*

Preuve : Soit $e = [u, v]$ une arête quelconque de E . L'inéquation $x_e \geq 0$ est valide pour le polytope $ITSP(n, p)$. Supposons qu'une inéquation $cx \geq c_0$ valide pour $ITSP(n, p)$ domine l'inéquation $x_e \geq 0$, alors il existe une tournée internationale T_0 qui utilise l'arête e telle que $cx^{T_0} = c_0$. Soit L_0 l'ensemble légal engendré par T_0 . L'inéquation $x_e \geq 0$ induit une facette du polytope $TSP(G[L_0]) = TSP(p)$. Soit $c^*x \geq c_0$ la restriction de $cx \geq c_0$ à l'ensemble L_0 . La tournée internationale T_0 sature $c^*x \geq c_0$ ainsi que toute tournée internationale de \mathcal{I}_{L_0} qui n'utilisent pas l'arête e . Pour que $x_e \geq 0$ soit dominée par $cx \geq c_0$ et, ainsi, aboutir à une contradiction, il suffit de montrer que $c^*x \geq c_0$ induit une face propre de $TSP(p)$. Soit T_1 une tournée internationale qui ne sature pas $cx \geq c_0$ telle que $|L_0 \cap L_1|$ est maximum, avec L_1 l'ensemble légal engendré par T_1 . Supposons que L_1 est différent de L_0 . Soit $w_1 \in L_1 \setminus L_0$. On note W le pays qui contient w_1 et $w_0 = L_0 \cap W$. Soient w' et w'' les sommets adjacents à w_1 dans T_1 . L'arête e appartient à T_1 ; sinon T_1 sature $x_e \geq 0$ et, par conséquent, T_1 sature $cx \geq c_0$. De plus, le sommet w_1 n'est pas une extrémité de l'arête e puisqu'elle est utilisée par T_0 et T_1 .

Par ailleurs, soient T' une tournée internationale appartenant à \mathcal{I}_{L_1} telle que les arêtes $[w_1, w']$ et $[w_1, w'']$ appartiennent à T' et l'arête e n'appartient pas à T' et $T'' = T' - [w_1, w'] - [w_1, w''] + [w_0, w'] + [w_0, w'']$. Les tournées

internationales T' et T'' aturent l'inéquation $cx \geq c_0$ puisqu'elles n'utilisent pas l'arête e , alors $c([w_1, w']) + c([w_1, w'']) = c([w_0, w']) + c([w_0, w''])$.

Soit la tournée internationale $T_2 = T_1 - [w_1, w'] - [w_1, w''] + [w_0, w'] + [w_0, w'']$. La tournée internationale T_2 ne sature pas l'inéquation $cx \geq c_0$ puisque $cx^{T_2} = cx^{T_1}$. Si L_2 est l'ensemble légal induit par T_2 , on a $|L_0 \cap L_2| = |L_0 \cap L_1| + 1$. Ceci contredit l'hypothèse que $|L_0 \cap L_1|$ est maximum, alors $L_0 = L_1$ et l'inéquation $c^*x \geq c_0$ induit une face propre de $TSP(p)$. \square

On appellera inéquations triviales les inéquations de non négativité et facettes triviales les facettes qu'elles induisent. Les facettes triviales vérifient la propriété des facettes fortes.

On pourrait développer une autre démonstration, plus directe, qui n'utilise pas le fait que les inéquations triviales induisent des facettes du polytope du TSP . La démonstration serait similaire à celle donnée par Grötschel et Padberg [47] pour montrer que les inéquations triviales induisent des facettes du polytope du TSP .

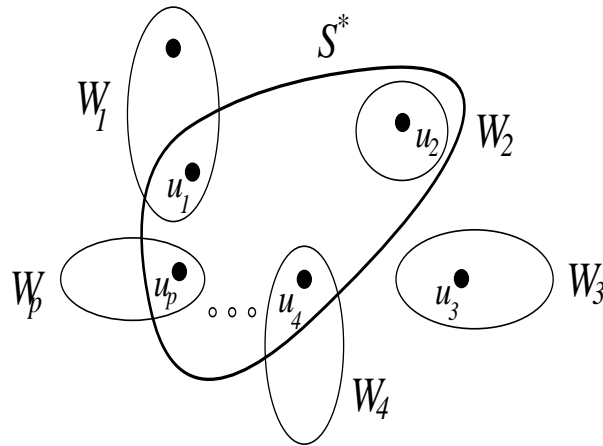


FIG. 3.3 –

On peut penser que les inéquations triviales sont les seules à vérifier la propriété du théorème 3.2.5 comme c'est le cas pour le polytope $P_0(n, p)$. Malheureusement, la réponse est négative. En effet, soit $L = \{u_1, u_2, u_3, \dots, u_p\}$ un ensemble légal, avec $u_i \in W_i$, pour $i = 1, \dots, p$, et supposons que $|W_1| \geq$

2. L'inéquation $x(\delta(S^*)) \geq 2$ associée à l'ensemble $S^* = L \setminus \{u_2, u_3\} \cup W_2$ (voir figure 3.3) est valide pour le polytope $ITSP(n, p)$. De plus, toutes les tournées internationales appartenant à \mathcal{I}_L satisfont $x(\delta(S^*)) \geq 2$. On montrera dans la section 3.3.6.1 que l'inéquation de coupe $x(\delta(S^*)) \geq 2$ induit une facette du polytope $ITSP(n, p)$.

3.3.4 Relation avec le polyèdre graphique

Le polytope $ITSP(n, p)$ n'est pas de pleine dimension ; des inéquations différentes peuvent induire une même facette. Soit $cx \geq c_0$ une inéquation induisant une facette F du polytope $ITSP(n, p)$, toutes les inéquations $fx \geq f_0$ qui induisent la même facette F , i.e. équivalentes à $cx \geq c_0$, sont données par les relations $f = \lambda A^* + kc$ et $f_0 = 2\lambda \mathbb{1} + kc_0$, avec A^* la matrice d'incidence sommet-pays, λ un vecteur ligne de \mathbb{R}^p et k un réel.

Étant donnée une facette F d'un polytope qui n'est pas de pleine dimension, une question que l'on peut se poser est la suivante : quelle inéquation (ou forme d'inéquation) induisant la facette F faut-il considérer ? Souvent, le choix d'une forme d'inéquation est étroitement associé à une relaxation du polytope considéré.

Une forme particulière d'inéquation peut être également utilisée comme un standard d'écriture lorsqu'elle est unique. Dans ce cas, deux inéquations induisant des facettes d'un polytope sont équivalentes si les écritures standards des deux inéquations sont identiques modulo une multiplication des deux membres des inéquations par un scalaire positif.

Pour le cas du TSP , de nombreux résultats ont été établis en considérant des formes particulières d'inéquations induisant des facettes. Les formes d'inéquations les plus connues sont la forme triangulaire serrée [69] et la forme normale [81] associées respectivement à la relaxation graphique et à la monotonisation du polytope du TSP .

Dans ce qui suit, on s'intéresse à la forme triangulaire serrée des inéquations induisant des facettes du polytope $ITSP(n, p)$ et la relation qui existe entre celui-ci et le polytope $P_0(n, p)$.

3.3.4.1 Forme triangulaire serrée

Le résultat suivant suggère une procédure pour obtenir la forme triangulaire serrée d'une inéquation induisant une facette de $ITSP(n, p)$.

Théorème 3.3.3 *Soit $cx \geq c_0$ une inéquation induisant une facette du polytope $ITSP(n, p)$. Une inéquation $fx \geq f_0$ équivalente à $cx \geq c_0$ est FTS si et seulement si $f = \lambda A^* + kc$ et $f_0 = 2\lambda \mathbb{I} + kc_0$, où k est un réel positif et λ un vecteur de \mathbb{R}^p tel que, pour un pays W_u , on a :*

$$\lambda_{W_u} = \frac{k}{2} (\max \{c(v, w) - c(u, v) - c(u, w)\}), \quad (3.3)$$

pour u, v et w des sommets appartenant respectivement aux pays W_u, W_v et W_w deux à deux différents.

Preuve : L'inéquation $fx \geq f_0$ est FTS si λ et k vérifient la condition suffisante 3.3. Supposons que $fx \geq f_0$ est une inéquation FTS . Les inéquations $fx \geq f_0$ et $cx \geq c_0$ sont équivalentes, alors il existe un vecteur $\lambda \in \mathbb{R}^p$ et un réel k tels que $f = \lambda A^* + kc$ et $f_0 = 2\lambda \mathbb{I} + kc_0$. L'inéquation $fx \geq f_0$ est triangulaire, alors pour tous sommets u, v et w appartenant respectivement à des pays W_u, W_v et W_w deux à deux différents, on a :

$$f(u, v) \geq f(u, w) + f(v, w),$$

et en remplaçant les composantes de f par leurs valeurs dans la relation $f = \lambda A^* + kc$, on obtient :

$$\lambda_{W_u} \geq \frac{k}{2} (c(v, w) - c(u, v) - c(u, w)).$$

L'inéquation $fx \geq f_0$ est triangulaire serrée, alors, pour tout pays W_u , la relation précédente est vérifiée avec égalité pour au moins un triplet (u, v, w) de sommets appartenant respectivement au pays W_u et à des pays W_v et W_w deux à deux différents. \square

Ainsi, pour trouver l'écriture FTS d'une inéquation induisant une facette du polytope $ITSP(n, p)$, il suffit de fixer la constante k , par exemple à 1, puis calculer le vecteur λ en utilisant la relation 3.3. La complexité d'une telle procédure est $O(n^3)$.

3.3.4.2 Relation avec $P_0(n, p)$

Théorème 3.3.4 *Soit $cx \geq c_0$ une inéquation FTS. Si $cx \geq c_0$ induit une facette non triviale de $ITSP(n, p)$, alors elle induit également une facette de $P_0(n, p)$.*

Preuve : Si $n = p$, le polytope $ITSP(n, p)$ est exactement le polytope $TSP(n)$. Dans ce cas, Naddef et Rinaldi [69] ont montré que le résultat est vrai.

Supposons que $n > p$. Soit \mathcal{I}' un ensemble de $|E| - p$ tournées internationales linéairement indépendantes appartenant à \mathcal{I}_c^- . On construit un ensemble \mathcal{T}' de p tours internationaux de la manière suivante. Pour chaque pays W_i , $i = 1, \dots, p$, considérons une arête $e = [u, v] \in \Delta_c(W_i)$. On note w un sommet de W_i tel que $c([u, v]) = c([u, w]) + c([v, w])$. Soit I_e une tournée internationale appartenant à \mathcal{I}_c^- qui utilise l'arête e ; une telle tournée existe sinon l'inéquation $cx \geq c_0$ est l'inéquation triviale associée à l'arête e . On définit le tour international $T_i = I_e - [u, v] + [u, w] + [v, w]$. On a $cx^{T_i} = cx^{I_e}$, alors T_i appartient à \mathcal{I}_c^- . L'ensemble $\mathcal{I}' \cup \mathcal{T}'$ est un ensemble de $|E|$ tours internationaux linéairement indépendants. En effet, supposons qu'un tour international de $\mathcal{I}' \cup \mathcal{T}'$ soit une combinaison linéaire des autres tours internationaux de $\mathcal{I}' \cup \mathcal{T}'$. Deux cas peuvent se présenter ;

1^{er} cas : un tour international de \mathcal{T}' , disons T_1 , est une combinaison linéaire des autres éléments de \mathcal{I}' et \mathcal{T}' ; i.e

$$T_1 = \sum_{i=1}^{|E|-p} \lambda_i I_i + \sum_{j=2}^p \mu_j T_j, \quad (3.4)$$

avec λ_i et μ_j des réels pour $i = 1, \dots, |E| - p$ et $j = 2, \dots, p$.

En additionnant les lignes associées aux arêtes incidentes au pays W_1 dans la relation 3.4, on obtient la relation

$$\sum_{i=1}^{|E|-p} \lambda_i + \sum_{j=2}^p \mu_j = 2. \quad (3.5)$$

Si on effectue la même opération en considérant un pays W_k différent de W_1 , on obtient la relation

$$\sum_{i=1}^{|E|-p} \lambda_i + \sum_{j=2}^p \mu_j + \mu_k = 1, \quad \text{pour } k = 2, \dots, p. \quad (3.6)$$

En soustrayant membre à membre les relations 3.5 et 3.6, on trouve que, pour $k = 2, \dots, p$, $\mu_k = -1$ et, par conséquent,

$$\sum_{k=2}^p \mu_k = -p + 1.$$

Si on additionne toutes les lignes dans la relation 3.4, on obtient la relation

$$n + 1 = n \left(\sum_{i=1}^{|E|-p} \lambda_i + \sum_{j=2}^p \mu_j \right) + \sum_{j=2}^p \mu_j$$

Or $\sum_{i=1}^{|E|-p} \lambda_i + \sum_{j=2}^p \mu_j = 2$ et $\sum_{j=2}^p \mu_j = -p + 1$, ce qui nous donne $n = p$.

Ceci contredit l'hypothèse initiale que $n > p$.

2^{ème} cas : une tournée internationale de \mathcal{I}' est une combinaison linéaire des autres éléments de \mathcal{I}' et \mathcal{T}' . Par hypothèse, \mathcal{I}' est un ensemble de tournées internationales linéairements indépendantes, alors au moins un élément de \mathcal{T}' a un coefficient non nul dans cette combinaison linéaire. Ceci nous ramène au premier cas.

L'ensemble $\mathcal{I}' \cup \mathcal{T}'$ est un sous-ensemble de $\mathcal{T}_c^=$ de $|E|$ tours internationaux linéairements indépendants. La face induite par l'inéquation $cx \geq c_0$ est de dimension $|E| - 1$; l'inéquation $cx \geq c_0$ induit une facette de $P_0(n, p)$. \square

Cette dernière preuve montre également que l'écriture FTS d'une inéquation induisant une facette du $ITSP(n, p)$ est unique. Ainsi, étant donnés deux inéquations induisant des facettes du polytope $ITSP(n, p)$, on peut vérifier en $O(n^3)$ si elles sont équivalentes ou non.

Dans le cas du TSP , on peut, évidemment, résoudre le problème de la même manière en $O(n^3)$. Queyranne et Wang [81] améliorent la complexité du problème en considérant la forme normale comme écriture standard des inéquations ; ils proposent une procédure en $O(m)$.

Dans le cas du *ITSP*, la définition de la forme normale d'une inéquation est la suivante.

Définition 3.3.2 *Une inéquation $cx \geq c_0$ induisant une facette du polytope $ITSP(n, p)$ est dite sous forme normale par rapport à un sommet h appartenant à un pays W_h , ou sous forme h -normale, si elle vérifie les conditions suivantes :*

- (i) $c_e = 0$ pour toute arête $e \in \delta(\{h\})$;
- (ii) $c \geq 0$ et il existe une arête $e \in E(V \setminus \{h\})$ telle que $c_e = 0$;
- (iii) $\min \{c_e > 0 : e \in E(V)\} = 1$.

Étant donné une facette F de *ITSP* (n, p) et un sommet h , une inégalité sous forme h -normale unique induit la facette F . La procédure en $O(m)$ décrite par Queyranne et Wang, dans [81], pour trouver l'écriture h -normale reste valide pour le cas du *ITSP*.

Le résultat suivant constitue la généralisation pour le cas du *ITSP* de la relation fondamentale qui existe entre les polytopes du *TSP* et du *GTSP*.

Théorème 3.3.5 *Toute facette non triviale de $ITSP(n, p)$ est contenue dans exactement $p + 1$ facettes de $P_0(n, p)$ dont p facettes sont les facettes induites par les inéquations de degré.*

Preuve : Soit F une facette non triviale du polytope *ITSP* (n, p) . La facette F est contenue dans chacune des p facettes du polytope $P_0(n, p)$ induites par les inéquations de degré. Soit $fx \geq f_0$ l'inéquation *F**T**S* unique qui induit la facette F . D'après le théorème 3.3.4, l'inéquation $fx \geq f_0$ induit une facette du polytope $P_0(n, p)$ qu'on notera F_1 . La facette F est contenue dans F_1 . Supposons qu'une facette F_2 de $P_0(n, p)$ différente de F_1 et des facettes induites par les inéquations de degré contient F . La facette F_2 est induite par une inéquation *F**T**S* $gx \geq g_0$ différente de $fx \geq f_0$. Les deux inéquations $fx \geq f_0$ et $gx \geq g_0$ sont *F**T**S* et induisent la même facette F de *ITSP* (n, p) . Ceci contredit le fait qu'une facette de *ITSP* (n, p) est induite par une inéquation *F**T**S* unique modulo une multiplication des deux membres de l'inéquation par un scalaire positif. \square

Dans la suite, on s'intéressera aux facettes non triviales du polytope $ITSP(n, p)$. Chacune des ces facettes peut être induite par une inéquation FTS . On distinguera deux classes de facettes : les facettes régulières et les facettes non régulières.

3.3.5 Facettes régulières

Une inéquation $cx \geq c_0$ est dite régulière si, pour toute paire de pays distincts W_i et W_j , les coefficients des arêtes de $(W_i : W_j)$ sont égaux.

Une facette est dite régulière si elle est induite par une inéquation régulière. De plus, toutes les inéquations qui induisent une facette régulière sont régulières.

Cette classe de facettes est étroitement liée à la structure faciale du polytope du TSP . Avant de présenter cette relation, on étudie les conditions dans lesquelles l'inéquation

$$\sum_{u \in W_u} \sum_{v \in W_v} x_{u,v} \leq 1 \quad (3.7)$$

induit une facette de $ITSP(n, p)$, pour W_u et W_v deux pays distincts.

Théorème 3.3.6 *Soient W_u et W_v deux pays différents. L'inéquation 3.7 induit une facette du polytope $ITSP(n, p)$ si et seulement si W_u et W_v sont deux pays dégénérés.*

Preuve : Supposons qu'un des deux pays soit non dégénéré, disons W_v , alors l'inéquation $x(\delta(W_u \cup \{v\})) \geq 2$, avec $v \in W_v$, est valide pour le polytope $ITSP(n, p)$ et elle domine l'inéquation 3.7.

Réciproquement, on montre la propriété par induction sur le nombre de sommets avec un nombre fixé de pays. Soit K_n^p un graphe multiparti complet. Si $n = p$, alors les polytopes $ITSP(n, p)$ et $TSP(p)$ sont identiques. Dans ce cas, Grötschel et Padberg [47] ont montré que l'inéquation 3.7 induit une facette du polytope $TSP(p)$, et par conséquent une facette du polytope $ITSP(n, p)$. Supposons que $n > p$. Soient un pays non dégénéré, disons W_p , et w un sommet appartenant à W_p . Le sous-graphe de K_n^p induit par l'ensemble $V \setminus \{w\}$ est le graphe K_{n-1}^p . Par hypothèse de récurrence, il existe

un ensemble \mathcal{S} de $m - |\delta(w)| - p$ tournées internationales affinements indépendantes dans le graphe K_n^p et non incidentes au sommet u qui saturent l'inéquation 3.7. Soit $\mathcal{S}_{W,w}$ un ensemble (W, w) -maximal de tournées internationales associé au sommet w et à un pays W différent de W_p, W_u et W_v . Si on suppose que W, W_u et W_v sont respectivement W_1, W_{p-2} et W_{p-1} , alors on peut choisir l'ensemble $\mathcal{S}_{W,w}$ de manière à ce que tous ses éléments saturent l'inéquation 3.7. D'après le lemme 3.3.2, l'ensemble $\mathcal{S}_{W,w} \cup \mathcal{S}$ représente un ensemble de $m - p$ tournées internationales affinements indépendantes de K_n^p . De plus, les tournées internationales de $\mathcal{S}_{W,w} \cup \mathcal{S}$ saturent l'inéquation 3.7. \square

L'inéquation 3.7 est équivalente à l'inéquation $x(\delta(S)) \geq 2$, avec $S = W_u \cup W_v$. Elle constitue la seule exception du résultat suivant donnant la relation principale qui existe entre les polytopes $ITSP(n, p)$ et $TSP(p)$.

Théorème 3.3.7 *Soient $cx \geq c_0$ une inéquation régulière valide pour le polytope $ITSP(n, p)$ qui n'est pas équivalente à l'inéquation 3.7 et $c^*y \geq c_0$ la restriction de $cx \geq c_0$ à un ensemble légal.*

*L'inéquation $cx \geq c_0$ induit une facette du polytope $ITSP(n, p)$ si et seulement si $c^*y \geq c_0$ induit une facette non triviale du polytope $TSP(p)$.*

Preuve : Si l'inéquation $cx \geq c_0$ induit une facette de $ITSP(n, p)$, alors la restriction $c^*y \geq c_0$ de $cx \geq c_0$ à un ensemble légal induit également une facette du polytope $TSP(p)$.

Réciproquement, supposons que l'inéquation $cx \geq c_0$ n'induit pas une facette de $ITSP(n, p)$, alors il existe une inéquation $bx \geq b_0$ valide et propre pour le polytope $ITSP(n, p)$ qui domine $cx \geq c_0$. L'inéquation $bx \geq b_0$ n'est pas nécessairement régulière. Soit T_0 une tournée internationale telle que $bx^{T_0} = b_0$ et $cx^{T_0} > c_0$. L'inéquation $c^*y \geq c_0$ est la restriction de $cx \geq c_0$ à l'ensemble légal L^{T_0} induit par T_0 . Soit $b^*y \geq b_0$ la restriction de l'inéquation $bx \geq b_0$ à l'ensemble L^{T_0} . L'inéquation $b^*y \geq b_0$ est valide pour le polytope $TSP(p)$. Toute tournée internationale appartenant à $\mathcal{I}_{L^{T_0}}$ qui sature $c^*y \geq c_0$ sature également $b^*y \geq b_0$ et, de plus, $b^*y^{T_0} = b_0$ et $c^*y^{T_0} > c_0$. Il suffit de montrer que $b^*y \geq b_0$ est propre pour qu'elle domine $c^*y \geq c_0$ et, ainsi, aboutir à une contradiction. Supposons que $b^*y \geq b_0$ n'est pas une inéquation propre ; i.e. $b^*y^T = b_0$, pour toute tournée internationale T appartenant à $\mathcal{I}_{L^{T_0}}$. Alors il existe un vecteur $\lambda \in \mathbb{R}^p$ tel que $b_{i,j}^* = \lambda_i + \lambda_j$

et $b_0 = 2 \sum_{i=1}^p \lambda_i$ (combinaison linéaire des équations de degré). Sans perte de généralité, on peut supposer que λ est le vecteur nul, ce qui annule b_0 et toutes les composantes du vecteur b^* .

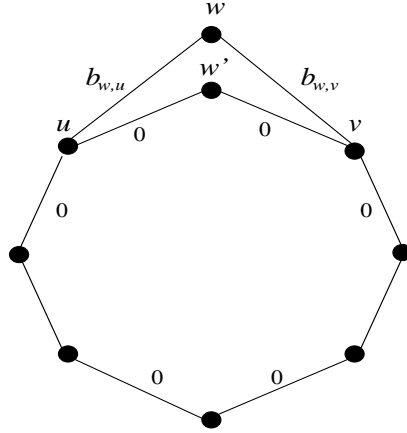


FIG. 3.4 –

Soit $e = [w, u]$ une arête quelconque telle que $w \notin L^{T_0}$ et $u \in L^{T_0}$. On note w' le sommet de L^{T_0} appartenant au même pays que celui de w . L'inéquation $c^*y \geq c_0$ n'est pas triviale, alors il existe une tournée internationale T_e qui utilise l'arête e telle que $cx^{T_e} = c_0$. L'inéquation $cx \geq c_0$ est régulière alors on peut supposer que l'ensemble légal L^{T_e} induit par la tournée internationale T_e est égal à $L^{T_0} \setminus \{w'\} \cup \{w\}$. Soit v le voisin de w dans T_e différent de u . La tournée internationale T_e sature également l'inéquation $bx \geq b_0$, alors $b_{w,v} = -b_{w,u}$. Soit $f = [w, z]$ une arête quelconque avec $z \in L^{T_e} \setminus \{u, v\}$. On a que $b_{w,z} \geq 0$ puisque $b_{w,z} + b_{w,u} \geq 0$ et $b_{w,z} - b_{w,u} \geq 0$. Soit T_f^u une tournée internationale telle que l'ensemble légal correspondant $L^{T_f^u}$ est égal à L^{T_e} , $cx^{T_f^u} = c_0$ et T_f^u utilisent les deux arêtes $e = [w, u]$ et $f = [w, z]$. Une telle tournée internationale existe sinon l'inéquation $x_e + x_f \leq 1$ valide pour $TSP(p)$ domine l'inéquation $c^*y \geq c_0$. La tournée internationale T_f^u sature également l'inégalité $bx \geq b_0$, alors on a $b_{w,u} = b_{w,z}$. De la même manière, on considère une tournée internationale T_f^v telle que l'ensemble légal correspondant $L^{T_f^v}$ est égal à L^{T_e} , $cx^{T_f^v} = c_0$ et T_f^v utilisent les deux arêtes $[w, u]$ et $f = [w, z]$. La tournée internationale T_f^v sature également l'inégalité $bx \geq b_0$, alors on a $b_{w,z} = b_{w,v} = -b_{w,u}$. Par conséquent, les coefficients $b_{w,u}$,

$b_{w,v}$ et $b_{w,z}$ sont nuls. En conclusion, l'ensemble des arêtes ayant une extrémité dans L^{T_0} et l'autre extrémité à l'extérieur de L^{T_0} ont des coefficients nuls dans l'inéquation $bx \geq b_0$.

Considérons une arête quelconque $e = [w_1, w_2]$ telle que w_1 et w_2 n'appartiennent pas à L^{T_0} . On notera w'_1 et w'_2 les sommets de L^{T_0} qui appartiennent respectivement au pays de w_1 et au pays de w_2 . Soit T_e une tournée internationale telle que $bx^{T_e} = b_0$, $e \in T_e$ et $L^{T_e} = L^{T_0} \setminus \{w'_1, w'_2\} \cup \{w_1, w_2\}$. Les arêtes de T_e , différentes de l'arête e , ont un coefficient nul dans l'inéquation $bx \geq b_0$, alors le coefficient $b_e = 0$.

Ainsi, le vecteur b est le vecteur nul et, par conséquent, $bx \geq b_0$ est une inéquation valide et non propre de $ITSP(n, p)$. Ceci contredit l'hypothèse que $bx \geq b_0$ est une inéquation valide et propre de $ITSP(n, p)$. L'inéquation $b^*y \geq b_0$ est valide et propre de $TSP(n)$. \square

Ainsi, toute inéquation régulière $cx \geq c_0$ induisant une facette du polytope $ITSP(n, p)$ correspond à une inéquation induisant une facette de $TSP(p)$ et déterminée par la restriction de $cx \geq c_0$ à un ensemble légal. Les facettes régulières sont des facettes fortes. L'étude de la classe des facettes régulières du polytope $ITSP(n, p)$ se ramène à l'étude du polytope $TSP(p)$. Tous les résultats connus sur la description linéaire du polytope du TSP sont valides pour le polytope du $ITSP$.

Dans la section 3.3.6, on présentera des inéquations non régulières induisant des facettes de $ITSP(n, p)$ qui généralisent certaines inéquations connues induisant des facettes du polytope du TSP ou, d'une certaine manière, des facettes régulières du polytope $ITSP(n, p)$. Un état de l'art de la description linéaire du polytope du TSP en inéquations induisant des facettes connues est dressé par Naddef dans [67]. Dans ce qui suit, on citera les principales inéquations connues induisant des facettes du polytope $TSP(n)$.

3.3.5.1 Les inéquations d'élimination de sous-tours

Grötschel et Padberg [48] ont montré que l'inéquation 3.8 induit une facette du polytope $TSP(n)$ pour $2 \leq |S| \leq n - 2$ et $n \geq 4$.

$$x(\delta(S)) \geq 2. \tag{3.8}$$

L'inéquation 3.8 est appelée inéquation d'élimination de sous-tour ou inéquation de coupe.

3.3.5.2 Les inéquations de peignes

Soient H et T_i , pour $i = 1, \dots, t$, des sous-ensembles de sommets. On notera $\mathcal{T} = \{T_1, \dots, T_t\}$ et on appellera le sous-ensemble H un manche et un élément de \mathcal{T} une dent. Grötschel et Padberg [48] ont montré que l'inéquation

$$x(\delta(H)) + \sum_{i=1}^t x(\delta(T_i)) \geq 3t + 1 \quad (3.9)$$

induit une facette du polytope $TSP(n)$, pour $n \geq 6$, si les conditions suivantes sont satisfaites :

- (a) $|T_i \cap H| \geq 1$, pour $i = 1, \dots, t$;
- (b) $|T_i \setminus H| \geq 1$, pour $i = 1, \dots, t$;
- (c) $T_i \cap T_j = \emptyset$, pour $1 \leq i < j \leq t$;
- (d) $t \geq 3$ et impair.

Le couple (H, \mathcal{T}) est dit peigne et l'inéquation 3.9 est dite inéquation de peigne. Dans la cas particulier où $|T_i| = 2$, pour $i = 1, \dots, t$, l'inéquation est dite inéquation de 2-couplage.

3.3.5.3 Les inéquations d'arbres de cliques

Soient H_j , pour $j = 1, \dots, h$ et T_i , pour $i = 1, \dots, t$ des sous-ensembles de sommets. On notera $\mathcal{H} = \{H_1, \dots, H_h\}$ et $\mathcal{T} = \{T_1, \dots, T_t\}$ et on appellera un élément de \mathcal{H} un manche et un élément de \mathcal{T} une dent. Pour toute dent T_i , on désigne par t_i le nombre de manches dont l'intersection avec T_i est non vide. Un arbre de cliques, noté $(\mathcal{H}, \mathcal{T})$, est un graphe partiel de K_n dont les cliques maximales sont les manches de \mathcal{H} et les dents de \mathcal{T} . Grötschel et Pulleyblank [49] ont montré que l'inéquation

$$\sum_{j=1}^h x(\delta(H_j)) + \sum_{i=1}^t x(\delta(T_i)) \geq \sum_{i=1}^t (2t_i + 1) + 1 \quad (3.10)$$

induit une facette du polytope $TSP(n)$, pour $n \geq 11$, si les conditions suivantes sont satisfaites :

- (a) $T_i \cap T_j = \emptyset$, pour $1 \leq i < j \leq t$;
- (a) $H_i \cap H_j = \emptyset$, pour $1 \leq i < j \leq h$;
- (c) $2 \leq |T_i| \leq n - 2$ et $T_i \setminus \bigcup_{j=1}^h H_j \neq \emptyset$, pour $i = 1, \dots, t$;
- (d) pour chaque manche H_i le nombre de dents qui l'intersectent est $2k_i + 1$, avec $k_i \geq 1$;
- (e) Si une dent T et un manche H admettent une intersection non vide, alors $|T \cap H|$ est un ensemble déconnectant de l'arbre de cliques $(\mathcal{H}, \mathcal{T})$.

L'inéquation 3.10 est dite inéquation d'arbre de cliques. Elle constitue une généralisation de l'inéquation de peigne.

3.3.6 Facettes non régulières

La classe des facettes non régulières présente des difficultés particulières dues à la nature des inéquations qui les induisent. Les premières facettes non régulières sont les facettes triviales.

On rappelle qu'un sous-ensemble S de sommets est dit valide s'il existe un pays contenu dans S et un autre dans $V \setminus S$, et il est dit non régulier s'il existe un pays qui rencontre à la fois S et $V \setminus S$. Un ensemble S est dit minimal si $S = W \cup \{u\}$, avec W un pays et u un sommet appartenant à un pays non dégénéré et différent de W . Dans la suite, on s'intéressera à la généralisation des sous-tours et des arbres de cliques pour le cas non régulier.

3.3.6.1 Les sous-tours non réguliers

L'inéquation 3.11 est dite inéquation d'élimination de sous-tour non régulière ou inéquation de coupe non régulière si l'ensemble S correspondant est un ensemble non régulier de sommets.

$$x(\delta(S)) \geq 2 \tag{3.11}$$

Théorème 3.3.8 *L'inéquation 3.11 est une inéquation valide pour le polytope $ITSP(n, p)$ si et seulement si S est un ensemble valide.*

Preuve : Soit S un ensemble valide. Toute tournée internationale T admet au moins deux arêtes dans $\delta(S)$, alors $x^T(\delta(S)) \geq 2$.

Réciproquement, supposons que S n'est pas un ensemble valide. Alors, il existe un ensemble légal L^* tel que $S \cap L^* = \emptyset$. Soit T^* une tournée internationale dont l'ensemble légal induit est L^* . Il n'existe pas d'arête de T^* appartenant à $\delta(S)$, alors $x^{T^*}(\delta(S)) = 0$. \square

Théorème 3.3.9 *Soit S un ensemble valide dont l'intersection est non vide avec exactement deux pays; i.e. $p - \eta(S) = 2$. L'inéquation 3.11 induit une facette du polytope $ITSP(n, p)$ si et seulement si S est un ensemble minimal ou S est l'union de 2 pays dégénérés.*

Preuve : Soit S un ensemble valide dont l'intersection est non vide avec exactement deux pays. Si S est régulier, alors, d'après le théorème 3.3.6, S est l'union de 2 pays dégénérés. Pour le cas où S est non régulier, on pose $S = W_1 \cup S'$, avec W_1 un pays et S' un ensemble strictement inclus dans un pays W_2 différent de W_1 . Supposons que S n'est pas un ensemble minimal, alors $|S'| \geq 2$. Soient $u \in S'$ et $S^* = W_1 \cup \{u\}$. L'ensemble S^* est valide, donc l'inéquation $x(\delta(S^*)) \geq 2$ est valide pour $ITSP(n, p)$. Soit T une tournée internationale telle que $x^T(\delta(S)) = 2$. Si T ne visite pas le sommet u , alors $x^T(\delta(S^*)) = x^T(\delta(W_1)) = 2$ et si T visite le sommet u , alors $x^T(\delta(S^*)) = x^T(\delta(S)) = 2$. Considérons une tournée internationale T^* qui visite un sommet de S' différent de u et qui n'utilise pas une arête appartenant à $E(S)$. Une telle tournée internationale existe et sature l'inéquation $x(\delta(S^*)) \geq 2$ mais pas l'inéquation $x(\delta(S)) \geq 2$. L'inéquation $x(\delta(S^*)) \geq 2$ est valide pour $ITSP(n, p)$ et elle domine l'inéquation $x(\delta(S)) \geq 2$. Ceci contredit l'hypothèse que cette dernière inéquation induit une facette de $ITSP(n, p)$, alors l'ensemble S est minimal.

Réciproquement, soit S un ensemble minimal. On peut écrire $S = W_1 \cup \{u\}$, avec W_1 un pays et u un sommet appartenant à un pays W_2 non dégénéré et différent de W_1 . Toute tournée internationale qui ne visite pas le sommet u sature l'inéquation $x(\delta(S)) \geq 2$. Le sous-graphe de K_n^p engendré par l'ensemble $V \setminus \{u\}$ est K_{n-1}^p et le polytope du $ITSP$ qui lui est associé est de pleine dimension, alors il existe un ensemble \mathcal{S} de $m - |\delta(\{u\})| - p + 1$

tournées internationales affinements indépendantes de K_n^p qui saturent l'inéquation $x(\delta(S)) \geq 2$ et qui n'utilisent pas le sommet u . Soit $\mathcal{S}_{W_1, u}$ un ensemble (W_1, u) -maximal. Toutes les tournées internationales de $\mathcal{S}_{W_1, u}$ saturent l'inéquation $x(\delta(S)) \geq 2$ sauf une seule. D'après le lemme 3.3.2, l'ensemble $\mathcal{S}_{W_1, u} \cup \mathcal{S}$ représente un ensemble de $m - p + 1$ tournées internationales affinements indépendantes de K_n^p et $m - p$ tournées internationales de $\mathcal{S}_{W_1, u} \cup \mathcal{S}$ saturent l'inéquation $x(\delta(S)) \geq 2$.

Le cas où l'ensemble S est l'union de 2 pays dégénérés a été établi dans le théorème 3.3.6. \square

Ainsi, l'inéquation de coupe non régulière associée à un ensemble S valide qui rencontre exactement deux pays n'induit une facette de $ITSP(n, p)$ que lorsque S est un ensemble minimal. Le résultat suivant montre que si S et $V \setminus S$ sont valides et rencontrent chacun plus de deux pays, alors l'inéquation de coupe non régulière associée à S induit une facette de $ITSP(n, p)$.

Théorème 3.3.10 *Si S est un ensemble valide tel que $p - \eta(S) \geq 3$ et $p - \eta(V \setminus S) \geq 3$, alors l'inéquation $x(\delta(S)) \geq 2$ induit une facette du polytope $ITSP(n, p)$.*

Preuve : On montre le résultat par induction sur le nombre de sommets sachant un nombre de pays fixé. Il est clair que le résultat est vérifié lorsque $n = p$. Supposons que $n > p$ et soit S un ensemble valide qui vérifie les hypothèses du théorème. et considérons le cas où S est un ensemble non régulier. On note W_1 un pays inclus dans S . Soit u un sommet de S appartenant à un pays non dégénéré et différent de W_1 et considérons l'ensemble valide $S_1 = S \setminus \{u\}$. Le sous-graphe de K_n^p engendré par l'ensemble $V \setminus \{u\}$ est K_{n-1}^p . On distingue deux cas :

1^{er} cas : S_1 vérifie l'hypothèse du théorème. Par hypothèse de récurrence, il existe un ensemble \mathcal{S} de $m - |\delta(\{u\})| - p$ tournées internationales affinements indépendantes qui saturent l'inéquation $x(\delta(S)) \geq 2$ et qui n'utilisent pas le sommet u . Soit $\mathcal{S}_{W_1, u}$ un ensemble (W_1, u) -maximal tel que toutes les tournées internationales de $\mathcal{S}_{W_1, u}$ saturent l'inéquation $x(\delta(S)) \geq 2$. Un tel ensemble existe puisque $p - \eta(S) \geq 3$. D'après le lemme 3.3.2, l'ensemble $\mathcal{S}_{W_1, u} \cup \mathcal{S}$ représente un ensemble de $m - p$ tournées internationales affinements indépendantes de K_n^p .

2^{ème} cas : S_1 ne vérifie pas l'hypothèse du théorème. Si S_1 est un ensemble minimal ou l'union de deux pays dégénérés, alors l'inégalité $x(\delta(S_1)) \geq 2$ induit une facette de $ITSP(n-1, p)$, et, par conséquent, il existe un ensemble \mathcal{S} de $m - |\delta(\{u\})| - p$ tournées internationales affinements indépendantes qui saturent l'inéquation $x(\delta(S)) \geq 2$ et qui n'utilisent pas le sommet u . Soit $\mathcal{S}_{W_1, u}$ un ensemble (W_1, u) -maximal tel que toutes les tournées internationales de $\mathcal{S}_{W_1, u}$ saturent l'inéquation $x(\delta(S)) \geq 2$. Un tel ensemble existe puisque $p - \eta(S) \geq 3$. D'après le lemme 3.3.2, l'ensemble $\mathcal{S}_{W_1, u} \cup \mathcal{S}$ représente un ensemble de $m - p$ tournées internationales affinements indépendantes de K_n^p .

Considérons le cas où S_1 n'est ni minimal, ni l'union de deux pays dégénérés. Si S_1 est un ensemble régulier, alors S_1 est l'union de W_1 et d'un autre pays W_2 dont au moins un des deux pays est non dégénéré. Supposons que W_2 est non dégénéré. Soient v un sommet appartenant à W_2 et $S_2 = S \setminus \{v\}$. L'ensemble S_2 vérifie l'hypothèse du théorème ou est un ensemble minimal. Dans les deux cas, il existe un ensemble \mathcal{S} de $m - |\delta(\{v\})| - p$ tournées internationales qui saturent l'inéquation $x(\delta(S)) \geq 2$ et qui n'utilisent pas le sommet v . Pour le cas où S_1 est un ensemble non régulier, on considère un sommet v appartenant à S différent de u et appartenant à un pays différent de W_1 et l'ensemble $S_2 = S \setminus \{v\}$. L'ensemble S_2 vérifie l'hypothèse du théorème ou est un ensemble minimal. Dans les deux cas, il existe un ensemble \mathcal{S} de $m - |\delta(\{v\})| - p$ tournées internationales affinements indépendantes qui saturent l'inéquation $x(\delta(S)) \geq 2$ et qui n'utilisent pas le sommet v . Soit $\mathcal{S}_{W_1, v}$ un ensemble (W_1, v) -maximal tel que toutes les tournées internationales de $\mathcal{S}_{W_1, v}$ saturent l'inéquation $x(\delta(S)) \geq 2$. Un tel ensemble existe puisque $p - \eta(S) \geq 3$. D'après le lemme 3.3.2, l'ensemble $\mathcal{S}_{W_1, v} \cup \mathcal{S}$ représente un ensemble de $m - p$ tournées internationales affinements indépendantes de K_n^p .

Si S est un ensemble régulier, d'après le théorème 3.3.7, l'inéquation $x(\delta(S)) \geq 2$ induit une facette de $ITSP(n, p)$. \square

La condition $p - \eta(V \setminus S) \geq 3$ du théorème précédent n'apparaît pas explicitement comme une condition nécessaire. Lorsque l'ensemble S est régulier, la condition est nécessaire pour l'application du théorème 3.3.7 dans la preuve.

Finalement, l'inéquation $x(\delta(S)) \geq 2$ induit une facette du polytope $ITSP(n, p)$, dite facette de sous-tour, si et seulement si l'ensemble S est valide et vérifie l'une des conditions suivantes :

- (i) S et $V \setminus S$ rencontrent chacun plus de deux pays ;
- (ii) S est un ensemble minimal ;
- (iii) $S = W_1 \cup W_2$, avec W_1 et W_2 deux pays dégénérés.

On notera que, dans [27], on trouve des résultats similaires sur les inéquations de sous-tours exprimées dans l'espace des arêtes et des sommets avec des preuves différentes.

On notera également que, pour un sous-ensemble S valide, l'inéquation

$$x(E(S)) + x(E(V \setminus S)) \leq p - 2 \quad (3.12)$$

est valide pour le polytope $ITSP(n, p)$ et elle est équivalente à l'inéquation de coupe $x(\delta(S)) \geq 2$.

Les facettes de sous-tours sont des facettes fortes et toutes les facettes de $ITSP(n, p)$ qu'on a présentées jusque là sont également des facettes fortes. Dans la suite, on distinguera une sous-classe particulière de la classe des facettes fortes. Une facette forte est dite *tsp*-forte si elle est induite par une inéquation $cx \geq c_0$ telle que, pour tout ensemble légal L , la restriction $c^*x \geq c_0$ de $cx \geq c_0$ à l'ensemble L induit une facette de $TSP(p)$. Les facettes régulières de $ITSP(n, p)$ sont *tsp*-fortes tandis que les facettes triviales sont fortes mais non *tsp*-fortes. Le théorème suivant montre qu'à l'exception de certaines facettes de sous-tours toutes les facettes fortes non triviales sont *tsp*-fortes.

Théorème 3.3.11 *Soit $cx \geq c_0$ une inéquation induisant une facette forte non triviale du polytope $ITSP(n, p)$.*

*L'inéquation $cx \geq c_0$ est une inéquation de sous-tour associée à un ensemble S tel que $\mu(S) = 1$ si et seulement si la facette induite par l'inéquation $cx \geq c_0$ n'est pas *tsp*-forte ; i.e. il existe un ensemble légal L tel que la restriction $c^*x \geq c_0$ de $cx \geq c_0$ à l'ensemble L induit une face de dimension $\dim(TSP(p))$.*

Preuve : Soient S un ensemble tel que $\mu(S) = 1$ et L un ensemble légal tel que $|L \cap S| = 1$. Toute tournée internationale T appartenant à $\mathcal{I}(L)$ est telle que $cx^T = c_0$.

Réciproquement, supposons qu'il existe un ensemble légal L tel que, pour toute tournée internationale T de $\mathcal{I}(L)$, on a $cx^T = c_0$. Soit S_m un ensemble maximal de sommets qui contient L tel que, pour tout ensemble légal $L' \subset S_m$, la restriction $c^*x \geq c_0$ de $cx \geq c_0$ à L' induit une face de dimension égale à $\dim(TSP(p))$. Sans perte de généralité, on peut supposer que $c_0 = 0$ et $c_{i,j} = 0$ pour tout $[i, j] \in E(S_m)$. La taille de l'ensemble S_m est strictement inférieure à n . Soit u un sommet quelconque appartenant à $V \setminus S_m$. On note W_u le pays tel que $u \in W_u$ et on définit l'ensemble légal $L_u = (L \setminus (L \cap W_u)) \cup \{u\}$. La restriction de $cx \geq c_0$ à l'ensemble L_u induit une facette du $TSP(p)$, alors, pour tout sommet $z \in L_u \setminus \{u\}$, il existe une tournée internationale $T \in \mathcal{I}_c^-(L_u)$ qui utilise l'arête $[u, z]$. On note v le voisin de u dans T différent de z . La tournée internationale T sature l'inéquation $cx \geq c_0$ alors $c_{u,z} = -c_{u,v} = k_u$, avec k_u une constante positive non nulle. Toutes les tournées internationales de $\mathcal{I}_c^-(L_u)$ utilisent l'arête $[u, v]$ et $c_{u,w} = k_u$, pour $w \in L_u \setminus \{v\}$. En effet, s'il existe une tournée internationale T' de $\mathcal{I}_c^-(L_u)$ n'utilisant pas l'arête $[u, v]$, alors tous les coefficients des arêtes $[u, w]$ sont nuls, pour $w \in S_m \setminus W_u$ et S_m ne serait plus maximal. On note W_u^- le pays qui contient le sommet v . De la même manière, on détermine les coefficients des toutes les arêtes $[u, z]$, pour $z \in S_m \setminus W_u$, et on obtient les coefficients suivants :

$$c_{u,z} = \begin{cases} -k_u & \text{si } z \in S_m \cap W_u^- \\ k_u & \text{si } z \in S_m \setminus W_u^- \end{cases}$$

Si $\mu(S_m) = p - 1$, alors les coefficients de l'inéquation $cx \geq c_0$ sont complètement déterminés. Pour que l'inéquation $cx \geq c_0$ induise une facette, il faut que $|S_m| = n - 1$. Dans ce cas, $cx \geq c_0$ est l'inéquation de sous-tour associée à l'ensemble minimal $W_u^- \cup \{u\}$. Supposons que $\mu(S_m) < p - 1$. Soient $a \in W_a \setminus (S_m \cap W_a)$ et $b \in W_b \setminus (S_m \cap W_b)$ tels que $W_a \neq W_b$. On définit l'ensemble légal $L_{a,b} = (L \setminus L \cap (W_a \cup W_b)) \cup \{a, b\}$ et on note a^- et b^- les sommets de $L_{a,b}$ appartenant respectivement à W_a^- et W_b^- . Soit \mathcal{P} l'ensemble des chaînes hamiltoniennes d'extrémités a et b qui n'utilisent que les sommets de $L_{a,b}$. Le coefficient de l'arête $[a, b]$ est tel que $c_{a,b} = -\text{Min} \{cx^P : P \in \mathcal{P}\}$. Pour le calcul de $c_{a,b}$, on distingue trois cas :

1^{er} cas : $W_a^- \neq W_b$ et $W_b^- \neq W_a$.

Si $W_a^- \neq W_b^-$, on vérifie facilement que $c_{a,b} = k_a + k_b$. Soit une tournée internationale $T \in \mathcal{I}_c^-(L_{a,b})$. T utilise nécessairement les arêtes $[a, a^-]$ et $[b, b^-]$ alors la restriction de $cx \geq c_0$ à l'ensemble $L_{a,b}$ n'induit pas une facette de $TSP(p)$. Les pays W_a^- et W_b^- sont identiques. Le coefficient $c_{a,b} = \text{Min} \{-k_a + k_b, k_a - k_b\} \leq 0$. Les constantes k_a et k_b sont égales. En effet, si les constantes k_a et k_b sont différentes, la restriction de $cx \geq c_0$ à l'ensemble $L_{a,b}$ est dominée par la même restriction avec des constantes k_a et k_b identiques; ce qui nous donne un coefficient $c_{a,b} = 0$.

2^{ème} cas : $W_a^- = W_b$ et $W_b^- \neq W_a$.

Dans ce cas $c_{a,b} = -k_a + k_b$. Une tournée internationale $T \in \mathcal{I}_c^-(L_{a,b})$ utilise nécessairement les arêtes $[a, b]$ et $[b, b^-]$. La restriction de $cx \geq c_0$ à l'ensemble $L_{a,b}$ est dominée par l'inéquation $x_{a,b} \leq 1$. Ce deuxième cas est impossible.

3^{ème} cas : $W_a^- = W_b$ et $W_b^- = W_a$.

Dans ce cas $c_{a,b} = -(k_a + k_b)$. Supposons que $\mu(S_m) < p - 2$. Soit d un sommet n'appartenant pas à S_m et appartenant à un pays W_d différent de W_a et W_b . D'après le premier cas, W_d^- est égal à $W_b^- = W_a$ ou à $W_a^- = W_b$. Supposons que $W_d^- = W_a$. $W_a^- = W_b \neq W_d$, alors on est ramené au deuxième cas (impossible). En conclusion, ce troisième cas n'est possible que lorsque $\mu(S_m) = p - 2$. Dans ce cas, les arêtes $[a', b']$, avec $a' \in W_a \setminus (S_m \cap W_a)$ et $b' \in W_b \setminus (S_m \cap W_b)$, ont des coefficients $c_{a',b'} = -(k_{a'} + k_{b'})$. Soit $T \in \mathcal{I}_c^-(S_m) \setminus \mathcal{I}_c^-(S_m)$, T utilise nécessairement une arête $[a', b']$, avec $a' \in W_a$ et $b' \in W_b$. L'inéquation $cx \geq c_0$ induit le sous-tour associé à un ensemble $S = W_b \cup \{a'\}$, avec $a' \in W_a \setminus (S_m \cap W_a)$ et $\mu(S) = 1$.

Finalement, il reste à expliciter l'inéquation $cx \geq c_0$ lorsque le premier cas est le seul possible. Pour tout sommet $a \notin S_m$, le pays W_a^- est contenu dans S_m et, de plus, il ne dépend pas du sommet a . On notera $W_a^- = W^-$, pour tout $a \notin S_m$. L'inéquation $cx \geq c_0$ est défini par les coefficients suivants :

$$c_{a,b} = \begin{cases} -k & \text{si } a \in W^- \text{ et } b \notin S_m \\ k & \text{si } a \in S_m \setminus W^- \text{ et } b \notin S_m \\ 0 & \text{sinon} \end{cases}$$

La constante k est positive non nulle et le second membre c_0 est nul. L'inéquation $cx \geq c_0$ induit la facette de sous-tour associé à l'ensemble $S = W^- \cup (V \setminus S_m)$. L'ensemble S vérifie $\mu(S) = 1$. \square

Dans la suite, on étudie les inéquations d'arbres de cliques dans le cas non régulier ; i.e. au moins une dent ou un manche est un ensemble non régulier. On ne présentera pas cette généralisation pour les inéquations de peignes. En effet, la définition et le résultat qui suivent prennent en compte le cas particulier où l'on a un seul manche.

3.3.6.2 Les arbres de cliques non réguliers

Soit un arbre de cliques $(\mathcal{H}, \mathcal{T})$ qui vérifie les conditions (a), (b), (c), (d) et (e) du paragraphe 3.3.5.3. L'arbre de cliques $(\mathcal{H}, \mathcal{T})$ est dit non régulier si au moins un manche ou une dent est un ensemble non régulier.

$$\sum_{j=1}^h x(\delta(H_j)) + \sum_{i=1}^t x(\delta(T_i)) \geq \sum_{i=1}^t (2t_i + 1) + 1. \quad (3.13)$$

Théorème 3.3.12 *L'inéquation 3.13 associée à un arbre de cliques non régulier $(\mathcal{H}, \mathcal{T})$ est valide pour le polytope $ITSP(n, p)$ si et seulement si les ensembles suivants sont valides :*

- (i) $T_i \setminus \mathcal{H}$ pour $i = 1, \dots, t$;
- (ii) $T_i \cap \mathcal{H}$ pour $i = 1, \dots, t$.

De plus, elle induit une facette du polytope $ITSP(n, p)$.

Preuve : Soient L un ensemble légal quelconque et $c^*x \geq c_0$ la restriction de $cx \geq c_0$ à l'ensemble L . On définit les ensembles $H_i^L = H_i \cap L$, pour $i = 1, \dots, h$ et $T_i^L = T_i \cap L$, pour $i = 1, \dots, t$. Les hypothèses de la validité (i) et (ii) garantissent qu'aucun de ces ensembles n'est vide et que l'arbre de cliques $\mathcal{C}^L = \{H_1^L, \dots, H_h^L, \dots, T_1^L, \dots, T_t^L\}$ vérifie les conditions (a), (b), (c), (d) et (e) du paragraphe 3.3.5.3. L'inéquation d'arbre de cliques $c^*x \geq c_0$ associée à \mathcal{C}^L induit une facette du polytope $TSP(p)$ pour tout ensemble légal L alors l'inéquation $cx \geq c_0$ est valide pour le

polytope $ITSP(n, p)$. Supposons que celle-ci n'induit pas une facette et qu'elle soit dominée par une inéquation $bx \geq b_0$. Pour tout ensemble légal L , la restriction $b^*x \geq b_0$ de $bx \geq b_0$ à l'ensemble L induit une face de dimension au moins égale à $\dim(TSP(p)) - 1$. L'inéquation $bx \geq b_0$ vérifie les hypothèses du théorème 3.3.11. Soient T une tournée internationale telle que $bx^T = b_0$ et $cx^T > c_0$ et L l'ensemble légal induit par T . La restriction $b^*x \geq b_0$ de l'inéquation $bx \geq b_0$ à l'ensemble L induit une face de dimension $\dim(TSP(p))$, alors, d'après le théorème 3.3.11, l'inéquation $bx \geq b_0$ est l'inéquation de sous-tour associée à un ensemble S tel que $\mu(S) = 1$. On note L' un ensemble légal tel que $|L' \cap S| \geq 2$. Soient $c^*x \geq c_0$ et $b^*x \geq b_0$ les restrictions respectives de $cx \geq c_0$ et $bx \geq b_0$ à l'ensemble L' . L'inéquation $c^*x \geq c_0$ induit un arbre de cliques de $TSP(p)$ et l'inéquation $b^*x \geq b_0$ induit un sous-tour de $TSP(p)$. Ce sont deux facettes différentes de $TSP(p)$, alors il existe une tournée internationale de $\mathcal{I}(L')$ qui sature l'inéquation $cx \geq c_0$ mais pas $bx \geq b_0$. L'inéquation de sous-tour $bx \geq b_0$ ne domine pas l'inéquation $cx \geq c_0$. \square

Ainsi, on obtient une description linéaire partielle de $ITSP(n, p)$. On peut élargir cette description partielle en considérant la généralisation pour le cas non régulier des inéquations induisant d'autres facettes régulières de $ITSP(n, p)$ telles que les inéquations de chemin [68] ou les inéquations d'échelle [11]. On évitera de présenter ce type de généralisations parce qu'elles entraînent des définitions et des notations complexes et, de plus, elles ne contribuent pas dans la résolution du $ITSP$. En revanche, dans la suite, on présente deux résultats qui décrivent des opérations simples sur les inéquations en vue d'obtenir de nouvelles inéquations induisant des facettes du $ITSP(n, p)$.

3.3.7 Extensions

Pour trouver de nouvelles inéquations, valides ou induisant des facettes du polytope du TSP , une technique consiste à construire de nouvelles inéquations à partir d'inéquations induisant des facettes du polytope du TSP . Plusieurs auteurs ont défini des opérations sur une ou plusieurs inéquations pour obtenir de nouvelles inéquations valides ou induisant une facette du polytope du TSP (voir [56]). Les résultats ainsi obtenus sur la structure faciale du polytope du TSP sont valides pour la classe des facettes régulières de

$ITSP(n, p)$.

Dans cette section, on décrit deux opérations différentes pour obtenir des inéquations induisant des facettes de $ITSP(n+1, p)$ à partir d'inéquations induisant des facettes de $ITSP(n, p)$. On notera $G = (V, E)$ et $G' = (V', E')$ respectivement les graphes multipartis complets K_{n+1}^p et K_n^p . Soit $\{v\} = V \setminus V'$, avec v un sommet appartenant à un pays non dégénéré, disons W_p .

Théorème 3.3.13 *Soient u un sommet différent de v appartenant à W_p et $cx \geq c_0$ une inéquation induisant une facette forte du polytope $ITSP(n, p)$.*

L'inéquation $bx \geq b_0$ définie par $b_0 = c_0$ et

$$b_{i,j} = \begin{cases} c_{u,j} & \text{si } i = v \\ c_{i,j} & \text{sinon} \end{cases}$$

induit une facette du polytope $ITSP(n+1, p)$ si et seulement si $cx \geq c_0$ n'est pas une inéquation de sous-tour associée à un ensemble minimal $S = W \cup \{u\}$, avec W un pays différent de W_p .

Preuve : Si $cx \geq c_0$ est une inéquation de sous-tour associée à un ensemble minimal $S = W \cup \{u\}$, alors l'inéquation $bx \geq b_0$ n'induit pas une facette de $ITSP(n+1, p)$. L'inéquation de sous-tour $b'x \geq b'_0$ associée à l'ensemble minimal $S = W \cup \{u\}$ induisant une facette de $ITSP(n+1, p)$ domine l'inéquation $bx \geq b_0$.

Réciproquement, supposons qu'il existe une inéquation $ax \geq a_0$ qui domine l'inéquation $bx \geq b_0$. Soit T' une tournée internationale telle que $ax^{T'} = a_0$ et $bx^{T'} > b_0$. Considérons la restriction $a^*x \geq a_0$ de l'inéquation $ax \geq a_0$ à l'ensemble légal L' engendré par T' . Si $a^*x \geq a_0$ induit une facette de $TSP(p)$, alors $a^*x \geq a_0$ domine la restriction $c^*x \geq c_0$ de $cx \geq c_0$ à l'ensemble L' si $v \notin L'$ ou à l'ensemble $L' \cup \{u\} \setminus \{v\}$ si $v \in L'$. La dimension de la face induite par $c^*x \geq c_0$ est strictement inférieure à $\dim(TSP(p)) - 1$, alors $cx \geq c_0$ n'induit pas une facette forte de $ITSP(n, p)$. Ceci contredit l'hypothèse du théorème, alors $a^*x \geq a_0$ induit une face de dimension $\dim(TSP(p))$. On distingue, alors, deux cas :

1^{er} cas : T' ne visite pas le sommet v .

Alors, la restriction de l'inéquation $ax \geq a_0$ à l'ensemble $V \setminus \{v\}$ induit une face de dimension $\dim (ITSP(n, p))$. D'après le théorème 3.3.11, $ax \geq a_0$ est l'inéquation de coupe associée à un ensemble minimal $W' \cup \{v\}$, avec W' un pays différent de W_p . Soient $a^*x \geq a_0$ et $b^*x \geq b_0$ les restrictions respectives de $ax \geq a_0$ et $bx \geq b_0$ à l'ensemble $V \setminus \{u\}$. L'inéquation $a^*x \geq a_0$ induit une facette de $ITSP(n-1, p)$ et l'inéquation $b^*x \geq b_0$ est exactement l'inéquation $cx \geq c_0$ qui induit une facette différente de celle induite par l'inéquation $a^*x \geq a_0$. Alors, il existe une tournée internationale T telle que $bx^T = b_0$ et $ax^T > a_0$.

2^{ème} cas : T' visite le sommet v .

Alors, la restriction de l'inéquation $ax \geq a_0$ à l'ensemble $V \setminus \{u\}$ induit une face de dimension $\dim (ITSP(n, p))$. D'après le théorème 3.3.11, $ax \geq a_0$ est l'inéquation de coupe associée à un ensemble minimal $W'' \cup \{u\}$, avec W'' un pays différent de W_p . Soit $a^*x \geq a_0$ la restriction de $ax \geq a_0$ à l'ensemble $V \setminus \{v\}$. L'inéquation $a^*x \geq a_0$ induit une facette de sous-tour associée à un ensemble minimal et l'inéquation $cx \geq c_0$ induit une facette qui n'est pas un sous-tour associé à un ensemble minimal. Alors, il existe une tournée internationale T telle que $bx^T = b_0$ et $ax^T > a_0$.

Dans les deux cas, l'inéquation $ax \geq a_0$ n'existe pas. □

Théorème 3.3.14 *Si $cx \geq c_0$ est une inéquation induisant une facette de $ITSP(n, p)$, avec $c \geq 0$, alors il existe $\alpha_{v,j} \geq 0$, pour tout $[v, j] \in \delta(v)$, tels que l'inéquation $bx \geq b_0$ définie par $b_0 = 3c_0$ et :*

$$b_{i,j} = \begin{cases} \alpha_{v,j} & \text{si } i = v \\ c_{i,j} + c_0 & \text{si } i \in W_p \setminus \{v\} \\ c_{i,j} & \text{sinon} \end{cases}$$

induit une facette du polytope $ITSP(n+1, p)$.

Preuve : Dans cette preuve on montre qu'on peut calculer itérativement les coefficients $\alpha_{v,j}$, pour $[v, j] \in \delta(v)$, tels que l'inéquation $bx \geq b_0$ induise une facette du polytope $ITSP(n+1, p)$. Soit $cx + c_0x(\delta(W_p \setminus \{v\})) \leq 3c_0$ l'inéquation du $ITSP(n, p)$ équivalente à $cx \geq c_0$ et notée $c'x \geq c'_0$. Initialement, on considère des coefficients $\alpha_{[v, j]} = 0$, pour $[v, j] \in \delta(v)$ sauf pour

une arête $[v, u_1]$ où $\alpha_{v, u_1} = 2c_0$. Soit W_1 le pays tel que $u_1 \in W_1$. On désigne par u_1, u_2, \dots, u_{n_1} les sommets de W_1 , avec $n_1 = |W_1|$, et w_1, w_2, \dots, w_k les sommets de $V \setminus (W_1 \cup W_p)$. L'inéquation $b'x \geq b'_0$ définie par $b'_0 = c'_0$ et

$$b'_{i,j} = \begin{cases} 2c_0 & \text{si } i = v \text{ et } j = u_1 \\ 0 & \text{si } i = v \text{ et } j \neq u_1 \\ c'_{i,j} & \text{sinon} \end{cases}$$

est valide pour $ITSP(n+1, p)$. Pour $i = 1, \dots, k$, considérons le problème d'optimisation $Z_{w_i} = \text{Max} \{b'x^T : T \in \mathcal{T}_{v,w_i}\}$, avec \mathcal{T}_{v,w_i} l'ensemble des tournées internationales qui utilisent l'arête $[v, w_i]$. Soit T_{w_i} une tournée internationale telle que $b'x^{T_{w_i}} = Z_{w_i}$ et posons $\alpha_{v,w_i} = c'_0 - Z_{w_i}$. Ainsi, on calcule itérativement les coefficients α_{v,w_i} , pour $i = 1, \dots, k$. Les coefficients α_{v,w_i} sont inférieurs ou égaux à c'_0 . En effet, si $\alpha_{v,w_i} > c'_0$, alors une tournée internationale T qui utilise les deux arêtes $[v, w_i]$ et $[v, u_1]$ est telle que $b'x^T > b'_0$. Soient $T_{w_1}, T_{w_2}, \dots, T_{w_k}$ la séquence des tournées internationales obtenues. Chacune des tournées internationales $T_{w_1}, T_{w_2}, \dots, T_{w_k}$ utilise l'arête $[v, u_1]$. En effet, supposons qu'il existe une tournée internationale T_{w_l} ($1 \leq l \leq k$) qui n'utilise pas l'arête $[v, u_1]$. La valeur Z_{w_l} correspondante est au moins égal à $2c_0$, alors $\alpha_{v,w_l} \geq c'_0 - 2c_0 = c_0$. Ainsi, toute tournée internationale T qui utilise les arêtes $[v, w_l]$ et $[v, u_1]$ est telle que $b'x^T = b'_0$, on pose une telle tournée internationale égale à T_{w_k} . Pour $i = 2, \dots, n_1$, on calcule Z_{u_i} , on pose $\alpha_{v,u_i} = c'_0 - Z_{u_i}$ et on note T_{u_i} la tournée internationale telle que $b'x^{T_{u_i}} = Z_{u_i}$. Supposons qu'il n'existe pas de tournée internationale T_{w_i,w_j} qui utilise deux arêtes $[v, w_i]$ et $[v, w_j]$, avec w_i et w_j appartenant à deux pays différents. Alors, il existe un réel positif ε tel que l'inéquation $bx \geq b_0$ définie par $b_0 = c'_0 = 3c_0$ et :

$$b_{i,j} = \begin{cases} \alpha_{v,j} + \varepsilon & \text{si } i = v \text{ et } j \notin W_1 \\ \alpha_{v,j} - \varepsilon & \text{si } i = v \text{ et } j \in W_1 \\ c'_{i,j} & \text{sinon} \end{cases}$$

est saturée par une tournée internationale $T_{w_{i_0},w_{j_0}}$ qui utilise deux arêtes $[v, w_{i_0}]$ et $[v, w_{j_0}]$, avec w_{i_0} et w_{j_0} appartenant à deux pays différents. Les tournées internationales $T_{w_1}, T_{w_2}, \dots, T_{w_k}, T_{u_2}, T_{u_3}, \dots, T_{u_{n_1}}$ et $T_{w_{i_0},w_{j_0}}$ saturent l'inéquation $bx \geq b_0$ et forment un ensemble $\mathcal{S}_{W_1,v}(W_1, v)$ -maximal. La restriction de l'inégalité $bx \geq b_0$ à l'ensemble $V \setminus \{v\}$ induit une facette de $ITSP(n, p)$, alors il existe un ensemble \mathcal{S} de $m - |\delta\{u\}| - p$ tournées

internationales affinements indépendantes qui saturent l'inéquation $bx \geq b_0$ et qui n'utilisent pas le sommet v . D'après le lemme 3.3.2, l'ensemble $\mathcal{S}_{W_1, v} \cup \mathcal{S}$ est un ensemble de $m - p$ tournées internationales affinements indépendantes qui saturent l'inégalité $bx \geq b_0$. L'inéquation $bx \geq b_0$ induit une facette de $ITSP(n + 1, p)$. \square

On peut développer une autre preuve du théorème 3.3.10 comme application de ce résultat. En effet, les problèmes d'optimisation Z_{v_i} définis dans la preuve précédente sont faciles dans le cas des inéquations de coupe.

3.4 Conclusion

La description linéaire partielle du polytope $ITSP(n, p)$ en inéquations induisant des facettes constitue le résultat principal de ce chapitre. On notera que toutes les facettes qu'on a présenté sont fortes. On ne sait pas s'il existe des facettes de $ITSP(n, p)$ qui ne sont pas fortes.

Dans le chapitre suivant, on décrira les principales phases d'un algorithme de branchement et de coupe pour la résolution d'un problème d'optimisation combinatoire quelconque et en particulier le TSP . L'objectif principal sera de décrire les principales étapes qui permettent d'obtenir un algorithme de branchement et de coupe pour la résolution du $ITSP$ à partir d'un algorithme de de branchement et de coupe pour la résolution du TSP . On présentera plusieurs procédures de séparation de certaines facettes de $ITSP(n, p)$ et on finira par donner des résultats numériques obtenus par l'implantation que nous avons réalisée.

Chapitre 4

Algorithme de branchement et de coupe

Le principe de l'algorithme de branchement et de coupe a déjà été développé dans la section 2.2.4 du chapitre 2. L'objectif principal de ce chapitre consiste à présenter les différentes modifications qu'il faut apporter à un algorithme de branchement et de coupe pour la résolution du *TSP*, ainsi que les différents modules qu'il faut rajouter, pour obtenir un algorithme de branchement et de coupe pour la résolution du *ITSP*.

Ces dernières années, plusieurs implantations de l'algorithme de branchement et de coupe pour la résolution du *TSP* ont été réalisées. Dans [78], Padberg et Rinaldi donnent la description détaillée de l'algorithme qu'ils ont implanté.

Pour notre part, on a utilisé l'implantation de l'algorithme de branchement et de coupe pour la résolution du *TSP* réalisée par Clochard et Naddef [17] comme support de l'implantation d'un tel algorithme pour la résolution du *ITSP* que nous avons réalisée.

Dans la section 4.1, on rappellera brièvement le principe d'un algorithme de branchement et de coupe et on décrira les principaux aspects qui le caractérisent. Dans la section 4.2, on citera les différentes modifications qu'il faut apporter à un algorithme de branchement et de coupe pour la résolution du *TSP* pour obtenir un algorithme de branchement et de coupe pour la

résolution du *ITSP*. Des algorithmes de séparation de certaines contraintes sont présentés dans la section 4.3. On finira le chapitre en donnant une procédure de Fischetti et al. [28] qui génère des instances du *ITSP* à partir d'instances euclidiennes du *TSP*, et des résultats numériques obtenus par notre implantation. On note que, dans ce chapitre, une contrainte désignera une inéquation induisant une facette.

4.1 Principales phases de l'algorithme

L'algorithme de branchement et de coupe peut être perçu comme un algorithme de branchement et d'évaluation où, dans chaque nœud de l'arbre d'énumération, on utilise une procédure de coupe (voir la section 2.2.4.2 du chapitre 2) pour calculer une borne inférieure de la solution optimale du problème. Les contraintes générées par la procédure de coupe dans chaque nœud sont valides pour l'ensemble des nœuds de l'arbre de branchement.

Avant d'appliquer l'algorithme, on utilise une heuristique amélioratrice pour obtenir une borne supérieure BS de la solution optimale du problème. On peut améliorer la valeur de BS en appliquant également, dans l'algorithme et avant le branchement, une heuristique amélioratrice en considérant une solution entière initiale qui exploite la solution optimale fractionnaire. Dans le cas où la solution optimale du problème est connue, on prendra la borne supérieure BS égal à l'optimum.

Dans la figure 4.1, on présente un algorithme de branchement et de coupe général. Il peut s'adapter à n'importe quel problème d'optimisation combinatoire dont on dispose une certaine connaissance polyédrale, et en particulier le *ITSP*.

L'algorithme présente trois aspects fondamentaux qui conditionnent son efficacité : la séparation des contraintes, la gestion des variables et les stratégies de branchement.

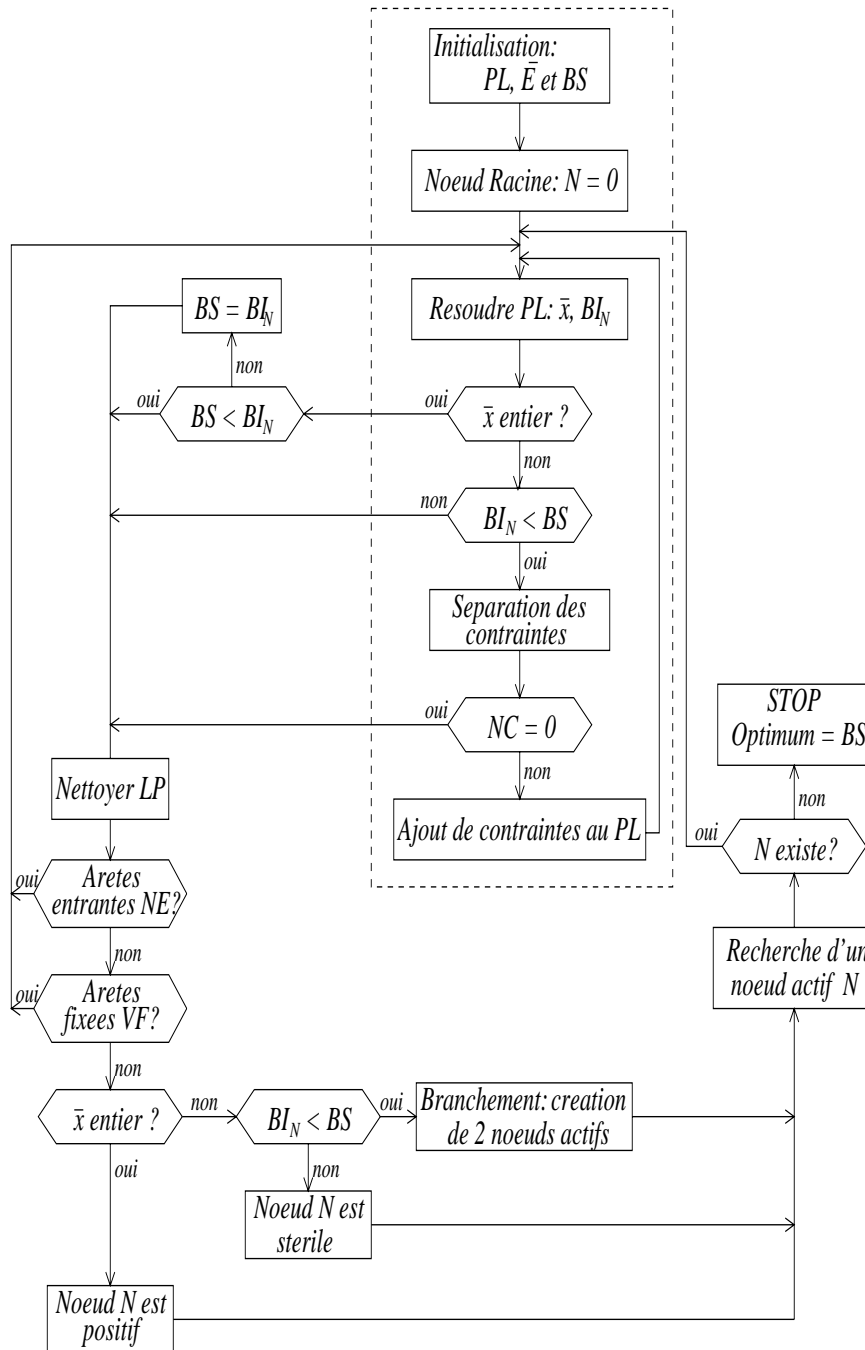


FIG. 4.1 – Algorithme de branchement et de coupe.

4.1.1 Séparation des contraintes

Nous avons vu dans la section 2.2.4.2 du chapitre 2 que, pour un problème donné, si la séparation des contraintes est rapide, alors l'algorithme de coupe qui en résulte l'est d'autant. Pour le *ITSP*, le problème de séparation des contraintes est évidemment difficile et, généralement, on utilise le branchement pour résoudre les problèmes. On s'intéressera à des classes particulières de contraintes où la séparation est relativement facile. Il est important de développer des procédures de séparation efficaces qui permettent, dans chaque nœud, de calculer une "bonne" borne inférieure pour obtenir un arbre de branchement de taille réduite.

Dans la section 4.3, on présentera plusieurs procédures pour la séparation des contraintes de sous-tours et de peignes. Les contraintes violées identifiées sont rajoutées au programme linéaire courant noté *PL*. Dans l'algorithme, on désigne par *NC* le nombre de ces contraintes. Toutes les contraintes du *PL* sont enregistrées dans une liste de contraintes notée *Contraintes_PL*; une contrainte de sous-tour est enregistrée par le plus petit ensemble de sommets qui la définit et une contrainte de peigne est enregistrée par les ensembles de sommets qui définissent le manche et les dents.

La résolution du *PL*, initialement réduit aux équations de degré et aux inéquations de non négativité, donne une solution optimale \bar{x} généralement fractionnaire et un optimum noté BI_N (N représente le numéro du nœud courant).

Dans la figure 4.1, la procédure de coupe est encadrée en trait pointillé. On sort de la procédure lorsque l'une des conditions suivantes est vérifiée :

- (i) la résolution du programme linéaire donne une solution optimale \bar{x} entière ou un optimum BI_N supérieur ou égal à la borne supérieure BS ;
- (ii) aucune contrainte violée par \bar{x} n'est identifiée dans la procédure de coupe.

On retire régulièrement du *PL* les contraintes dont l'écart entre le membre de gauche et le membre de droite est important au point \bar{x} . On dit qu'on

nettoie le programme linéaire PL . Généralement, on nettoie le PL à la sortie de la procédure de coupe ou lorsque le temps de résolution du PL est supérieur ou égal à une certaine limite qu'on fixera en fonction de la taille de l'instance. Les contraintes de sous-tours retirées du PL sont détruites de la liste $Contraintes_PL$ et les contraintes de peignes retirées du PL sont également détruites de la liste $Contraintes_PL$ mais enregistrées dans une autre liste, notée $Contraintes_Poubelle$, parce qu'elles sont relativement peu nombreuses et difficiles à générer. Dans la phase de la séparation des contraintes et avant d'appliquer les procédures d'identification des contraintes de peignes violées, on rajoutera au programme linéaire les contraintes de la liste $Contraintes_Poubelle$ qui ne sont pas satisfaites par le point fractionnaire \bar{x} .

4.1.2 Gestion des variables

L'algorithme de branchement et de coupe se propose de résoudre des instances de grande taille. Le nombre de variables est très élevé et il est très coûteux, en temps de calcul et en place mémoire, de gérer un programme linéaire qui considère l'ensemble des variables. La technique qu'on utilise et qui permet de considérer un nombre restreint de variables est connue sous le nom de génération de colonnes.

Initialement, on choisit un sous-ensemble \bar{E} d'arêtes dont le graphe partiel associé est susceptible de contenir une solution optimale. L'ensemble \bar{E} est généralement choisi en utilisant des critères géométriques. Pour notre part, l'ensemble \bar{E} représente les arêtes de la triangulation de Delaunay et les arêtes reliant deux sommets qui sont des extrémités d'une chaîne de 2 arêtes de la triangulation de Delaunay. Les colonnes du programme linéaire initial sont les arêtes de \bar{E} . On dira qu'une arête est active si elle est dans le programme linéaire, sinon elle est dite passive. Une arête passive est dite entrante si elle a un coût réduit négatif. La résolution du PL donne une solution optimale \bar{x} sur l'ensemble des colonnes \bar{E} . Le point \bar{x} est également une solution optimale sur l'ensemble des arêtes E s'il n'existe pas d'arête entrante. Pour obtenir une solution optimale sur l'ensemble des arêtes E , on intègre dans le programme linéaire toutes les variables entrantes à la sortie de la procédure de coupe. De telles variables passives entrent en base et deviennent actives.

Dans l'algorithme, on désigne par NE le nombre de variables entrantes.

On notera que le coût réduit CR_e d'une variable x_e est donné par la relation :

$$CR_e = p_e - \sum_{i=1}^q c_{i,e} y_i^*,$$

avec p_e le poids de l'arête e , q le nombre de contraintes dans le programme linéaire courant PL , $c_{i,e}$ le coefficient de la variable x_e dans la contrainte i et y_i^* la valeur optimale de la variable duale correspondante à la contrainte i . Pour calculer rapidement le coût réduit d'une variable passive, il faut pouvoir générer facilement ses coefficients dans les différentes contraintes du PL . Pour cela, on utilise la liste *Contraintes-PL*.

Une autre manière de limiter le nombre de variables consiste à fixer une variable à 0 lorsqu'on est certain qu'il existe une solution optimale qui ne l'utilise pas et à 1 lorsqu'on est certain qu'il existe une solution optimale qui l'utilise.

En effet, si la composante \bar{x}_e d'une arête e vaut 1 et son coût réduit CR_e vérifie la relation $CR_e < -(BS - BI_N)$, alors il existe une solution optimale qui utilise l'arête e . Dans ce cas, la variable x_e est fixée à 1 dans l'ensemble des nœuds de la branche de l'arbre d'énumération issue du nœud courant N . D'une manière similaire, si la composante \bar{x}_e d'une arête e est nulle et son coût réduit CR_e vérifie la relation $CR_e > (BS - BI_N)$, alors il existe une solution optimale qui n'utilise pas l'arête e . Dans ce cas, la variable x_e est fixée à 0 dans l'ensemble des nœuds de la branche de l'arbre d'énumération issue du nœud courant N .

On peut également fixer certaines variables à 0 ou à 1 par des déductions logiques. En effet, si une variable $x_{u,v}$ est fixée à 1, alors on peut fixer à 0 toutes les variables $x_{u',v'}$ associées à des arêtes non incidentes à u ou à v telles que $u' \in W_u$ ou $v' \in W_v$, avec u et v appartenant respectivement aux pays W_u et W_v . De la même manière, si deux arêtes $[v, u]$ et $[u, w]$ sont fixées à 1, alors on peut fixer les autres arêtes incidentes au sommet u et l'arête $[v, w]$ à 0. Dans l'algorithme, on désigne par VF le nombre de variables fixées à 0 ou à 1.

4.1.3 Stratégies de branchement

Avant de décrire les différentes stratégies de branchement, on donnera quelques définitions. Initialement, le problème, non encore subdivisé, représente le nœud racine de l'arbre de branchement. On dira qu'un nœud est traité lorsqu'on sort de la procédure de coupe et aucune variable ne vient d'être fixée.

Un nœud traité est dit positif si \bar{x} est une solution entière et il est dit stérile si BI_N est supérieure à BS . Un nœud stérile ou positif est une feuille de l'arbre de Branchement. On dit qu'un nœud est passif s'il est une feuille positive ou stérile, ou un nœud interne admettant deux nœuds fils passifs, sinon il est dit actif. On atteint l'optimum lorsque tous les nœuds deviennent passifs.

Un nœud traité, non positif et non stérile, est subdivisé en 2 nœuds fils actifs en utilisant une stratégie de branchement.

Le critère de Branchement le plus connu consiste à brancher sur une variable x_{e_0} . Dans l'un des deux nœuds fils, on fixe la variable x_{e_0} à 0 et, dans l'autre nœud, on fixe la variable x_{e_0} à 1. On peut choisir la variable x_{e_0} dont la valeur \bar{x}_{e_0} est la plus proche de 0.5 ou, parmi un ensemble de variables dont les valeurs sont proches de 0.5, celle qui a le poids le plus élevé (voir [78]).

On peut également brancher sur une contrainte de sous-tour. Soit S tel que $\bar{x}(\delta(S))$ est proche d'un nombre impair, disons $2t + 1$. Le branchement s'effectue en ajoutant la contrainte $\bar{x}(\delta(S)) \leq 2t$ dans l'un des deux nœuds fils et la contrainte $\bar{x}(\delta(S)) \geq 2t + 2$ dans l'autre nœud. Il n'est pas facile de trouver un tel ensemble S . Dans [16], Clochard et Naddef considèrent les manches des différentes contraintes de peignes générées.

Généralement, on choisit une stratégie qui définit deux nœuds fils dont la borne inférieure (après quelques itérations du simplexe dual) la plus élevée dans les deux nœuds est la plus grande possible pour, éventuellement, rendre passif très rapidement le nœud fils dont la borne inférieure est la plus élevée.

La recherche d'un nœud actif, dans l'arbre de branchement, peut s'effectuer selon des stratégies de parcours différentes. Lorsque l'optimum n'est pas connu et que la borne supérieure BS est la valeur d'une solution entière trouvée par une heuristique, on considère la stratégie qui propose le nœud actif admettant la plus petite borne inférieure dans l'ensemble des nœuds actifs. On notera que la borne inférieure d'un nœud actif est l'optimum atteint par le programme linéaire correspondant au nœud à l'extérieur de la procédure de coupe. On choisit cette stratégie parce qu'on pense que les nœuds positifs donneront plus rapidement l'optimum.

En revanche, lorsque la borne supérieure est égal à l'optimum, l'arbre de branchement ne dépend de son parcours. Dans ce cas, on utilise la stratégie "en profondeur d'abord" parce qu'elle est simple à gérer.

En parcourant l'arbre de branchement, d'un nœud fils vers le nœud père, on doit défixer toutes les variables fixées à 1 ou à 0 dans le nœud fils et retirer, éventuellement, du programme linéaire la contrainte de sous-tour de branchement qui a défini ce nœud fils. D'une manière similaire, en parcourant l'arbre de branchement, d'un nœud père vers le nœud fils, on doit refixer toutes les variables fixées à 1 ou à 0 dans le nœud fils et remettre, éventuellement, dans le programme linéaire la contrainte de sous-tour de branchement qui a défini ce nœud fils.

On notera que, dans [27], Fischetti et al. utilisent également les contraintes de sous-tours et les variables pour le branchement.

4.2 Adaptation d'un algorithme de TSP

On obtient une implantation de l'algorithme de branchement et de coupe pour la résolution du *ITSP* à partir d'une implantation d'un tel algorithme pour la résolution du *TSP* en considérant les principales modifications suivantes :

- (i) intégrer une heuristique adaptée au *ITSP* pour calculer une borne supérieure de la solution optimale ;

- (ii) dans le programme linéaire initial, il faut considérer les équations de degré des pays au lieu des équations de degré des sommets et ne pas considérer de borne supérieure sur les variables ;
- (iii) enlever toutes les variables dont les deux extrémités appartiennent à un même pays ;
- (vi) considérer les procédures de séparation décrites dans la section 4.3 ;
- (v) considérer la fixation des variables par des déductions logiques décrite dans la section 4.1.2.

Nous avons apporté ces modifications à l'implantation de l'algorithme de branchement et de coupe pour la résolution du *TSP* réalisée par Clochard et Naddef [17]. Le critère de branchement et la stratégie pour le recherche d'un nœud actif ne changent pas. Clochard et Naddef [17] utilisent les critères de branchement sur les contraintes de sous-tour et sur les variables et recherchent un nœud actif selon la stratégie décrite précédemment pour le cas où l'optimum est inconnu.

Les procédures de séparation qui suivent sont les principaux modules que nous avons développé pour obtenir un algorithme de branchement et de coupe pour la résolution du *ITSP*.

4.3 Procédures de séparation

Dans cette section, on présentera des algorithmes exacts et des heuristiques pour la séparation de certaines contraintes non régulières. Pour le cas des contraintes régulières, on montrera que l'on peut utiliser les algorithmes de séparation des contraintes du *TSP* (p).

On notera que, dans [27], on trouve des algorithmes exacts et des heuristiques pour la séparation des contraintes de sous-tours et de peignes exprimées dans l'espace des arêtes et des sommets différentes de celles qu'on présente.

Soit \bar{x} la solution fractionnaire du programme linéaire PL . On désigne par \bar{G} le graphe partiel de $G = (E, V)$ (G est le graphe multiparti complet K_n^p) défini par l'ensemble des arêtes $\bar{E} = \{e \in E : \bar{x}_e > 0\}$. Le graphe \bar{G} est dit support du point \bar{x} . Si on associe, à chaque arête de \bar{G} , un poids égal à la composante correspondante de \bar{x} , on obtient un graphe valué qu'on désigne par le couple (\bar{G}, \bar{x}) .

On définit un graphe complet d'ordre p , noté $G_c = (V_c, E_c)$ et appelé graphe contracté de G , par l'ensemble de sommets V_c dont chaque élément s'identifie à un pays de G . On dira qu'un pays de G est un sommet de G_c . On associe au point \bar{x} un point \bar{x}^c de $\mathbb{R}^{|E_c|}$, appelé point contracté de \bar{x} , défini par :

$$\bar{x}_{W_i, W_j}^c = \sum_{l \in W_i} \sum_{k \in W_j} \bar{x}_{l, k}, \quad \forall [W_i, W_j] \in E_c.$$

De la même manière que pour le point \bar{x} , on définit le graphe support de \bar{x}^c , noté \bar{G}_c , comme étant le graphe partiel de G_c défini par l'ensemble des arêtes $\bar{E}_c = \{e \in E_c : \bar{x}_e^c > 0\}$.

Dans la suite, on s'intéressera à la séparation des contraintes de sous-tours et de peignes. On décrira des algorithmes exacts et des heuristiques de séparation des contraintes de sous-tours pour différents types d'ensembles. Pour les contraintes de peignes, on citera des heuristiques, développées et testées par différents auteurs, pour la séparation des contraintes de peignes de $TSP(p)$, et donc des peignes réguliers. Puis, on présentera une heuristique pour la séparation des contraintes de 2-couplages non réguliers et une autre heuristique pour la séparation des peignes non réguliers.

On supposera que toutes les procédures qui vont suivre retournent le nombre de contraintes violées qu'elles identifient.

4.3.1 Contraintes de sous-tours

Soit S un sous-ensemble de sommets. La coupe $\delta(S)$ est dite valide si l'ensemble S est valide, i.e. il existe un paire de pays W' et W'' telle que $W' \subset S$ et $W'' \subset V \setminus S$. Le problème de séparation des contraintes de sous-tours consiste à trouver une ou plusieurs coupes valides, dans le graphe

(\bar{G}, \bar{x}) , de valeur strictement inférieure à 2, ou montrer qu'une telle coupe n'existe pas.

Dans la suite, on s'intéressera à une coupe valide de poids minimum dans le graphe (\bar{G}, \bar{x}) . En effet, si la valeur d'une telle coupe est supérieure ou égale à 2 alors aucune contrainte de ce type n'est violée, et si, en revanche, sa valeur est inférieure à 2 alors elle définit une contrainte violée. Toutefois, cette approche pose un problème. En effet, un algorithme qui calcule une coupe valide de poids minimum ne détermine qu'une seule contrainte violée lorsque la valeur de la coupe valide de poids minimum est strictement inférieure à 2, alors qu'il est plus intéressant d'obtenir plusieurs contraintes violées si possible.

Dans ce qui suit, on décrira des procédures qui déterminent plusieurs contraintes violées en calculant la coupe valide de poids minimum. On s'intéressera aux algorithmes de séparation des contraintes de sous-tours définies par un ensemble minimal, puis celles définies par un ensemble régulier, et, enfin, celles définies par un ensemble non régulier.

4.3.1.1 Ensemble minimal

On rappelle qu'un ensemble S est dit minimal s'il est égal à l'union d'un pays W et d'un sommet u n'appartenant pas à W . Le nombre de contraintes de sous-tours associées à un ensemble minimal est égal à $n(p-1)$. Bien que le nombre de ces contraintes soit en $O(np)$, on ne peut pas se permettre de lister toutes ces contraintes dans le programme linéaire. Pratiquement, ce nombre, bien que polynomial en n , reste très élevé. Pour résoudre le problème de séparation de ce type de contraintes, il suffit de vérifier toutes les contraintes et de retenir celles qui sont violées. Ainsi, étant donné un point fractionnaire \bar{x} , on obtient une procédure de complexité $O(np)$ qui peut s'exprimer comme suit :

Procédure Sous_tour_minimal_1;

Début

 Pour i allant de 1 à n faire

 Soit W_k le pays de i ;

 Pour j allant de 1 à p faire


```

    Si  $W_j \neq W_k$  alors
       $S = W_j \cup \{i\}$ ;
      Si  $\bar{x}(\delta(S)) < 2$  alors
        Enregistrer  $S$  (contrainte violée);
      Fin Si
    Fin Si
  Fin Pour
Fin Pour
Fin.

```

Lors de nos expériences numériques, nous avons remarqué que les contraintes de sous-tours associées à un ensemble minimal les plus fréquemment violées sont celles associées à un ensemble défini par un pays et un sommet “prochent” l’un de l’autre. On mesure l’éloignement d’un sommet i à un pays W par le minimum des longueurs des arêtes d’extrémités i et j , pour j appartenant à W . On considère, pour chaque sommet i , la liste, notée $Sommet_Voisin[i]$, des q premiers pays les plus proches de i . Pratiquement, on fixe la constante q à 5. Dans l’heuristique qui suit, on vérifie, pour chaque sommet i , les contraintes de sous-tours associées à un ensemble minimal $S = W \cup \{i\}$, pour un pays W appartenant à $Sommet_Voisin[i]$.

```

Procédure Sous_tour_minimal_2;
Début
  Pour  $i$  allant de 1 à  $n$  faire
     $Liste\_Pays = Sommet\_Voisin[i]$ ;
    Répéter
       $Successeur = Liste\_Pays.succ$ ;
       $Pays = Liste\_Pays.pays$ ;
       $S = Pays \cup \{i\}$ ;
      Si  $\bar{x}(\delta(S)) < 2$  alors
        Enregistrer  $S$  (contrainte violée);
      Fin Si
    Jusqu’à  $Successeur == NULL$ ;
  Fin Pour
Fin.

```

On ne fera appel à la procédure exacte que lorsque l'heuristique ne trouve aucune contrainte violée. On résout le problème de séparation de ces contraintes selon la schéma suivant :

```

Procédure Sous_tour_minimal;
Début
  Si ( Sous_tour_minimal_2 == 0) alors
    Sous_tour_minimal_1;
  Fin Si
Fin.

```

4.3.1.2 Ensemble régulier

On rappelle qu'un ensemble S non vide est dit régulier si aucun pays ne rencontre à la fois S et $V \setminus S$. Pour trouver des contraintes de sous-tours violées associées à un ensemble régulier, on utilise une heuristique ou un algorithme exact d'identification des contraintes de sous-tours du TSP . En effet, une inéquation de sous-tour du TSP violée par le point contracté \bar{x}^c définit une contrainte de sous-tour régulier violée par \bar{x} , et, réciproquement, si aucune contrainte de sous-tour de TSP n'est violée par \bar{x}^c , alors aucune contrainte de sous-tour régulier n'est violée par \bar{x} .

On dispose de plusieurs algorithmes exacts pour la séparation de contraintes de sous-tours réguliers. Le plus connu est l'algorithme de Gomory et Hu [40] de complexité $O(|V|^2 |E| \log(|V|^2 / |E|))$, ou, pour un graphe dense, $O(|V|^4)$. Pratiquement, on utilise l'implantation réalisée par Padberg et Rinaldi [76] qui est de même complexité mais beaucoup plus rapide. Ils utilisent des opérations de contraction qui réduisent la taille du graphe avant d'appliquer l'algorithme de Gomory et Hu. Pour les problèmes de grande taille, ces algorithmes exigent un temps de calcul considérable. On préfère utiliser des heuristiques rapides [20, 44] et ne faire appel à l'algorithme exact que lorsque les heuristiques ne trouvent aucune contrainte violée. On désignera par *Sous_tour_régulier* une procédure globale qui résout le problème de séparation des contraintes de sous-tours réguliers violées qui intègre des heuristiques et un algorithme exact.

On s'intéressera à la recherche de contraintes de sous-tours réguliers vio-

lées que lorsqu'aucune contrainte de sous-tour associée à un ensemble minimal n'est violée. En effet, si on applique la procédure *Sous_tour_régulier* lorsque des contraintes de sous-tours associées à un ensemble minimal sont violées, elle peut générer des contraintes de sous-tours réguliers associées à un ensemble $S = W_1 \cup W_2$, avec W_1 et W_2 deux pays dont au plus un est dégénéré. Or, ce type d'inéquations n'induit pas une facette du polytope $ITSP(n, p)$.

4.3.1.3 Ensemble non régulier

Ici, on s'intéresse à des contraintes de sous-tours valides, non réguliers et de poids inférieur à 2. Ce type de contraintes présente des difficultés particulières. En effet, la coupe de poids minimum dans le graphe (\bar{G}, \bar{x}) obtenue en utilisant un algorithme classique, est généralement définie par un ensemble non valide. On ne peut pas appliquer un algorithme qui calcule une coupe de poids minimum directement à (\bar{G}, \bar{x}) pour déterminer une coupe valide de poids minimum.

Soit S un sous-ensemble de sommets. S'il existe deux pays W_i et W_j tels que $W_i \subset S$ et $W_j \subset V \setminus S$, alors la coupe $\delta(S)$ est dite une (W_i, W_j) -coupe. Une telle coupe est, par définition, une coupe valide.

Soient S^* un ensemble qui définit une coupe valide de poids minimum, et W_s et W_t deux pays tels que $W_s \subset S^*$ et $W_t \subset V \setminus S^*$. La coupe $\delta(S^*)$ est une (W_s, W_t) -coupe de poids minimum. Si, pour toute paire de pays W_i et W_j , on peut calculer une (W_i, W_j) -coupe de poids minimum, la coupe valide de poids minimum $\delta(S^*)$ sera la coupe de poids minimum parmi toutes ces coupes.

Dans ce qui suit, on décrira un algorithme qui, étant donnés deux pays W_i et W_j , détermine une (W_i, W_j) -coupe de poids strictement inférieur à 2 ou montre qu'une telle (W_i, W_j) -coupe n'existe pas.

Pour toute paire de pays W_i et W_j , on considère le graphe $\bar{G}_{i,j}$ obtenu à partir du graphe \bar{G} en contractant chacun des deux pays W_i et W_j en un sommet. On désignera par w_i et w_j les sommets de $\bar{G}_{i,j}$ représentant respectivement les pays W_i et W_j . Si les deux pays W_i et W_j sont dégénérés

alors le graphe $\bar{G}_{i,j}$ est exactement le graphe \bar{G} . On définit un vecteur $\bar{x}^{i,j}$ de l'espace des arêtes de $\bar{G}_{i,j}$ par :

$$\bar{x}_{u,v}^{i,j} = \begin{cases} \sum_{k \in W_i} \bar{x}_{k,v} & \text{si } u = w_i \text{ et } v \neq w_j \\ \sum_{l \in W_j} \bar{x}_{u,l} & \text{si } u \neq w_i \text{ et } v = w_j \\ \sum_{k \in W_i} \sum_{l \in W_j} \bar{x}_{k,l} & \text{si } u = w_i \text{ et } v = w_j \\ \bar{x}_{u,v} & \text{sinon} \end{cases}$$

Une $(w_i - w_j)$ -coupe dans le graphe $(\bar{G}_{i,j}, \bar{x}^{i,j})$ définit d'une manière unique une (W_i, W_j) -coupe dans le graphe (\bar{G}, \bar{x}) . De plus, les deux coupes ont une même valeur. Pour trouver une (W_i, W_j) -coupe de poids minimum dans le graphe (\bar{G}, \bar{x}) , on construit le graphe $(\bar{G}_{i,j}, \bar{x}^{i,j})$, puis on calcule une $(w_i - w_j)$ -coupe de poids minimum dans ce nouveau graphe.

On notera que, pratiquement, on définit le graphe $(\bar{G}_{i,j}, \bar{x}^{i,j})$ d'une autre manière. Pour tout pays W_i , on définit, un ensemble d'arêtes C_i tel que C_i est un cycle élémentaire reliant tous les sommets de W_i si $|W_i| \geq 3$, ou une arête reliant les deux sommets de W_i si $|W_i| = 2$, ou vide si W_i est dégénéré. Pour toute paire de pays W_i et W_j , le graphe $\bar{G}_{i,j}$ est le graphe obtenu à partir de \bar{G} en ajoutant les ensembles d'arêtes C_i et C_j associés respectivement à W_i et à W_j . Dans ce cas, on définit le vecteur $\bar{x}^{i,j}$ par :

$$\bar{x}_e^{i,j} = \begin{cases} \bar{x}_e & \text{si } e \in E \\ \alpha & \text{si } e \in C_i \\ \beta & \text{si } e \in C_j \end{cases}$$

avec

$$\alpha = \begin{cases} 1 & \text{si } |C_i| \geq 2 \\ 2 & \text{si } |C_i| = 1 \end{cases} \quad \text{et} \quad \beta = \begin{cases} 1 & \text{si } |C_j| \geq 2 \\ 2 & \text{si } |C_j| = 1 \end{cases}$$

Pour trouver une coupe valide de poids minimum dans le graphe (\bar{G}, \bar{x}) , on calcule, pour toute paire de pays W_i et W_j , une (W_i, W_j) -coupe de poids

minimum dans le graphe (\bar{G}, \bar{x}) . La coupe valide de poids minimum est une coupe de poids minimum parmi toutes ces coupes. L'algorithme qui suit détermine une ou plusieurs coupes valides violées. Lorsqu'il ne trouve aucune coupe violée, alors, dans le graphe (\bar{G}, \bar{x}) , aucune coupe valide n'est violée.

Procédure Sous_tour_non_regulier_1;

Début

Pour i allant de 1 à p faire

Pour j allant de 1 à p faire

Construire $G_{i,j}$ et $\bar{x}^{i,j}$ correspondants aux pays W_i et W_j ;

Calculer $\delta(S_{w_i, w_j}^*)$: une (w_i, w_j) -coupe de poids minimum dans le graphe $\bar{G}_{i,j}$;

Si $\bar{x}^{i,j}(\delta(S_{w_i, w_j}^*)) < 2$ alors

Enregistrer S_{w_i, w_j}^* (contrainte violée);

Fin Si

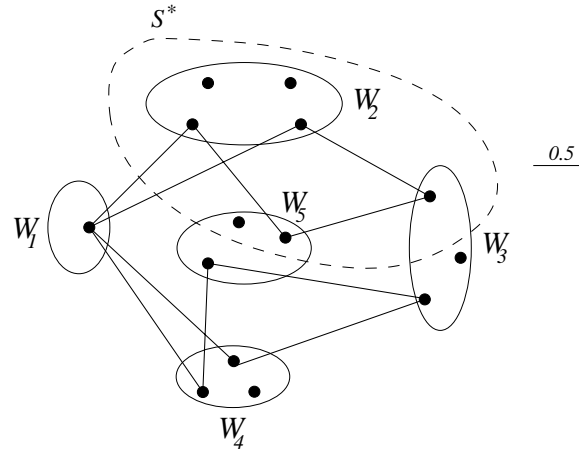
Fin Pour

Fin Pour

Fin.

Cette procédure est un algorithme exact pour la séparation des contraintes de sous-tours non réguliers et réguliers. Pour deux sommets quelconques k et l de $\bar{G}_{i,j}$, on peut déterminer une (k, l) -coupe de poids minimum dans le graphe $\bar{G}_{i,j}$ en $O(|V||\bar{E}|\log(|V|^2/|\bar{E}|))$ si on utilise un algorithme de flot maximum [38]. Ainsi, on obtient un algorithme pour la séparation des contraintes de sous-tours non réguliers en $O(p^2|V||\bar{E}|\log(|V|^2/|\bar{E}|))$, ou en $O(p^2n^3)$ pour un graphe dense. Notons qu'il existe un autre algorithme, développé par Nagamachi et Ibaraki [71, 70], qui calcule une (k, l) -coupe de poids minimum en $O(|V||\bar{E}| + |V|^2 \log |V|)$ sans calculer le flot maximum.

L'algorithme précédent peut générer plusieurs fois une coupe valide violée; ce qui peut créer une certaine dégénérescence dans la résolution du programme linéaire résultant. Dans l'exemple de la figure 4.2, l'ensemble S^* définit une (W_2, W_4) -coupe de poids minimum violée, mais aussi une (W_1, W_2) -coupe de poids minimum violée. Notons que le point fractionnaire correspondant ne viole aucune contrainte régulière, et en particulier de sous-tour, et aucune contrainte de sous-tour minimal.

FIG. 4.2 – Un graphe (\bar{G}, \bar{x}) .

Pour éviter de générer plusieurs fois une même coupe valide violée, il suffit de retenir, dans l'algorithme précédent, une coupe valide violée que lorsqu'elle n'est pas la (W_i, W_j) -coupe de poids minimum retenue pour une paire de pays W_i et W_j déjà considérée. Soit $S_{k,l}^*$ la coupe minimum trouvée en considérant la paire de pays W_{i_0} et W_{j_0} . Il faut calculer les ensembles $\mathcal{W}_{S_{k,l}^*}$ et $\mathcal{W}_{V \setminus S_{k,l}^*}$ des pays qui sont strictement contenus dans respectivement $S_{k,l}^*$ et $V \setminus S_{k,l}^*$. S'il existe une paire de pays W_i et W_j , déjà considérée, telle que $W_i \in \mathcal{W}_{S_{k,l}^*}$ et $W_j \in \mathcal{W}_{V \setminus S_{k,l}^*}$, et une (W_i, W_j) -coupe de poids minimum, violée et retenue, est également une (W_{i_0}, W_{j_0}) -coupe de poids minimum, alors il ne faut pas retenir l'ensemble $S_{k,l}^*$.

Pratiquement, on utilise cette approche d'une manière heuristique. En effet, étant donnée une (W_{i_0}, W_{j_0}) -coupe violée, il est très coûteux de vérifier exactement si elle a été retenue pour une paire de pays déjà considérée. On décidera de ne pas retenir une (W_{i_0}, W_{j_0}) -coupe violée si certaines conditions nécessaires pour qu'elle soit déjà retenue sont vérifiées.

Pour toute (W_i, W_j) -coupe violée et retenue, on enregistre la paire de pays W_i et W_j correspondante, la valeur de la coupe et la cardinalité du plus petit ensemble qui la définit. On définit des conditions nécessaires faciles à vérifier pour qu'une (W_{i_0}, W_{j_0}) -coupe violée, définie par un ensemble de sommets S_{i_0, j_0}^* , soit la coupe retenue pour une paire de pays W_i et W_j , définie par un

ensemble de sommets $S_{i,j}^*$. Nous avons retenu les trois conditions nécessaires suivantes :

- i) il faut que l'un des deux pays W_i et W_j soit strictement inclus dans S_{i_0,j_0}^* et l'autre dans $V \setminus S_{i_0,j_0}^*$;
- ii) il faut que la valeur de la coupe associée à l'ensemble S_{i_0,j_0}^* soit égale à celle de la coupe associée à l'ensemble $S_{i,j}^*$;
- iii) il faut que les cardinalités des ensembles S_{i_0,j_0}^* et $S_{i,j}^*$ soient égales si on suppose que $|S_{i_0,j_0}^*|$ et $|S_{i,j}^*|$ sont respectivement inférieures ou égales à $|V \setminus S_{i_0,j_0}^*|$ et $|V \setminus S_{i,j}^*|$.

On notera que ces conditions sont nécessaires mais non suffisantes. Elles garantissent qu'une coupe violée ne soit pas générée plus d'une fois. Cependant, on peut générer des coupes qui ont un degré de "ressemblance" élevé. Par expérience numérique, nous avons constaté que l'effet sur la valeur de la fonction économique de plusieurs coupes différentes violées qui ont un degré de "ressemblance" élevé est sensiblement le même que celui d'une seule coupe. Pratiquement, on sera plus exigeant. Au lieu de considérer des cardinalités égales dans la condition iii) précédente, on considère des cardinalités qui vérifient la relation suivante :

- iv) $|S_{i,j}^*| - k \leq |S_{i_0,j_0}^*| \leq |S_{i,j}^*| + k$, avec k une constante qu'on fixe à $\lceil p/4 \rceil$.

Si on intègre, dans la procédure *Sous_tour_non_regulier_1*, un module qui vérifie, chaque fois qu'on trouve une coupe violée, les trois conditions précédentes et cela pour toute paire de pays déjà considérée, on obtient une nouvelle procédure qu'on désignera par *Sous_tour_non_regulier_2*. On notera que la procédure *Sous_tour_non_regulier_2* est également un algorithme exact pour la séparation des contraintes de sous-tours non réguliers. Généralement, elle génère moins de contraintes qu'une procédure qui vérifie les conditions nécessaires et suffisantes pour qu'une coupe trouvée violée soit déjà retenue, mais elle est plus rapide.

En revanche, pour les instances de grande taille, cette procédure reste très coûteuse et exige un temps de calcul considérable. Pratiquement, on

utilise des heuristiques rapides et efficaces, et on ne fait appel à la procédure *Sous_tour_non_regulier_2* que lorsque les heuristiques ne trouvent aucune contrainte violée. Le principe d'heuristique que nous avons retenu consiste à limiter le nombre des paires de pays W_i et W_j pour lesquelles on calcule une (W_i, W_j) -coupe de poids minimum. On peut définir l'ensemble des paires de pays qu'on considère de plusieurs manières différentes. Dans la suite, on décrira une heuristique qui considère, à chaque itération, deux pays proches l'un de l'autre.

On dira qu'un pays W_i est proche d'un pays W_j s'il existe un sommet v de W_i tel W_j appartient à la liste *Sommet_Voisin*[v] définie dans 4.3.1.1. Ainsi, pour chaque pays W_i , on considère la liste, notée *Pays_Voisin*[W_i], des pays qui lui sont proche. On calcule une (W_i, W_j) -coupe de poids minimum pour W_j appartenant à la liste *Pays_Voisin*[W_i]. Pour éviter de calculer une même coupe deux fois, on considère un indice j supérieur à i . On désignera par *Sous_tour_non_regulier_3* la version de la procédure *Sous_tour_non_regulier_2* qui prend en compte cette restriction sur le calcul des (W_i, W_j) -coupes de poids minimum.

La séparation des contraintes de sous-tours non réguliers exige un temps de calcul et de la place mémoire considérables, contrairement à la séparation des contraintes de sous-tours minimaux et des sous-tours réguliers qui ne posent pas de problèmes particuliers. Très rapidement, on obtient un point fractionnaire qui ne viole aucune contraintes de ces deux types. C'est pour ces raisons, qu'il est très important d'obtenir un algorithme, rapide et efficace, pour la séparation des contraintes de sous-tours non réguliers.

La procédure *Sous_tour_non_regulier_3* peut générer un nombre très élevé de coupes violées de valeurs très différentes. Pratiquement, on arrête la séparation lorsque le nombre de coupes identifiées atteint une limite maximum. Dans la suite, on présente une procédure qui identifie des coupes dont les valeurs sont très proches les unes des autres.

On définit une variable *Seuil* réelle et globale qui prend ses valeurs entre 0 et 2, et initialisée à 2. Dans ce qui suit, on suppose que la procédure *Sous_tour_non_regulier_3* identifie des coupes valides de valeur strictement

inférieure à la valeur de la variable *Seuil* et que la valeur de la coupe minimum qu'elle identifie est dans le variable *CoupeMin*. L'objectif qu'on se fixe est de générer un nombre suffisamment élevé de coupe dont la valeur est proche de *CoupeMin*. L'algorithme de séparation des contraintes de sous-tours non réguliers que nous avons retenu est le suivant :

Procédure Sous_tour_non_régulier;

Début

Tant que (*Sous_tour_non_regulier_3* == 0) et (*Seuil* ≤ 2) Faire

Seuil = *Seuil* + α ;

Fin Tant que

Si *Seuil* > 2 alors /* i.e. *Sous_tour_non_regulier_3* = 0 */

Seuil = *Seuil* - α ;

Sous_tour_non_regulier_2 ;

Sinon /* On rectifie *Seuil* pour le prochain appel */

Si (*CoupeMin* < *Seuil* - 2 * α) et (*Seuil* > α)

alors *Seuil* = *Seuil* - α ;

Si (*CoupeMin* > *Seuil* - $\alpha/2$) et (*Seuil* ≤ 2 - α)

alors *Seuil* = *Seuil* + α ;

Fin Si

Fin.

On notera que la procédure *Sous_tour_non_régulier*, décrite ci-dessus, est appelée plusieurs fois et qu'au premier appel la valeur de la variable globale *Seuil* est égal à 2.

Dans la procédure *Sous_tour_non_régulier*, on augmente la valeur de *Seuil* d'une constante α lorsque la valeur de *CoupeMin* est jugée proche de *Seuil* et on diminue la valeur de *Seuil* de α lorsque la valeur de *CoupeMin* est jugée éloignée de *Seuil*. L'objectif de la variation de la variable *Seuil* est de générer, à chaque itération, des coupes violées de valeur proche de *CoupeMin*. L'introduction de la variable *Seuil* ne limite pas considérablement le nombre de coupes générées. En effet, si on suppose que dans l'algorithme la variable *CoupeMin* est croissante, la variable *Seuil* décroît pendant les premières itérations, puis elle croît pour atteindre la valeur 2 (voir figure 4.3).

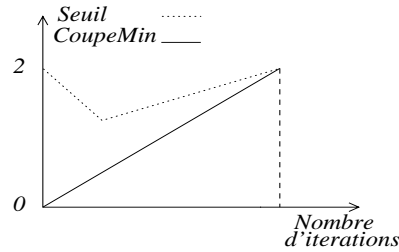


FIG. 4.3 –

Dans les premières itérations, le nombre de coupes violées est très élevé ; une petite valeur de *Seuil* ne limite pas considérablement le nombre de coupes générées. Pratiquement, on fixe α à 0.4.

L'algorithme pour la séparation des contraintes de sous-tours général peut s'exprimer comme suit :

```

Procédure Sous_tour;
Début
  Si Sous_tour_minimal == 0 alors
    Si Sous_tour_régulier == 0 alors
      Sous_tour_non_régulier;
    Fin Si
  Fin Si
Fin.

```

4.3.2 Contraintes de peigne

Un peigne est un arbre de cliques admettant un seul manche. Si H représente le manche et T_1, \dots, T_t représentent les dents, la contrainte de peigne correspondante est :

$$x(\delta(H)) + \sum_{i=1}^t x(\delta(T_i)) \geq 3t + 1 \quad (4.1)$$

La séparation des ces contraintes est difficile. Il n'existe pas un algorithme polynomial pour résoudre le problème de séparation correspondant.

Dans ce qui suit, on citera les différentes heuristiques de séparation des peignes du $TSP(p)$, et, par conséquent, des contraintes de peignes réguliers du $ITSP(n, p)$. Puis, on présentera une heuristique pour la séparation d'une classe particulière des contraintes de peignes non réguliers : les contraintes de 2-couplages non réguliers. Pour finir, on décrira une heuristique pour l'identification de contraintes de peignes non réguliers.

4.3.2.1 Peigne régulier

Pour la séparation des peignes réguliers, on utilise une procédure pour la séparation des peignes du TSP appliquée au graphe contracté (\bar{G}_c, \bar{x}_c) .

Il n'existe pas un algorithme polynomial en n pour la séparation des contraintes de peignes. En revanche, si on fixe le nombre de dents à k , Carr [12] a montré qu'on peut résoudre le problème de séparation des contraintes de peignes à k dents par un algorithme polynomial en n . À notre connaissance, il n'existe pas encore une implantation pratique de cet algorithme.

Le seul algorithme exact disponible résout le problème de séparation de la classe particulière des contraintes de 2-couplages. Plusieurs auteurs ont montré que ce problème est équivalent au problème de la coupe impaire de poids minimum suivant : étant donné un graphe, où chaque sommet est étiqueté paire ou impaire, trouver une coupe impaire de poids minimum, sachant qu'une coupe est dite impaire si elle est définie par un ensemble de sommets admettant un nombre impair de sommets étiquetés impair. Dans [75], Padberg et Rao proposent un algorithme polynomial pour résoudre ce problème. Pratiquement, pour obtenir un algorithme de séparation rapide, on utilise des opérations (voir [74, 43, 77]) qui réduisent la taille du graphe avant d'appliquer l'algorithme de Padberg et Rao.

Dans [77, 44, 17], des heuristiques pour la séparation des contraintes de peigne général sont proposées. Le principe de la plupart d'entre elles consiste à construire une contrainte de peigne général violée à partir d'une contrainte de 2-couplage violée en utilisant des opérations d'extension par "zero node-lifting" [56]. On désignera par *Peigne_1* une heuristique pour la séparation de contraintes de peignes violées.

4.3.2.2 Contraintes de 2-couplage non réguliers

On rappelle qu'une contrainte de 2-couplage est une contrainte de peigne telle que chaque dent est l'union d'exactly deux pays. Elle est régulière ou non régulière selon que le manche est un ensemble régulier ou non régulier.

Soit \mathcal{W}_1 l'ensemble des pays qui sont incidents à une arête e de poids 1 dans le graphe (\bar{G}_c, \bar{x}^e) . On note \mathcal{L}_1 l'ensemble des contraintes de 2-couplages non réguliers dont chaque dent est constituée de deux pays de \mathcal{W}_1 et admet un cocycle de valeur égal à 2.

L'heuristique que nous proposons résout le problème de séparation des contraintes de l'ensemble \mathcal{L}_1 en l'algorithme de Padberg et Rao [75] pour le calcul d'une coupe impaire de poids minimum.

Soit le graphe G_1 obtenu à partir du graphe (\bar{G}, \bar{x}) en contractant chaque pays de \mathcal{W}_1 en un sommet. On note $V_1' = V \setminus \bigcup_{W_i \in \mathcal{W}_1} W_i$ et V_1'' le sous-ensemble des sommets de G_1 représentant les pays contractés. Ainsi, l'ensemble des sommets de G_1 est $V_1 = V_1' \cup V_1''$. Si un sommet j de V_1'' représente le pays W_j de \mathcal{W}_1 , on définit deux points x^1 et \bar{x}^1 de l'espace des arêtes de G_1 par :

$$x_{i,j}^1 = \begin{cases} \bar{x}_{i,j} & \text{si } i \in V_1' \text{ et } j \in V_1' \\ \sum_{k \in W_j} \bar{x}_{i,k} & \text{si } i \in V_1' \text{ et } j \in V_1'' \\ \sum_{l \in W_i} \sum_{k \in W_j} \bar{x}_{l,k} & \text{si } i \in V_1'' \text{ et } j \in V_1'' \end{cases} \quad \text{et} \quad \bar{x}_{i,j}^1 = \begin{cases} 0 & \text{si } x_{i,j}^1 = 1 \\ x_{i,j}^1 & \text{sinon} \end{cases}$$

On associe à chaque sommet de V_1 , incident à exactement une arête de poids 1 dans le graphe (G_1, x^1) , une étiquette impaire et, pour les autres sommets, une étiquette paire. Soit S^* un sous-ensemble de sommets de V_1 qui définit une coupe impaire de (G_1, \bar{x}^1) de poids minimum. Si on suppose qu'aucune contrainte de sous-tour et de 2-couplage régulier n'est violée par le point \bar{x} , on obtient la proposition suivante :

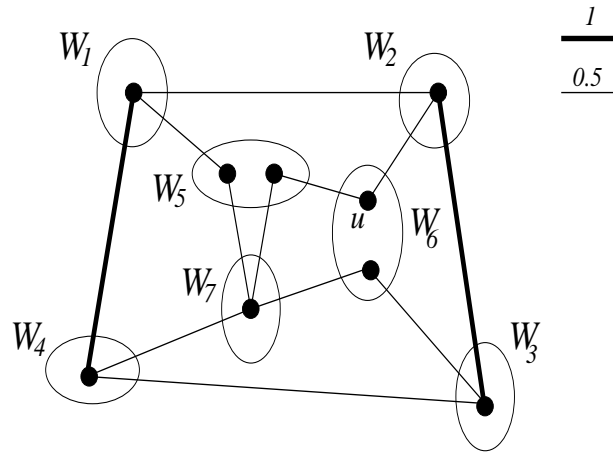
Proposition 4.3.1 $\bar{x}^1(\delta(S^*)) < 1$ si et seulement si il existe une contrainte de \mathcal{L}_1 violée par le point \bar{x} .

Preuve : Supposons que $\bar{x}^1(\delta(S^*)) < 1$. Soient v_1, \dots, v_t les sommets impairs de S^* adjacents, dans le graphe (G_1, x^1) , respectivement aux sommets impairs u_1, \dots, u_t de $V \setminus S^*$ par des arêtes de poids égal à 1. Le nombre t est impair, et, de plus, il est supérieur à 1. En effet, si t est égal à 1, alors $\bar{x}(\delta(S^*)) < 2$. Dans le graphe (\bar{G}, \bar{x}) , soit le peigne de manche $H = S^*$ et de dents $T_i = W_{v_i} \cup W_{u_i}$, pour $i = 1, \dots, t$. La contrainte correspondante au peigne ainsi défini appartient à l'ensemble \mathcal{L}_1 et elle est violée par le point \bar{x} .

Réciproquement, soit une contrainte de \mathcal{L}_1 , définie par un manche H et un ensemble de dents $\mathcal{T} = \{T_1, \dots, T_t\}$, qui est violée par le point \bar{x} . On suppose que, pour $i = 1, \dots, t$, $W_i = T_i \cap H$ n'est pas adjacent, dans le graphe (G_1, \bar{x}^1) , à un pays de H par une arête de poids 1. En effet, s'il existe une dent T_{i_0} telle que $W_{i_0} = T_{i_0} \cap H$ est adjacent, dans le graphe (G_1, \bar{x}^1) , à un pays W_{j_0} de H par une arête de poids 1, on considère le peigne défini par le manche $H' = H \setminus W_{i_0}$ et l'ensemble des dents $\mathcal{T}' = \mathcal{T} \setminus \{T_{i_0}\} \cup \{T'_{i_0}\}$, avec $T'_{i_0} = W_{i_0} \cup W_{j_0}$. La contrainte correspondante au peigne ainsi défini est violée par le point \bar{x} . Soit H_1 l'ensemble des sommets de G_1 correspondant au manche H . Dans le graphe (\bar{G}, \bar{x}) , le cocycle de chaque dent a une valeur égale à 2 et $\bar{x}^1(\delta(H_1)) \leq \bar{x}(\delta(H)) - t$. Une condition nécessaire pour que la contrainte soit violée est $\bar{x}(\delta(H)) < t + 1$, alors on aura $\bar{x}^1(\delta(H_1)) < 1$. Soient I' l'ensemble des sommets impairs de H_1 deux à deux adjacents, dans le graphe (G_1, \bar{x}^1) , par une arête de poids 1 et I'' l'ensemble des sommets impairs de H_1 adjacents, dans le graphe (G_1, \bar{x}^1) , à un sommet n'appartenant pas à l'ensemble H_1 par une arête de poids 1. Chaque sommet de I'' représente un pays appartenant à une dent du peigne violé. L'ensemble des sommets impairs de H_1 est $I = I' \cup I''$ et le nombre de ces sommets est impair. Ainsi, $\bar{x}^1(\delta(H_1))$ est une coupe impaire de poids inférieure à 1. \square

L'heuristique qui découle de ce résultat consiste à construire le graphe (G_1, \bar{x}^1) étiqueté, puis lui appliquer un algorithme qui calcule une ou plusieurs coupes impaires de poids strictement inférieures à 1. On désignera une telle procédure par *Peigne_2*.

Le point fractionnaire de l'exemple de la figure 4.4 vérifie toutes les contraintes régulières (le point correspondant \bar{x}^c est une combinaison convexe des sommets du polytope *TSP* (7)) et toutes les contraintes de sous-tours non régulières. L'heuristique, décrite ci-dessus, identifie la contrainte violée de 2-

FIG. 4.4 – Un graphe (\bar{G}, \bar{x}) .

couplage non régulière définie par le manche $H = W_1 \cup W_2 \cup W_5 \cup \{u\}$ et les dents $T_1 = W_1 \cup W_4$, $T_2 = W_5 \cup W_7$ et $T_3 = W_2 \cup W_3$.

4.3.2.3 Peigne non régulier

Le principe d'heuristique que nous avons retenu consiste à construire à partir d'un peigne régulier violé un ou plusieurs peignes non réguliers violés.

Soit un peigne régulier de manche H et de dents T_1, \dots, T_t tel que la contrainte correspondante soit violée par le point \bar{x} . On désigne par \mathcal{P} l'ensemble $\{H, T_1, \dots, T_t\}$ et $v_{\mathcal{P}}$ l'écart entre le membre de gauche et le membre de droite de la contrainte de peigne correspondante ; i.e. $v_{\mathcal{P}} = 3t + 1 - \sum_{P \in \mathcal{P}} \bar{x}(\delta(P))$. Soient P un élément de \mathcal{P} et S_P un sous-ensemble non régulier de sommets qui ne rencontre aucun élément de \mathcal{P} . On définit l'ensemble $\mathcal{P}' = \mathcal{P} \setminus \{P\} \cup \{P'\}$, avec $P' = P \cup S_P$. L'ensemble \mathcal{P}' définit un peigne non régulier violé si et seulement si $\bar{x}(\delta(S_P)) - 2\bar{x}(P : S_P) < v_{\mathcal{P}}$. L'heuristique consiste à essayer de trouver, pour chaque élément de \mathcal{P} , un sous-ensemble S_P non régulier qui ne rencontre aucun élément de \mathcal{P} tel que la contrainte non régulière correspondante à \mathcal{P}' soit violée.

Il n'est pas facile de trouver un sous-ensemble S_P non régulier quelconque qui engendre une contrainte violée, pour un certain élément de \mathcal{P} . Pratique-

ment, on ne considère que des sous-ensembles S_P à un seul sommet pour obtenir une heuristique rapide qui peut s'exprimer comme suit :

Procédure Peigne-3;

Début

Soit $\mathcal{P} = \{T_0, T_1, \dots, T_t\}$ un peigne violé; $/*T_0$ est le manche $\star/$

$V_0 = V \setminus \cup_{i=0}^t T_i$;

Pour i allant de 0 à t faire

$V_1 = V_0$;

Tant que $V_1 \neq \emptyset$ faire

Soit $s \in V_1$;

$T'_i = T_i \cup \{s\}$ et $\mathcal{P}' = \mathcal{P} \setminus \{T_i\} \cup \{T'_i\}$;

Si le peigne \mathcal{P}' définit une contrainte violée alors

Enregistrer \mathcal{P}'

Fin Si

$V_1 = V_1 \setminus \{s\}$;

Fin Tant que

Fin Pour

Fin.

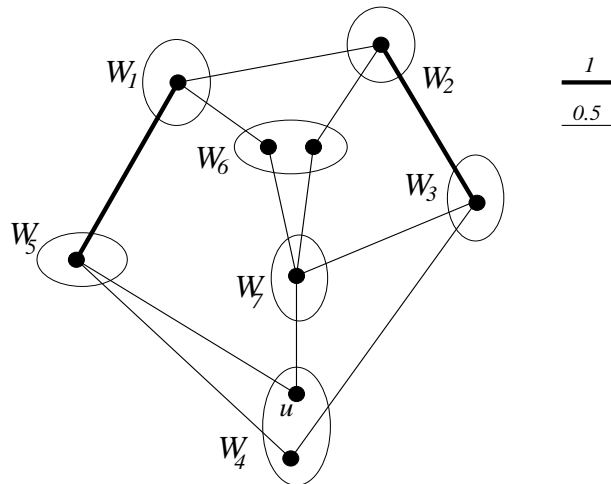


FIG. 4.5 – Un graphe (\bar{G}, \bar{x}) .

Le point fractionnaire de l'exemple de la figure 4.5 vérifie toutes les contraintes de sous-tours régulières et non régulières. Un algorithme pour la séparation des contraintes de 2-couplages régulières identifie la contrainte de 2-couplage violée et définie par le manche $H = W_1 \cup W_2 \cup W_6$ et les dents $T_1 = W_1 \cup W_5$, $T_2 = W_6 \cup W_7$ et $T_3 = W_2 \cup W_3$. L'heuristique *Peigne_3*, décrite ci-dessus, construit à partir du peigne violé précédent une contrainte de peigne non régulière violée et définie par le manche $H' = H$ et les dents $T'_1 = T_1$, $T'_2 = T_2 \cup \{u\}$ et $T'_3 = T_3$.

On peut penser que la procédure *Peigne_3* n'a aucun intérêt et que les contraintes de peignes non régulières violées qu'elle identifie contribuent très peu dans la résolution du programme linéaire. Ceci peut effectivement arriver dans certains cas. La procédure *Peigne_3* est rapide et génère plusieurs contraintes violées. Il n'y a pas d'inconvénient dans son utilisation, bien au contraire, les contraintes qu'elle identifie peuvent améliorer sensiblement la fonction économique dans certains cas.

L'algorithme de séparation de contraintes de peignes que nous avons retenu est le suivant :

```

Procédure Peigne;
Début
  Si Peigne_1 > 0 alors
    Peigne_3
  Fin Si
  Peigne_2
Fin.

```

Si dans la procédure *Peigne_3*, on fait varier l'indice i de 1 à t et, dans *Peigne_2*, on ne considère que des coupes impaires non régulières, alors un peigne violé ne peut pas être généré par plus d'une procédure parmi les trois procédures *Peigne_1*, *Peigne_3* et *Peigne_2*. Ainsi, en appliquant successivement ces trois procédures pour identifier un nombre maximum de contraintes de peignes violées, on ne risque pas de générer des contraintes redondantes.

Des heuristiques pour la séparation d'autres contraintes régulières ont

été également proposées. Dans [77], Padberg et Rinaldi proposent deux heuristiques pour la séparation des contraintes d'arbre de cliques et, dans [16, 17], Clochard et Naddef proposent des heuristiques pour la séparation des contraintes de chemins.

4.4 Instances et résultats numériques

Les résultats numériques qui suivent ont été obtenus en appliquant notre code à des instances du *ITSP* obtenues à partir d'instances du *TSP* de la librairie des problèmes tests de Reinelt [82] accessibles par la page *WEB* : <http://www.iwr.uni-heidelberg.de/iwr/comopt/soft/TSPLIB95/TSPLIB.html>.

4.4.1 Génération d'instances

Pour générer des instances du *ITSP*, on a utilisé une procédure de Fischetti et al. [28] qui partitionne l'ensemble des sommets d'une instance euclidienne du *TSP* pour simuler des régions géographiques.

Pour une instance du *TSP* de taille n , le nombre de pays est fixé à $p = \lceil n/5 \rceil$. On associe à chaque pays W_i un sommet v_i appelé centre de W_i qui correspondra au premier sommet appartenant à W_i . L'ensemble des centres est choisi de manière à ce qu'ils soient le plus éloignés possible les uns des autres. Chacun des autres sommets est affecté au pays dont le centre est le plus proche. Pour une instance donnée du *TSP*, cette procédure peut générer des instances différentes du *ITSP* différentes si on fait varier le nombre de pays.

4.4.2 Résultats numériques

Dans ce qui suit, on donnera des résultats numériques obtenu par notre implantation sur un jeu d'essais de 23 instances et on comparera certaines stratégies entre elles. Le solveur de programme linéaire qu'on a utilisé est le logiciel *Cplex 3.0* [19].

On notera que toutes les instances présentées ont été résolues au nœud racine, i.e. sans branchement. Ceci ne veut pas dire qu'on résout toutes les

Problème	Opt	Colonne	Ligne	Sous-tour	Peigne	Tps CPU
10att48	5394	153	70	703	1	51.9
11eil51	174	1013	86	314	1	24.3
16eil76	209	345	138	1018	2	200.4
16pr76	64925	320	133	1580	4	341.3
20rat99	497	428	162	867	2	158.6
20kroA100	9711	288	159	1061	2	154.6
20kroB100	10328	195	150	921	0	134.8
20kroC100	9554	269	159	843	0	135.0
20kroD100	9450	455	143	862	0	135.2
20kroE100	9523	214	141	636	2	72.6
20rd100	3650	437	186	1341	0	288.8
21lin105	8213	401	169	761	0	111.7
22pr107	27898	665	120	225	0	12.7
28pr136	42570	274	208	1291	0	237.7
29pr144	45886	1384	142	222	0	28.1
30kroA150	11018	653	241	1428	10	291
30kroB150	12196	334	212	1709	8	356.8
31pr152	51576	240	516	12339	121	7677.2
40kroA200	13406	588	279	2976	20	916.3
40kroB200	13111	650	300	1778	2	623.0
46pr226	64007	637	287	14066	18	2784.7
53pr264	29549	553	357	3113	19	1308.5
60pr299	22615	1115	457	3745	26	1733.1

TAB. 4.1 – Résultats numériques.

instances sans branchement. Bien au contraire, le résolution de beaucoup d'instances s'effectue avec branchement et on ne sait pas encore résoudre d'autres instances.

On se limite à présenter des résultats numériques concernant des instances résolues sans branchement parce que notre objectif consiste à analyser les résultats de la séparation des contraintes, et, plus particulièrement, les contraintes de sous-tours. On notera qu'on a utilisé les procédures *Sous-tour*, *Peigne_1* et *Peigne_3* pour la séparation des contraintes de sous-tours et de peignes.

Les colonnes 1 et 2 de la table 4.1 représentent respectivement le nom et la valeur de la solution optimale de l'instance. Le nom de l'instance est la juxtaposition du nombre de ces pays et du nom de l'instance du *TSP* qui l'a engendrée. Par exemple, l'instance 10att48 a 10 pays et 48 villes.

Les colonnes 3 et 4 de la même table représentent respectivement le nombre de variables et le nombre de contraintes dans le programme linéaire final. Le nombre total de contraintes de sous-tours générées est représenté dans la colonne 5 et le nombre total de contraintes de peignes générées est représenté dans la colonne 6. La dernière colonne représente le temps de calcul total exprimé en secondes de CPU pour une station SUN SPARC 10 – 20.

L'analyse de ces résultats numériques montre le nombre total élevé de contraintes de sous-tours générées pour chaque instance ce qui explique les temps de calcul plus élevés que ceux obtenus par Fischetti et al. dans [27]. On ne sait pas si ces résultats sont dûs au fait que les contraintes sont exprimées dans l'espace des arêtes ou à l'algorithme de séparation des contraintes de sous-tours qu'on utilise. Toutefois, nous pensons que trouver un algorithme pour la séparation des contraintes de sous-tours nettement plus performant que celui qu'on utilise est difficile.

Lors de nos expériences numériques, nous avons également remarqué que notre implantation est très peu stable. Pour certaines instances, si on modifie légèrement la valeur d'un paramètre ou l'appel d'une procédure alors les résultats varient énormément. Cette remarque est également constatée dans

la résolution du *TSP* par un algorithme de branchement et de coupe.

Dans cette expérience numérique, nous discutons également certains choix de l'implantation. Nous avons retenu deux aspects : l'introduction dans le programme linéaire des bornes 0 et 1 pour chaque variable et l'utilisation de la variable *Seuil* pour la séparation des contraintes de sous-tours. Si on désigne par *A* l'implantation initial, on désignera par *B* et *C* les implantations obtenues à partir de *A* en considérant respectivement des variables bornées et une séparation des contraintes de sous-tours sans l'utilisation de la variable *Seuil*.

Pour chacune des ces trois implantations, nous donnons, dans la table 4.2, le nombre total de contraintes générées et le temps de calcul total exprimé en secondes de CPU pour une station SUN SPARC 10 – 20.

On notera que l'implantation *A* est plus rapide que les implantations *B* et *C* dans 74% des instances.

Problème	A		B		C	
	Contr	Tps CPU	Contr	Tps CPU	Contr	Tps CPU
10att48	704	51.9	752	65	736	58.8
11eil51	315	24.3	400	32.8	384	26.3
16eil76	1020	200.4	1167	225.5	1310	251.9
16pr76	1584	341.3	1538	328.5	1733	348.4
20rat99	869	158.6	871	162.5	1364	197.8
20kroA100	1063	154.6	992	148.4	1437	160.6
20kroB100	921	134.8	920	143.4	1109	115.99
20kroC100	843	135.0	924	137.6	1151	139.7
20kroD100	862	135.2	964	167.4	1586	241.9
20kroE100	638	72.6	614	74.5	790	98.7
20rd100	1341	288.8	1499	308.5	3032	540.2
21lin105	761	111.7	818	118.9	1267	163.4
22pr107	225	12.7	224	12.4	225	13.3
28pr136	1291	237.7	1164	245.7	1652	204.9
29pr144	222	28.1	213	23.2	222	23.5
30kroA150	1438	291.0	1729	329.4	1933	267.7
30kroB150	1717	356.8	1696	397.8	2302	384.7
40kroA200	2996	916.36	3047	878.7	3774	966.2
40kroB200	1780	623.0	1860	617.5	2816	527.5
46pr226	14084	2784.7	15776	4114.2	12534	3389.3
53pr264	3132	1308.5	2943	1256	3514	880.8
60pr299	3771	1733.1	3326	1349.3	6741	2369.2

TAB. 4.2 – Résultats numériques pour différentes variantes.

Conclusion

Ce travail se propose d'étudier une généralisation du problème du voyageur de commerce appelée problème du voyageur de commerce international et notée *ITSP*. Les principales contributions sont à la fois d'ordre pratiques et théoriques.

Dans une première partie, nous décrivons plusieurs types d'heuristiques pour le *ITSP* et nous donnons une réduction polynomiale du *ITSP* au *TSP*. Nous donnons également une nouvelle formulation du *ITSP* en un programme linéaire en nombres entiers.

La deuxième partie de ce travail est réservée à l'approche polyédrale. L'objectif principal étant de réaliser un algorithme de branchement et de coupe pour la résolution du *ITSP*.

Nous avons étudié le polyèdre de la relaxation graphique du *ITSP* et sa relation avec le polytope du *ITSP*. Nous montrons en particulier que les résultats connus pour le *TSP* sont valables pour le *ITSP*.

Nous distinguons deux types de facettes du polytope du *ITSP*: les facettes régulières et les facettes non régulières. Nous donnons des résultats sur la relation polyédrale qui existe entre le *TSP* et le *ITSP* et nous introduisons plusieurs classes d'inéquations induisant des facettes. Nous donnons également des résultats sur quelques propriétés que vérifient les facettes du polytope du *ITSP* et des résultats sur le "lifting" qui permettent d'obtenir de nouvelles facettes.

Nous présentons de nombreuses heuristiques et algorithmes exacts pour la séparation des inéquations de sous-tours et de peignes qui nous ont permis d'implanter un algorithme de branchement et de coupe pour la résolution du *ITSP* à partir d'un tel algorithme pour le *TSP*. Nous finissons ce travail en donnant quelques résultats numériques et une analyse de notre implémentation.

Les perspectives de ce travail sont nombreuses et elles sont d'ordre théoriques et algorithmiques. Sur le plan théorique, on ne sait toujours pas si toutes les facettes du *ITSP* sont *tsp*-fortes et on ne connaît pas la description linéaire complète du polytope du *ITSP* pour les petites valeurs du nombre de pays. On peut également étudier la généralisation de certaines classes de facettes du *TSP* pour le *ITSP*. Sur le plan algorithmique, on peut tester d'autres heuristiques pour la séparation des contraintes de sous-tours et de peignes.

Bibliographie

- [1] Aarts (E. H. L.), Van Laarhoven (P. J. M.), Lenstra (J. K.) et Ulder (N. L. J.). – A computational study of local search algorithms for job shop scheduling. *ORSA Journal on Computing*, vol. 6, n° 2, 1994.
- [2] Applegate (D.), Bixby (R.), Chvátal (V.) et Cook (W.). – Special session on tsp. *In: 15th International Symposium on Mathematical Programming*. – University of Michigan, USA.
- [3] Araque (J. R.), Kudva (G.), Morin (T.) et Pekny (J.). – A branch and cut for vehicle routing problems. *Annals of Operations Research*, vol. 50, 1994, pp. 37–59.
- [4] Augerat (P.). – *Approche Polyédrale du Problème de Tournées de Véhicules*. – Thèse de PhD, ARTEMIS-IMAG, Institut National Polytechnique de Grenoble, 1995.
- [5] Bachem (A.) et Grötschel (M.). – New aspects of polyhedral theory. *In: Modern Applied Mathematics, Optimization and Operations research*, éd. par Korte (B.). – North-Holland, Amsterdam.
- [6] Balas (E.). – The prize collecting traveling salesman problem. *Network*, vol. 19, 1989, pp. 621–636.
- [7] Balas (E.). – *The Prize Collecting Traveling Salesman Problem: II. Polyhedral Results*. – Rapport technique, Carnegie Mellon University, July 1993. Rapport de travail.
- [8] Balas (E.). – New classes of efficiently solvable generalized traveling salesman problem. *In: Symposium on Combinatorial Optimization*. –

- Imperial College of Science, Technology and medicine, London, March 1996.
- [9] Bellman (R.). – *Dynamic Programming*. – Princeton University Press, Princeton, NJ, 1957.
- [10] Bovet (J.). – The selective traveling salesman problem. *In: EURO VI Conference, Vienna*.
- [11] Boyd (S.), Cunningham (W.), Queyranne (M.) et Wang (Y.). – Ladders for travelling salesman problem. – 1993. Prprint, Department of Combinatorics and Optimization, University of Waterloo, Canada.
- [12] Carr (R.). – Separating clique tree and bipartition inequalities in polynomial time. *In: 4th Conference On Integer Programming and Combinatorial Optimization*. pp. 446–461. – Springer, Verlag.
- [13] Christofides (N.). – *Worst Case Analysis of a New Heuristic for the Traveling Salesman Problem*. – Rapport technique, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, 1976. Rapport 388.
- [14] Chvátal (V.). – *Linear Programming*. – Freeman, New York, 1983.
- [15] Clarke (G.) et Wright (J.). – Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, vol. 12, 1964, pp. 568–581.
- [16] Clochard (J.) et Naddef (D.). – Using path inequalities in a branch and cut code for the symmetric traveling salesman problem. *In: 3rd Conference On Integer Programming and Combinatorial Optimization, Core, Louvain-La-Neuve*, éd. par Rinaldi (G.) et Wolsey (L.), pp. 291–311.
- [17] Clochard (J.) et Naddef (D.). – *Some fast and effecient heuristics for comb separation in the symmetric traveling problem*. – Rapport technique, Institut IMAG, Grenoble, 1994.
- [18] Cornuéjols (G.), Fonlupt (J.) et Naddef (D.). – The traveling salesman problem on a graph and some related polyhedra. *Mathematical Programming*, vol. 33, 1985, pp. 1–27.

-
- [19] CPLEX Optimization Inc. – *Using The cplex callable library and cplex mixed integer library*, 1993.
- [20] Crowder (H.) et Padberg (M.). – Solving large-scale symmetric traveling salesman problems to optimality. *Management Science*, vol. 26, 1980, pp. 495–509.
- [21] Dantzig (G.), Fulkerson (D.) et Johnson (S.). – Solution of a large scale traveling salesman problem. *Operations Research*, vol. 2, 1954, pp. 393–410.
- [22] Dantzig (G.), Fulkerson (D.) et Johnson (S.). – On a linear programming, combinatorial approach to the traveling salesman problem. *Operations Research*, vol. 7, 1959, pp. 58–66.
- [23] Davis (L.) (édité par). – *Genetic algorithms and simulated annealing*. – Pitman, Londres, 1987.
- [24] Davis (L.) (édité par). – *Handbook of genetic algorithms*. – Van Nostrand Reinhold, New York, 1991.
- [25] Eastman (W.). – *Linear Programming with Pattern Constraints*. – Thèse de PhD, Harvard University, Cambridge, MA., 1958.
- [26] Edmonds (J.). – Maximum matching and a polyhedron with 0,1-vertices. *Journal of Research of the National Bureau of Standards B*, vol. 69, 1965, pp. 125–130.
- [27] Fischetti (M.), Salazar (G.) et Toth (P.). – The symmetric generalized traveling salesman polytope. *Networks*, vol. 26/2, 1995, pp. 113–123.
- [28] Fischetti (M.), Salazar (G.) et Toth (P.). – A branch and cut algorithm for the symmetric generalized traveling salesman problem. – 1996. à paraître dans *Operations research*.
- [29] Fischetti (M.) et Toth (P.). – An additive approach for the optimal solution of the prize collecting traveling salesman problem. *In: Vehicle routing: methods and studies*, éd. par Golden (B.) et A. (A.). pp. 319–343. – Elsevier, Amsterdam.

- [30] Gàcs (M.) et Lovász (L.). – Khachiyan’s algorithm for linear programming. *Mathematical Programming Study*, vol. 14, 1981, pp. 61–68.
- [31] Gavish (K.) et Srikanth (K.). – *Efficient Branch and Bound Code for Solving Large Scale Traveling Salesman Problems to Optimality*. – Rapport technique n° QM8329, Graduate School of Management, University of Rochester, 1983. Rapport de travail.
- [32] Giles (R.). – *Submodular Functions, Graphs, and Integer Polyhedra*. – Thèse de PhD, University of Waterloo, 1975.
- [33] Glover (F.). – Tabu search - part i. *Operations Research Society of America Journal on Computing 1*, 1989, pp. 190–206.
- [34] Glover (F.). – Tabu search - part ii. *Operations Research Society of America Journal on Computing 2*, 1990, pp. 4–32.
- [35] Glover (F.). – Ejection chains, reference structures and alternating path : Methods for traveling salesman problems. – April 1992. Rapport de travail.
- [36] Glover (F.) et Punnen (A. P.). – The traveling salesman problem : Linear time heuristics with exponential combinatorial leverage. – January 1994. Rapport de travail.
- [37] Glover (F.), Taillard (E.) et de Werra (D.). – A user’s guide to tabu search. *In: Annals of Operations Research*, éd. par Glover (F.), Laguna (M.), Taillard (E.) et de Werra (D.), pp. 3–28.
- [38] Goldberg (A.) et Tarjan (R.). – A new approach to the maximum flow problem. *Journal of the ACM*, vol. 35, 1988, pp. 921–940.
- [39] Golden (B.) et Stewart (W.). – Empirical analysis of heuristics. *In: The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*, éd. par Lawler (E.), Lenstra (J.), Rinnooy Kan (A.) et D.B. (S.). pp. 210–249. – Wiley.
- [40] Gomory (R.) et Hu (T.). – Multi-terminal network flows. *SIAM Journal on Applied Mathematics*, vol. 9, 1961, pp. 551–570.

-
- [41] Grötschel (M.). – The monotone 2-matching polytope on a complete graph. *Operations Research*, vol. 24, 1970, pp. 72–84.
- [42] Grötschel (M.). – *Polyedrische Charakterisierungen Kombinatorischer Optimierungsprobleme*. – Thèse de PhD, Hain, Meisenheim am Glam, 1977.
- [43] Grötschel (M.) et Holland (O.). – A cutting plane algorithm for minimum perfect 2-matching. *Computing*, vol. 39, 1987, pp. 327–344.
- [44] Grötschel (M.) et Holland (O.). – Solution of large-scale symmetric traveling salesman problems. *Mathematical Programming*, vol. 51, 1991, pp. 141–202.
- [45] Grötschel (M.), Lovász (L.) et Schrijver (A.). – The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, vol. 1, 1981, pp. 169–197.
- [46] Grötschel (M.), Lovász (L.) et Schrijver (A.). – *Geometric Algorithms and Combinatorial Optimization*. – Springer-Verlag, Berlin-Heidelberg, 1988.
- [47] Grötschel (M.) et Padberg (M.). – On the symmetric traveling salesman problem i. *Mathematical Programming*, vol. 16, 1979, pp. 265–280.
- [48] Grötschel (M.) et Padberg (M.). – On the symmetric traveling salesman problem ii. *Mathematical Programming*, vol. 16, 1979, pp. 281–302.
- [49] Grötschel (M.) et Pulleyblank (W.). – Clique tree inequalities and the symmetric traveling salesman problem. *Mathematics of Operations Research*, vol. 11, 1986, pp. 537–569.
- [50] Grünbaum (B.). – *Convex Polytopes*. – Wiley, London, 1967.
- [51] Hao (J.) et Orlin (J.). – A faster algorithm for finding the minimum cut in a graph. In: *Proceedings of the 3rd ACM-SIAM Symposium on Discrete Algorithms, Orlando, Florida*, pp. 165–174.
- [52] Held (M.) et Karp (R.). – The traveling salesman problem and minimum spanning trees. *Operations Research*, vol. 18, 1970, pp. 1138–1162.

- [53] Henry-Labordere (A.). – The record balancing problem: A dynamic programming solution of a generalized travelling salesman problem. *RIRO*, vol. B-2, 1969, pp. 43–49.
- [54] Holland (J. H.). – *Adaptations in natural and artificial systems*. – presses de l'Université de Michigan, Ann Arbor, 1976.
- [55] Johnson (D.). – Local optimization and the traveling salesman problem. In: *17th Colloquium on Automata, Languages and Programming*. pp. 446–461. – Springer, Verlag.
- [56] Jünger (M.), Reinelt (G.) et Rinaldi (G.). – *The Traveling Salesman Problem*. – Rapport technique, Institut für Informatik, Universität Zu Köln, February 1994.
- [57] Jünger (M.), Reinelt (G.) et Thienel (S.). – *Optimal and provably good solutions for the symmetric traveling salesman problem*. – Rapport technique n° 92.114, Angewandte Mathematik und Informatik, Universität Zu Köln, 1992.
- [58] Kirkpatrick (S.). – Optimization by simulated annealing: Quantitative studies. *Journal of Statistical Physics*, vol. 34, n° 5, 1984.
- [59] Kirkpatrick (S.), Gelatt (C. D.) et Vecchi (M. P.). – Optimization by simulated annealing. *Science*, no220, 1983.
- [60] Laporte (G.), Mercure (H.) et Nobert (Y.). – Generalized traveling salesman problem through n sets of nodes: the asymmetrical case. *Discrete Applied Mathematics*, vol. 18, 1987, pp. 185–197.
- [61] Laporte (G.) et Nobert (Y.). – Generalized traveling salesman through n sets of nodes: an integer programming approach. *INFOR*, vol. 21, 1983, pp. 61–75.
- [62] Laporte (G.) et Nobert (Y.). – Exact algorithms for the vehicle routing problem. *Annals of Discrete Mathematics*, vol. 31, 1987, pp. 147–184.
- [63] Lawler (E.), Lenstra (J.), Rinnooy Kan (A.) et D.B. (S.) (édité par). – *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*. – Wiley, 1985.

-
- [64] Lin (S.) et Kernighan (B.). – An effective heuristic algorithm for the traveling salesman problem. *Operations Research*, vol. 21, 1973, pp. 498–516.
- [65] Little (J.), Murty (K.), Sweeney (D.) et Karel (C.). – An algorithm for the traveling salesman problem. *Operations Research*, vol. 11, 1963, pp. 972–989.
- [66] Metropolis (N.), Rosenbluth (A.), Rosenbluth (M.), Teller (A.) et Teller (E.). – Equation of state calculations by fast computing machines. *Journal of Chem. Physics*, vol. 21, 1953, pp. 1087–1092.
- [67] Naddef (D.). – Handles and teeth in the symmetric traveling salesman polytope. In: *Polyhedral Combinatorics DIMACS series in Discrete Mathematics and Theoretical Computer Science 1, A.M.S.*, éd. par Cook (W.) et SEYMOUR (P.), pp. 61–74.
- [68] Naddef (D.) et Rinaldi (G.). – *The symmetric traveling salesman polytope: New facets from the graphical relaxation*. – Rapport technique n° R. 248, IASI-CNR Rome, 1988.
- [69] Naddef (D.) et Rinaldi (G.). – The graphical relaxation: A new framework for the symmetric traveling salesman polytope. *Mathematical Programming*, vol. 58, 1993, pp. 53–88.
- [70] Nagamochi (H.) et Ibaraki (T.). – Computing edge-connectivity in multigraphs and capacitated graphs. *SIAM Journal on Discrete Mathematics*, vol. 5, 1992, pp. 54–66.
- [71] Nagamochi (H.) et Ibaraki (T.). – A linear-time algorithm for finding a sparse k -connected spanning subgraph of a k -connected graph. *Algorithmica*, vol. 7, 1992, pp. 583–596.
- [72] Noon (C.). – The generalized traveling salesman problem. – 1988. Unpublished Dissertation, Department of Industrial and Operations Research, University of Tennessee.
- [73] Noon (C.) et Bean (J.). – A lagrangian based approach for the asymmetric generalized traveling salesman problem. *Operations Research*, vol. 23, 1991, pp. 623–632.

- [74] Padberg (M.) et Grötschel (M.). – *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*, chap. Polyhedral computations, pp. 307–360. – Wiley, 1985.
- [75] Padberg (M.) et Rao (M.). – Odd minimum cut-sets and b -matchings. *Mathematics of Operations Research*, vol. 7, 1982, pp. 67–80.
- [76] Padberg (M.) et Rinaldi (G.). – An efficient algorithm for the minimum capacity cut problem. *Mathematical Programming*, vol. 47, 1990, pp. 19–36.
- [77] Padberg (M.) et Rinaldi (G.). – Facet identification for the symmetric traveling salesman polytope. *Mathematical Programming*, vol. 47, 1990, pp. 219–257.
- [78] Padberg (M.) et Rinaldi (G.). – A branch and cut algorithm for the resolution of large-scale symmetric traveling salesman problems. *SIAM Review*, vol. 33, 1991, pp. 60–100.
- [79] Prim (R.). – Shortest connection networks and some generalizations. *The Bell System Technical Journal*, vol. 36, 1957, pp. 1389–1401.
- [80] Queyranne (M.) et Wang (Y.). – *Facet tree composition for symmetric traveling salesman polytopes*. – Rapport technique n° 90-MS-C-001, Faculty of Commerce, University of British Columbia, Vancouver, B.C., Canada, 1992.
- [81] Queyranne (M.) et Wang (Y.). – Hamiltonian path and symmetric travelling salesman polytopes. *Mathematical Programming*, vol. 58, 1993, pp. 89–110.
- [82] Reinelt (G.). – TspLib - a traveling salesman problem library. *ORSA Journal on Computing*, vol. 3, 1991, pp. 97–101.
- [83] Rockafellar (R.). – *Convex Analysis*. – Princeton University Press, Princeton, NJ, 1970.
- [84] Rousseau (J.). – Customization versus a general purpose code routing and scheduling problems: a point of view. *In: Vehicle routing: methods and studies*, éd. par Golden (B.) et A. (A.). pp. 469–479. – Elsevier, Amsterdam.

-
- [85] Saksena (J.). – Mathematical model of scheduling clients through welfare agencies. *CORS J.*, vol. 8, 1970, pp. 185–200.
- [86] Salazar (J.). – *Algoritmi per il problema del Commesso Viaggiatore Generalizzato*. – Thèse de PhD, Royal Spanish College in Bologna, 1992.
- [87] Schrijver (A.). – *Theory of Linear and Integer Programming*. – Wiley, Chichester, 1986.
- [88] Sepehri (M.). – *The Symmetric Generalized Traveling Salesman Problem*. – Thèse de PhD, University of Tennessee, Knoxville, 1991.
- [89] smith (T.) et Thompson (G.). – A lifo implicit enumeration search algorithm for the symmetric traveling salesman problem using held and karp's 1-tree relaxation. *Annals of Discrete Mathematics*, vol. 1, 1977, pp. 479–493.
- [90] Srivastava (S.), Kumar (S.), Garg (R.) et Sen (P.). – Generalized traveling salesman problem through n sets of nodes. *CORS J.*, vol. 7, 1969, pp. 97–101.
- [91] Stoer et Witzgall (C.). – *Convexity and Optimization in Finite Dimensions I*. – Springer, Berlin, 1970.
- [92] Taillard (E.). – *Recherches itératives dirigées parallèles*. – Thèse de PhD, EPFL, 1993.
- [93] Ulder (N. L. J.), Pesch (E.), Van Laarhoven (P. J. M.), Bandelt (H. J.) et Aarts (E. H. L.). – *Improving TSP exchange heuristics by population genetics*. – Preprint, Erasmus Universiteit Rotterdam, 1990.
- [94] Van Laarhoven (P. J. M.). – *Theoretical and Computational Aspects of Simulated Annealing*. – Thèse de PhD, Erasmus University, Rotterdam, 1988.
- [95] Van Laarhoven (P. J. M.), Aarts (E. H. L.) et Lenstra (J. K.). – Job-shop scheduling by simulated annealing. *Operations Research*, vol. 40, n° 1, 1992.

- [96] Volgenant (T.) et Jonker (R.). – A branch and bound algorithm for the symmetric traveling salesman problem based on the 1-tree relaxation. *European Journal of Operations Research*, vol. 9, 1982, pp. 83–89.